

# UNIVERSIDAD CATÓLICA DE SANTA MARÍA

FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



**“PROPUESTA DE UNA METODOLOGÍA PARA LA MIGRACIÓN DE APLICATIVOS  
LEGADOS A APLICATIVOS CON TECNOLOGÍA MODERNA EN UNA PLATAFORMA  
WEB, HACIENDO USO DE MEJORES PRÁCTICAS.”**

Tesis presentada por los Bachilleres:

**Candia Serrano, Elvis Frederik**

**Valencia Navarro, José Manuel**

Para obtener el Título Profesional de

**INGENIERO DE SISTEMAS**

**AREQUIPA-PERÚ**

**2016**

## AGRADECIMIENTOS



*Primeramente, agradecemos a Dios por  
estar siempre presente en nuestras vidas,  
a nuestros abuelos, padres, hermanos  
y tíos por su apoyo incondicional a lo largo  
de nuestras vidas personales y profesionales.*

## ÍNDICE

### Contenido

ÍNDICE.....	2
RESUMEN .....	14
ABSTRACT .....	15
INTRODUCCIÓN.....	16
CAPÍTULO I: PLANTEAMIENTO DE LA INVESTIGACIÓN .....	18
1.1. Planteamiento del Problema .....	18
1.2. Objetivos de la Investigación .....	19
1.2.1. General.....	19
1.2.2. Específicos.....	19
1.3. Preguntas de Investigación .....	20
1.4. Línea y Sub-línea de Investigación a la que corresponde el Problema .....	20
1.5. Palabras Clave .....	20
1.6. Solución Propuesta .....	20
1.6.1. Justificación e Importancia.....	20
1.6.2. Descripción de la Solución.....	22
CAPÍTULO II: FUNDAMENTOS TEÓRICOS.....	24
2.1. Estado del Arte .....	24
2.2. Bases Teóricas .....	33

2.2.1. Aplicaciones legadas .....	33
2.2.2. Reingeniería de Software.....	39
2.2.3. Reingeniería de Procesos.....	45
2.2.4. Pmbok.....	53
2.2.5. Juicio de Expertos.....	60
2.2.6. Modelo Vista Controlador – MVC.....	61
2.2.7. Code First .....	63
2.2.8. Scrum.....	64
<b>CAPÍTULO III: MARCO METODOLOGICO .....</b>	<b>68</b>
3.1. Alcances y Limitaciones.....	68
3.2. Aporte .....	68
3.3. Tipo y Nivel de la Investigación.....	68
3.4. Población y Muestra o Universo .....	68
3.5. Técnicas e Instrumentos de Recolección de Datos.....	69
3.6. Plan de análisis estadístico de los datos .....	70
<b>CAPÍTULO IV: PLAN DE TRABAJO .....</b>	<b>71</b>
4.1. Proyecto de Desarrollo de Tesis .....	71
4.2. Diagrama de Gantt del Desarrollo de Tesis.....	73
<b>CAPÍTULO V: PROPUESTA METODOLÓGICA .....</b>	<b>74</b>
5.1. Planificación Inicial de la Migración.....	75
5.1.1. Planificación del proyecto .....	76

5.1.1.1. Equipo de Trabajo .....	76
5.1.1.2. Levantamiento de Necesidades del Aplicativo Migrado .....	82
5.1.1.3. Instrumentos de seguimiento .....	82
5.1.1.4. Recepción del Proyecto .....	84
5.1.2. Entregables .....	86
5.2. Análisis y Diseño.....	86
5.2.1. Involucrados .....	87
5.2.2. Captura de Requisitos.....	88
5.2.3. Inicio de Análisis – Arquitectura.....	89
5.2.3.1. Definición de Objetivos.....	89
5.2.3.2. Definición de Audiencia.....	90
5.2.3.3. Definición de Contenidos de la Migración.....	92
5.2.3.4. Definición de Estructura de la Migración.....	96
5.2.3.5. Definición de Diseño Visual.....	103
5.2.4. Entregables .....	107
5.2.5. Control de Calidad – Análisis y Diseño .....	108
5.3. Construcción.....	108
5.3.1. Involucrados .....	110
5.3.2. Pre-Desarrollo.....	111
5.3.2.1. Transferencia de conocimientos .....	111
5.3.2.2. Tecnologías de Desarrollo .....	111

5.3.2.3. Preparación de Ambientes para Desarrollo .....	112
5.3.3. Desarrollo .....	112
5.3.3.1. Diseño para Acceso Rápido.....	112
5.3.3.2. Normas Mínimas para Facilidad el Acceso - vía conexión telefónica .....	113
5.3.3.3. Normas para Incorporar Elementos Gráficos y Multimedia .....	119
5.3.3.4. Interpolaría.....	120
5.3.3.5. Estándares Internacionales .....	121
5.3.3.6. Diseño para la Accesibilidad .....	122
5.3.3.7. Diseño de la Experiencia del Usuario.....	123
5.3.3.8. Como Atender a los Usuarios .....	129
5.3.4. Aplicar Pruebas Unitarias Especificas.....	130
5.3.5. Entregables .....	131
5.3.6. Control de Calidad.....	133
5.3.7. Proceso de Congelamiento .....	133
5.4. Certificación .....	134
5.4.1. Involucrados .....	135
5.4.2. Pre-Certificación.....	136
5.4.3. Inicio de Certificación .....	138
5.4.3.1. Prueba de Interfaces y Contenidos .....	139
5.4.3.2. Prueba de Funcionabilidades y Operación .....	141
5.4.3.3. Prueba de Carga.....	142

5.4.3.4. Prueba de Seguridad .....	143
5.4.3.5. Prueba de Respaldo y Recuperación .....	144
5.4.4. Registro de Control de pruebas y Errores.....	144
5.4.5. Ciclos de Prueba .....	146
5.4.6. Entregables .....	147
5.5. Implementación .....	147
5.5.1. Involucrados .....	148
5.5.2. Pre-Lanzamiento.....	149
5.5.3. Lanzamiento de Plan Piloto.....	150
5.5.4. Puesta en Marcha.....	151
5.5.5. Lecciones Aprendidas.....	152
5.5.6. Entregables .....	153
<b>CAPÍTULO VI: ANÁLISIS Y DISCUSIÓN DE RESULTADOS .....</b>	<b>154</b>
6.1. Resultados por parte del Área de Tecnologías de Información.....	155
6.1.1. Consideración para la Evaluación de la Metodología .....	155
6.1.2. Instrumentos para la Evaluación de la Metodología .....	155
6.1.3. Descripción del Proceso para la Evaluación de la Metodología .....	155
6.1.4. Resultados de la Evaluación Metodológica.....	156
6.1.5. Análisis de Objetivos Específicos con la Metodología .....	161
6.2. Resultados por parte del Usuario.....	164
6.2.1. Consideración para la Evaluación del Aplicativo Migrado.....	164

6.2.2. Instrumentos para la Evaluación del Aplicativo Migrado .....	164
6.2.3. Descripción de Procesos para la Evaluación del Aplicativo Migrado .....	165
6.2.4. Resultado de la Evaluación del Aplicativo Migrado .....	166
6.2.5. Análisis de los Objetivos Específicos del Aplicativo Migrado .....	175
6.3. Resultados del Aplicativo Migrado VS Aplicativo Legado .....	179
6.3.1. Ingreso al Aplicativo Comisiones.....	179
6.3.2. Menú Principal del Aplicativo Comisiones.....	180
6.3.3. Mantenimiento de Rango Criterio Comisión.....	184
6.3.4. Reporte de Vendedores del Aplicativo Comisión .....	187
6.3.5. Análisis de Resultados con ISO 9126.....	188
CONCLUSIONES.....	202
RECOMENDACIONES .....	203
REFERENCIAS BIBLIOGRÁFICAS .....	204
ANEXOS .....	208

## ÍNDICE DE FIGURAS

Figura 1: Propuesta de simulación propuesto por Pinheiro [11] .....	36
Figura 2: Proceso de mantenimiento de software. [10] .....	38
Figura 3: Propuesta de reingeniería como solución de un problema delimitado.....	39
Figura 4: Pasos de la Reingeniería de Software .....	41
Figura 5: Las Tres C. [15] .....	47
Figura 6: Tres niveles estructurales. [15] .....	48
Figura 7: Proceso metodológico de reingeniería. [15] .....	49
Figura 8: Evolución de la reingeniería. [15].....	50
Figura 9: Grupos básicos de proceso .....	54
Figura 10: Enfoque tradicional para estimación de proyectos. [17].....	57
Figura 11: Estimación de proyectos por método COCOMO [17].....	60
Figura 12: Visión general del proceso [23] .....	64
Figura 13: Estructura del desarrollo ágil – Adaptado [23] .....	65
Figura 14: Visión general de Scrum [24] .....	66
Figura 15: Visión general del modelo Scrum [23] .....	67
Figura 16 Población y muestra del proyecto .....	69
Figura 17: Diagrama de Gantt de la Presente Tesis .....	73
Figura 18: Metodología General de la Migración .....	75
Figura 19: Metodología de la migración - Planificación.....	75
Figura 20: Metodología de Migración - Análisis .....	87

Figura 21: Árbol Funcional .....	98
Figura 22: Estructura de la pagina.....	104
Figura 23: Diagrama de flujo de un proceso funcional .....	105
Figura 24: Boceto de diseño .....	106
Figura 25: Metodología de Migración - Construcción .....	108
Figura 26: Esquema del ambiente de desarrollo.....	109
Figura 27: Ejemplo de tablas para separar.....	115
Figura 28: Ejemplo de tablas que dificultan.....	116
Figura 29: Ejemplo de uso de frames para .....	117
Figura 30: Ejemplo de diagrama de interacción.....	124
Figura 31: Congelar entregables en el servidor de desarrollo .....	134
Figura 32: Metodología de migración - Certificación.....	134
Figura 33: Esquema del ambiente de certificación.....	135
Figura 34: Proceso de descongelamiento en certificación.....	137
Figura 35: Proceso de congelamiento y reversión.....	138
Figura 36: Ciclos de prueba.....	147
Figura 37: Metodología de migración – Implementación .....	148
Figura 38: Esquema de ambiente de producción.....	148
Figura 39: Proceso de descongelamiento en producción.....	149
Figura 40: Esquema del Plan Piloto .....	151
Figura 41: Gráfico de la pregunta 1 .....	156

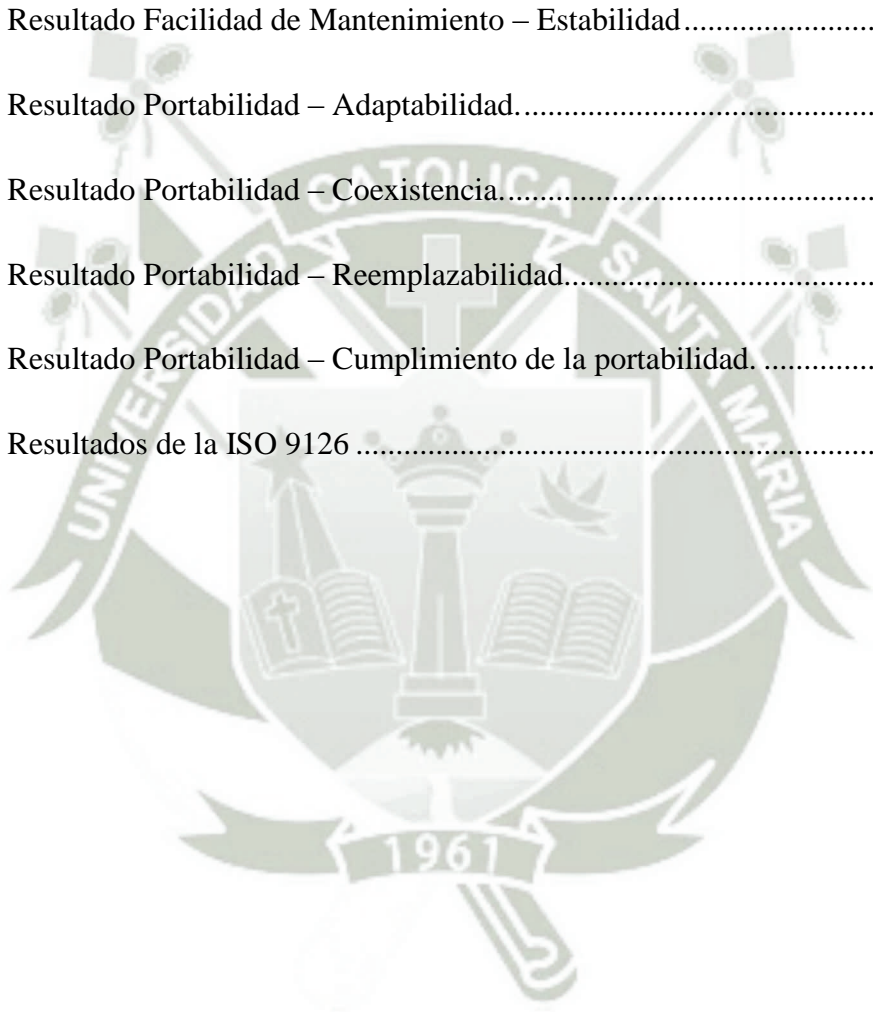
Figura 42: Gráfico de la pregunta 2.....	157
Figura 43: Gráfico de la pregunta 3.....	158
Figura 44: Gráfico de pregunta 4.....	159
Figura 45: Gráfico de la pregunta 5.....	160
Figura 46: Gráfico de resultados del objetivo específico 1 .....	161
Figura 47: Gráfico de resultado del objetivo específico 2.....	162
Figura 48: Gráfico de resultado del objetivo específico 3.....	163
Figura 49: Gráfico de la pregunta 1.....	166
Figura 50: Gráfico de la pregunta 2.....	167
Figura 51: Gráfico de la pregunta 3.....	168
Figura 52: Gráfico de la pregunta 4.....	169
Figura 53: Gráfico de la pregunta 5.....	170
Figura 54: Gráfico de la pregunta 6.....	171
Figura 55: Gráfico de la pregunta 7.....	172
Figura 56: Gráfico de la pregunta 8.....	173
Figura 57: Gráfico de la pregunta 9.....	174
Figura 58: Gráfico de resultado del objetivo específico 1 .....	176
Figura 59: Gráfico de resultados del objetivo específico 2 .....	176
Figura 60: Gráfico de resultados del objetivo específico 3 .....	177
Figura 61: Gráfico de resultados del objetivo específico 4 .....	178
Figura 62: Ingreso al aplicativo legado .....	179

Figura 63: Ingreso al aplicativo migrado.....	180
Figura 64: Menú principal - Legado.....	180
Figura 65: Sub-Menú Mantenimiento - Legado .....	180
Figura 66: Sub-Menú Reportes - Legado .....	181
Figura 67: Menú Principal - Migración.....	181
Figura 68: Menú de opciones - Migración .....	182
Figura 69: Menú de Widgets – Migración .....	183
Figura 70: Mantenimiento de rango - Legado.....	184
Figura 71: Editar Rango - Legado .....	184
Figura 72: Mantenimiento de Rango - Migración.....	185
Figura 73: Editar Rango - Migración .....	186
Figura 74: Reporte de vendedores - Legado.....	187
Figura 75: Reporte de vendedores – Migración .....	188

## ÍNDICE DE TABLAS

Tabla 1: Evolución de un sistema - Gibson y Nolan 1974 .....	27
Tabla 2: Técnica e instrumento de recolección de datos. ....	69
Tabla 3: Proyecto de Desarrollo de la Presente Tesis .....	72
Tabla 4: Resultado de la pregunta 1 .....	156
Tabla 5: Resultado de la pregunta 2 .....	157
Tabla 6: Resultado de la pregunta 3 .....	158
Tabla 7: Resultado de la pregunta 4 .....	159
Tabla 8: Resultado de la pregunta 5 .....	160
Tabla 9: Análisis Objetivos Específicos – Área de TI.....	161
Tabla 10: Resultado de la pregunta 1 .....	166
Tabla 11: Resultado de la pregunta 2 .....	167
Tabla 12: Resultados de la pregunta 3.....	168
Tabla 13: Resultado de la pregunta 4 .....	169
Tabla 14: Resultado de la pregunta 5 .....	170
Tabla 15: Resultados de la pregunta 6.....	171
Tabla 16: Resultado de la pregunta 7 .....	172
Tabla 17: Resultado de la pregunta 8 .....	173
Tabla 18: Resultado de la pregunta 9 .....	174
Tabla 19: Análisis de Objetivos Específicos - Usuario .....	175
Tabla 20: Resultado de la Funcionalidad - Aplicabilidad .....	189

Tabla 21: Resultado de Fiabilidad - Madurez .....	190
Tabla 22: Resultados de Usabilidad - Entendibilidad .....	191
Tabla 23: Resultado de eficiencia – Comportamiento en el Tiempo .....	193
Tabla 24: Resultado Facilidad de Mantenimiento – Analizabilidad .....	194
Tabla 25: Resultado Facilidad de Mantenimiento – Cambiabilidad .....	195
Tabla 26: Resultado Facilidad de Mantenimiento – Estabilidad.....	196
Tabla 27: Resultado Portabilidad – Adaptabilidad.....	197
Tabla 28: Resultado Portabilidad – Coexistencia.....	198
Tabla 29: Resultado Portabilidad – Reemplazabilidad.....	199
Tabla 30: Resultado Portabilidad – Cumplimiento de la portabilidad.....	200
Tabla 31: Resultados de la ISO 9126 .....	201



## RESUMEN

El objetivo principal del presente trabajo de tesis, es plantear una metodología donde nos permita realizar una migración adecuada, de aplicativos legados a aplicativos con tecnología moderna haciendo uso de una plataforma web, basándonos además en las mejores prácticas.

Para satisfacer las necesidades, se realizó la metodología donde se propone contar con cinco fases, que son las siguientes: Planificación, Análisis, Construcción, Certificación y como fase final la Implementación. Siendo cada una de estas fases de vital importancia para lograr un adecuado proceso de migración y llegar al objetivo principal de la presente tesis.

Los resultados que se han obtenido de esta metodología han sido satisfactorios debido al orden y los controles que se han dado en cada fase, además de haber satisfecho las necesidades de las personas involucradas en el proceso de migración y sobre todo la satisfacción de los usuarios finales.

En conclusión, queda demostrado que, si es posible realizar una migración de aplicativos legados usando la metodología propuesta, además de que en nuestra región no se cuenta con metodología alguna que nos permita realizar este tipo de migraciones, siendo esta propuesta muy útil para solucionar dicho problema que gran parte de empresas tecnológicas de nuestra ciudad afrontan actualmente.

## ABSTRACT

The main objective of this thesis is to propose a methodology which allows us to make a proper migration of legacy applications to modern technology applications, using a web platform based on best practices.

To meet the needs of the methodology, it is proposed to have five phases, which are the following: Planning, Analysis, Construction, Certification and the final phase of Implementation. Each of these phases are vital for a proper migration process and to reach the main goal of this thesis.

The results obtained from this methodology have been satisfactory because of the order and controls that have been performed in each phase, as well as having satisfied the needs of the people involved in the migration process and especially the satisfaction of the end-users.

In conclusion, it is shown that the migration of legacy applications using the proposed methodology is possible, in addition to that; our region does not have a methodology that allows us to perform this type of migration. This proposal is useful and solves the problem that most of the technology companies are currently facing in our city.

## INTRODUCCIÓN

Hoy en día las empresas a nivel mundial tienen problemas con los sistemas de información, debido a la antigüedad de estas; el principal problema es que no se cuenta con el personal de soporte capacitado, ya que estos sistemas cuentan con aplicativos que fueron realizados con lenguajes de programación antiguos que hoy en día ya no son comerciales, o en caso de que se encuentre la persona idónea, suele ser demasiado costosa para las empresas y tal vez por soluciones sencillas. Por otro lado, las empresas como Microsoft dejaron de brindar soporte a varios de sus programas, como SQL server 2000, SQL server 2005, Visual Basic.NET 2002, Visual FoxPro 6.0, etc.

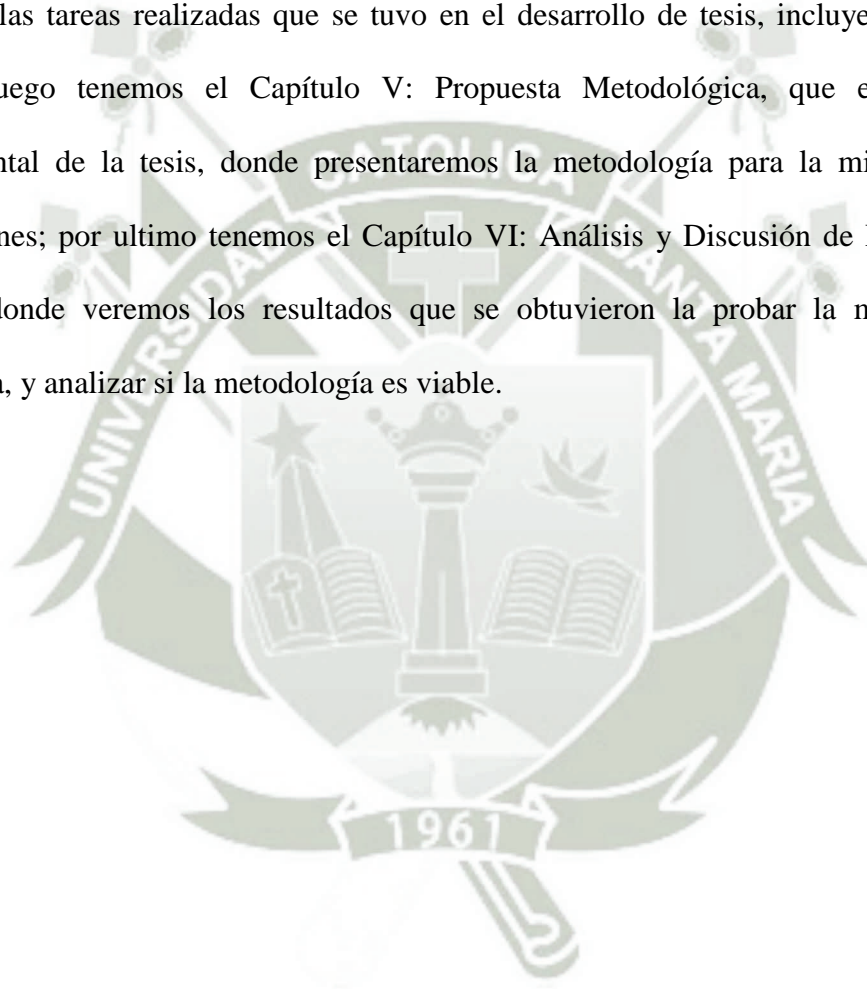
Los problemas mencionados llegaron a nuestro país hace unos años, sobre todos a instituciones bancarias, ya que estas son las que cuentan con sistemas de información antiguos, viendo la opción de adquirir nuevos sistemas de información denegada, ya que son muy costosas y de gran riesgo. Este problema no es ajeno en la ciudad de Arequipa, donde se empezó a dar los casos mencionados.

En base a lo indicado surgen las siguientes interrogantes: ¿Qué sucedería si nosotros proponemos una metodología para que sea usada en alguna fábrica de software en la ciudad de Arequipa? ¿será adecuada? ¿Solucionará el problema?

Para resolver las interrogantes, se propone la presente metodología, donde se brinda una solución, que es la migración de aplicativos legados a un nuevo aplicativo migrado usando tecnología moderna, empleando los siguientes pasos: Planificación, Análisis, Construcción, Certificación e Implementación.

La presente tesis está dividida en 6 capítulos, siendo el Capítulo I: Planeamiento de la investigación, donde se presenta el objetivo de la tesis de acuerdo al problema que

encontramos; seguidamente tenemos el Capítulo II: Fundamentos Teóricos, en el que veremos el estado del arte de acuerdo a investigaciones pasadas que involucren el tema de tesis, además de las bases teóricas de la investigación; luego tenemos al Capítulo III: Marco Metodológico, en donde se presenta el alcance y limitaciones de la tesis, además del tipo de investigación y las mediciones de los resultados que se obtendrán al final de la tesis; seguidamente está el Capítulo IV: Plan de Trabajo, en que básicamente se presenta las tareas realizadas que se tuvo en el desarrollo de tesis, incluyendo fechas reales; luego tenemos el Capítulo V: Propuesta Metodológica, que es la parte fundamental de la tesis, donde presentaremos la metodología para la migración de aplicaciones; por ultimo tenemos el Capítulo VI: Análisis y Discusión de Resultados, que es donde veremos los resultados que se obtuvieron la probar la metodología propuesta, y analizar si la metodología es viable.



## CAPÍTULO I: PLANTEAMIENTO DE LA INVESTIGACIÓN

### 1.1. Planteamiento del Problema

En nuestro país, a finales de la década de los 90 se tiene un mejor concepto sobre las tecnologías de información, y a partir del año 2000 las organizaciones en su ambición de cumplir las necesidades del negocio, optimizar los servicios y los tiempos de respuesta que manejan en la actualidad, optan por adquirir sistemas de Información. Las organizaciones han recurrido a la utilización de la tecnología de información para lograr una ventaja competitiva, como base para la toma de decisiones y lograr ventajas competitivas o reducir la ventaja de la competencia. Dichas tecnologías de información deben brindar información exacta, oportuna y veraz, que les permita colaborar en el planeamiento de las estrategias de negocio y reforzar sus fortalezas internas como empresa frente a la competencia.

Ya en el año 2016 los sistemas de información creados a principios del 2000, son llamados sistemas legados, que se han definido como software que es vital para las organizaciones, ya que brinda información importante para la toma de decisiones. Los sistemas legados son uno de los principales problemas del mantenimiento de software. A pesar que la crisis Y2K forzó a muchas organizaciones a sustituir los sistemas antiguos, existe aún software que supera los 15 años. Se trata de software monolítico, dividido en subsistemas que cumplen funciones importantes para las organizaciones, pero de los cuales no se dispone documentación, los desarrolladores originales ya no trabajan en la organización y solo se dispone de los códigos fuentes e incluso en algunos casos ni siquiera eso, en otros casos ya no existen desarrolladores con conocimientos en lenguajes de programación arcaicos, haciendo dependientes a un sistema de información de un desarrollador [1].

Generalmente las organizaciones no invierten en reingeniería a las nuevas tecnologías sobre la base de que los costes de mantenimiento son menores. Sin embargo, en un punto el mantenimiento adaptativo obliga a realizar el mayor esfuerzo y más riesgoso, el desarrollo de un nuevo sistema, la re-ingeniería, la recuperación de la lógica de negocios, es decir una migración de aplicativos. Los procesos de reingeniería son complejos, y no existen fórmulas preestablecidas que se puedan aplicar con éxito a todos los casos, es por esta razón que se desarrolla el presente proyecto planteando una metodología para la migración de aplicaciones legadas una plataforma web.

## **1.2. Objetivos de la Investigación**

### **1.2.1. General**

Resolver el problema que se tiene al migrar aplicaciones legadas usando las mejores prácticas para realizar una metodología.

### **1.2.2. Específicos**

- Demostrar que es factible realizar una migración con la metodología propuesta.
- Utilizar procesos de calidad para la migración de aplicaciones.
- Demostrar que las mejores prácticas son útiles para migrar un aplicativo.
- Demostrar que un aplicativo migrado, puede ser mejor usado que un aplicativo legado.
- Demostrar que el aplicativo migrado es igual de confiable que el aplicativo legado, respecto a la información que muestra.
- Demostrar que el aplicativo migrado es portable.

- Demostrar que el aplicativo migrado es más flexible en comparación al aplicativo legado.

### 1.3. Preguntas de Investigación

¿Será posible realizar una metodología que nos permita ejecutar la migración de aplicativos legados a aplicativos con tecnología moderna, haciendo uso de las mejores prácticas?

### 1.4. Línea y Sub-línea de Investigación a la que corresponde el Problema

**Línea:** Ingeniería del Software

**Sub-línea:**

- Métodos para la innovación tecnológica
- Gestión de procesos

### 1.5. Palabras Clave

- Metodología.
- Mejores prácticas.
- Aplicaciones Legadas.

### 1.6. Solución Propuesta

#### 1.6.1. Justificación e Importancia

Los sistemas de software tienen un tiempo de vida muy versátil, numerosos sistemas han sido creados desde hace más de 15 años y cuentan con aplicativos de trabajo crítico. Esto es, la continuidad operativa de las empresas que los operan depende de los servicios provistos por el sistema y una falla en este puede traer consecuencias negativas sobre la operación diaria del negocio. Las

decisiones que se toman sobre este tipo de sistemas son de orden estratégico debido a su importancia.

Luego de que los sistemas de información se encuentran en funcionamiento, existen diversos factores, ya sean internos, externos, económicos, de mercado, legales, administrativos o políticas de organización, que exigen cambios continuos en el negocio. Dichos cambios originan alteraciones en los requerimientos de software que recaen sobre los sistemas tecnológicos, obligando a cambios en los aplicativos para alinearse a los nuevos requerimientos del negocio.

El número elevado de modificaciones en un sistema tiende a desestabilizar su estructura y hacer que los cambios subsiguientes sean difíciles de incorporar. Entre mayor sea el número de cambios en el sistema, menor se hace la calidad del sistema.

Brindar mantenimiento a sistemas legados usualmente es complejo ya que se carece de lo siguiente:

- Los diversos aplicativos del sistema se encuentran desarrollados en lenguajes de programación desactualizados. Contar con recursos que dominen estos lenguajes no es sencillo por lo que el costo de mantenimiento del sistema es elevado.
- La documentación de los sistemas es escasa e insuficiente.
- El número de años de mantenimiento continuo sobre el sistema hace que la estructura del software se incremente sustancialmente y sea propensa a la existencia de código muerto dentro de algunos programas.

- Existen limitaciones propias de las plataformas y sistemas legados, como limitantes en tamaño de los nombres de archivo, limitantes en el tamaño disponible para carga de programas en memoria, limitantes en la longitud de las variables, etc.
- Existencia de duplicidad de datos, datos desactualizados y datos incompletos.
- La base de soporte disponible para un sistema legado es limitada, y en ocasiones requerida tanto para aspectos técnicos, como funcionales y de proceso. La mayor parte del conocimiento del sistema se encuentra concentrado en pocas personas, y la disponibilidad de estos recursos puede no ser inmediata o no responder con la agilidad que se requiere.

### **1.6.2. Descripción de la Solución**

Cuando se trata de sistemas legados con aplicativos de trabajo crítico, se enfrenta un dilema propio a la evolución de un sistema:

- Continuar utilizando el sistema e introduciendo cambios en él (implica que los costos de mantenimiento se vean gradualmente incrementados) y la complejidad del sistema continúe en ascenso.
- Reemplazar el sistema por uno nuevo puede ser muy costoso y el soporte de los procesos de negocio puede resultar no ser tan eficiente como el ofrecido por el sistema legado.
- Realizar la migración de los aplicativos legados a una tecnología moderna, manteniendo la lógica de negocios de dicho sistema legado.

El presente proyecto presenta una metodología de software para resolver los problemas mencionados de los sistemas legados, migrando los aplicativos a un entorno web con una tecnología moderna. El propósito es usar las mejores prácticas para proporcionar una alternativa de solución en el proceso de la migración.



## CAPÍTULO II: FUNDAMENTOS TEÓRICOS

### 2.1. Estado del Arte

Según [2], nos muestra una breve descripción de la evolución de un sistema de información, indicando que durante los últimos años los sistemas de información constituyen uno de los principales ámbitos de estudio en el área de organización de empresas. El entorno donde las compañías desarrollan sus actividades se vuelve cada vez más complejo. La creciente globalización, el proceso de internacionalización de la empresa, el incremento de la competencia en los mercados de bienes y servicios, la rapidez en el desarrollo de las tecnologías de información, el aumento de la incertidumbre en el entorno y la reducción de los ciclos de vida de los productos originan que la información se convierta en un elemento clave para la gestión, así como para la supervivencia y crecimiento de la organización empresarial. Si los recursos básicos analizados hasta ahora eran tierra, trabajo y capital, ahora la información aparece como otro insumo fundamental a valorar en las empresas.

Los Sistemas de información han ido evolucionando durante los últimos años hasta constituir los denominados sistemas de información estratégicos. Primeramente, los sistemas de información empresariales eran considerados como un instrumento simplificador de las distintas actividades de la empresa, una herramienta con la cual se facilitaban los tramites y reducía la burocracia. Su finalidad era básicamente llevar la contabilidad y el procesamiento de los documentos que a nivel operativo.

Posteriormente el desarrollo de la informática y las telecomunicaciones permitieron incrementar la eficacia en la realización de las tareas, ahorrar tiempo en el

desarrollo de las actividades y almacenar la mayor cantidad de información en el menor espacio posible, lo cual aumentó en las organizaciones el interés en los sistemas de información. Con el transcurrir del tiempo las empresas fueron observando como las tecnologías y sistemas de información permitían a la empresa obtener mejores resultados que sus competidores, constituyéndose por sí mismas como una fuente de ventaja competitiva y una poderosa arma que permitía diferenciarse de sus competidores y obtener mejores resultados que estos. De este modo los sistemas de información se constituyeron como una de las cuestiones estratégicas de la empresa, que ha de considerarse siempre en todo proceso de planificación empresarial.

Dada la clasificación de Kennet y J Laudon, los primeros sistemas de información en desarrollarse fueron los sistemas de procesamiento de operaciones. Con el transcurrir del tiempo, fueron apareciendo en primer lugar los sistemas de información para la administración y finalmente los sistemas de apoyo a las decisiones, así como los sistemas estratégicos. Se produjo un desarrollo vertical de los sistemas de información, partiendo de los niveles inferiores de la organización hasta abarcar al equipo directivo de la empresa. [2]

A la hora de analizar el progreso de los sistemas de información, uno de los trabajos fundamentales fue el propuesto por Gibson y Nolan (1974) [2], demostrando que en esos años ya se tenía idea de cómo tendría que ser el ciclo de vida de un sistema de información, obviando que algún día tendríamos que requerir a una migración de sistemas. Ellos describieron la evolución de los sistemas de información basándose en la evolución de las tecnologías de información (Ver tabla 1). En la medida en que se desarrollaron los equipos informáticos, el software, el hardware, las bases de

datos y las telecomunicaciones, los sistemas de información fueron adquiriendo una mayor relevancia en las organizaciones, empezándose a considerar como un elemento más del proceso de planificación.

Etapas de la evolución de los sistemas	Características
<p><b>1.- Iniciación</b></p>	<ul style="list-style-type: none"> <li>• Introducción de la informática en la empresa.</li> <li>• Aplicaciones informáticas orientadas a la mecanización y automatización de los procesos ordinarios.</li> <li>• Escaso gasto en informática y escasa formación del personal.</li> </ul>
<p><b>2.- Contagio</b></p>	<ul style="list-style-type: none"> <li>• Las aplicaciones de las tecnologías de información originan resultados espectaculares.</li> <li>• Difusión de las tecnologías de información en todas las áreas de la empresa.</li> <li>• Aumenta la cualificación del personal.</li> <li>• Existe gran descoordinación y poca planificación en el desarrollo de los sistemas de información</li> </ul>
<p><b>3.- Control</b></p>	<ul style="list-style-type: none"> <li>• La alta dirección de la organización se preocupa de los sistemas de información como consecuencia del alto coste en ellos.</li> <li>• Centralización de los proyectos de inversión en tecnologías de información</li> </ul>

<p><b>4.- Integración</b></p>	<ul style="list-style-type: none"> <li>• Se controla el incremento del gasto.</li> <li>• Se produce la integración de los sistemas de información existentes en las distintas áreas de la empresa.</li> <li>• Mejora y perfeccionan los sistemas de información.</li> </ul>
<p><b>5.- Administración de la Migración</b></p>	<ul style="list-style-type: none"> <li>• El sistema de información adquiere una dimensión estrategia en la empresa.</li> <li>• Descentralización de ciertas aplicaciones informáticas.</li> </ul>
<p><b>6.- Madurez</b></p>	<ul style="list-style-type: none"> <li>• Desarrollo de los Sistemas de información en los niveles superiores de la organización apareciendo los Sistemas Estratégicos de información.</li> <li>• Adquiere gran importancia la creatividad y la innovación</li> </ul>

*Tabla 1: Evolución de un sistema - Gibson y Nolan 1974*

Revisando la tabla 1, podemos apreciar que aún no se tenía conocimiento de que el soporte técnico por parte de los proveedores algún día serían escasos, o que aparecerían nuevas herramientas y lenguajes de programación. En la presente tesis analizamos que al cuadro presentado por Gibson y Nolan, se tendría que agregar una nueva etapa, que sería la de migración.

Por otro lado [3], nos da una perspectiva más contemporánea de que sucedió con los distintos sistemas legados y haciendo mención de que ya existen nuevas tecnologías, indicando que la evolución de cualquier producto de software es

inevitable. La misma es causada por errores que deben ser reparados, nuevos requerimientos que se suscitan, o nuevas tecnologías que aparecen durante el ciclo de vida del producto. En consecuencia, los sistemas legados de información son candidatos a evolucionar. Por definición, un sistema legado es software en funcionamiento que todavía permanece activo, pero implementado con criterios de diseño y tecnologías desactualizados.

Un ejemplo emblemático de sistema legado son los sistemas monolíticos en mainframes, que usualmente pertenecen a bancos o entidades gubernamentales. Estos sistemas mantienen décadas de información en un *mainframe* que les aporta disponibilidad, escalabilidad, seguridad y gran capacidad para la realización de consultas, transacciones y almacenamiento de datos utilizando tecnología específica para realizar transacciones como COBOL (De acuerdo a Gartner (<http://www.gartner.com>), más de 200 billones de líneas de código fuente COBOL operativo están aún corriendo mundialmente.). Actualmente, algunas organizaciones se ven forzadas a evolucionar sus sistemas legados debido a la escasez de personal experto en tecnologías legadas, altos costos para incrementar la potencia de cómputo de los *mainframes*, y problemas de integración con tecnologías.

La evolución de un sistema implica tres etapas principales: mantenimiento, modernización y reemplazo. Un sistema es creado y luego necesita ser mantenido a fin de reparar inconsistencias y errores menores. Cuando se requieren cambios mayores, el sistema entra en etapa de modernización. Finalmente, cuando el sistema original no alcanza para satisfacer la evolución de los requerimientos, este debe ser reemplazado.

De acuerdo a esto y por definición, la migración en sí está comprendida dentro de la etapa de modernización de un sistema.

Una estrategia que ha ganado interés en los últimos años es la migración hacia *Service-Oriented Architectures* (SOAs). Con SOA, los sistemas son compuestos por funciones reutilizables e independientes entre sí denominadas servicios, los cuales son generalmente materializados mediante Servicios Web. La migración hacia SOA permite incrementar la reusabilidad de funcionalidades legadas generando interfaces de servicios independientes entre sí, y mejorar la integración con tecnologías modernas. La mayoría de los sistemas legados aún no se han adaptado hacia SOA, como tampoco a arquitecturas Web convencionales. Por esto, existen enfoques de reemplazo que abarcan los aspectos relevantes que implica un proceso de migración, como la ingeniería reversa para el entendimiento de la funcionalidad legada y el relevamiento del modelo relacional de las bases de datos, pero generan un nuevo sistema. [3]

Según [4] nos indica que hoy en día es frecuente encontrar dentro de las organizaciones aplicaciones que fueron desarrolladas desde los años 80 hasta esta época. A raíz de las restricciones que existían en aquel entonces, estas aplicaciones eran dependientes de los grandes fabricantes de hardware y software, generando una dependencia total por parte de las organizaciones. Estos sistemas frecuentemente no estaban bien documentados, lo que llevaba a una tarea de mantenimiento bastante compleja. Fueron construidos de forma rígida, con el objetivo específico de solucionar los requerimientos funcionales; dejando de lado aspectos importantes como la evolución, extensibilidad, portabilidad y compatibilidad entre otros.

Estas necesidades de evolución y de implementación de nuevos requerimientos, hacen necesario un mantenimiento que no es fácil de llevar a cabo, lo que termina desencadenando en el planteamiento de algunas alternativas como: Llevar a cabo un nuevo desarrollo que contemple la funcionalidad del sistema legado más los nuevos requerimientos, otra alternativa es evaluar un cambio al sistema legado que permita adicionar nuevos requerimientos de una forma sencilla, o la última opción es simplemente desechar el sistema y adquirir una herramienta del mercado que supla las necesidades actuales de la organización.

Una manera de darse cuenta que trabaja con un sistema legado, es ver si está relacionada con los siguientes puntos:

- Se trata de un sistema escrito en lenguajes de varios años atrás como: Cobol, RPG, PL, Assembler o lenguajes de programación que ya no cuentan con soporte por parte del proveedor.
- Normalmente estos sistemas son soportados por un DBMS (manejador de base de datos) ya obsoleto.
- La interface de cliente está basada en terminales (3270 [16] o 5250 [7]) las cuales no son gráficas.
- Las aplicaciones que componen el sistema, son frecuentemente monolíticas, las cuales fueron desarrolladas para suplir una necesidad de la organización, separadas del resto del sistema, presentando una integración normalmente vertical.
- Es un sistema fundamentalmente dirigido a la parte operativa de las organizaciones, normalmente de misión crítica para la organización que requiere estar operando todo el tiempo.

- Suelen ser sistemas bastante complejos y grandes (millones de líneas de código y lógica de negocio).
- Estos tipos de sistema suelen no tener ningún tipo de documentación, no es posible hacer trazabilidad de funcionalidad.

Hoy en día es una realidad para todas las personas involucradas en tecnología, encontrarse con cientos de sistemas legados dentro de las organizaciones.

Las tareas de mantenimiento del sistema eran tareas operativas asociadas a la constante evolución del software. Últimamente estas labores de mantenimiento han adquirido una gran importancia que requiere de un profundo conocimiento del sistema, lo que es conocido en la reingeniería del software como reingeniería de tipo *white-box*, o limitarse a las interfaces externas del sistema conocidas como tipo *black-box*.

El tipo *white-box*, asociado generalmente al encapsulamiento de funcionalidad legada.

El tipo *black-box*, asociado la reingeniería de la funcionalidad legada, se limita al estudio de las interfaces externas del sistema para la adición de encapsulamiento (*wrapping*).

En la ingeniería reversa, el proceso de análisis del sistema, comprende la identificación de todos los módulos existentes, y la interrelación existente entre ellos, creando una representación de alto nivel de abstracción. A partir de esto, si es posible llegar a pensar en sustituir el sistema.

El *wrapping*, evita la comprensión detallada de la estructura interna del sistema, limitándose a conocer las interfaces ofrecidas y que se encuentran funcionando

correctamente. El *wrapping* se basa en el principio de encapsulamiento, idea original del paradigma de programación orientada a objetos, y el cual consiste en ocultar el detalle de la implementación de un objeto, mientras se provee una interfaz por medio de sus métodos permitidos.

Una aproximación a las alternativas existentes para el tratamiento de los sistemas legados puede ser:

- Retirar el sistema, excluir el sistema y reemplazarlo con un sistema comercial.
- Sustitución completa, reescribir el sistema en su totalidad.
- Migración gradual, realizar una transformación del sistema, de forma gradual.
- Integración, considerar el mantener el sistema, sin modificarlo exportando su funcionalidad efectuando *wrapping* con algún tipo de tecnología.

Una de las posibles soluciones para aplicativos legados, según Nolberto Rojas, nos indica que una migración puede considerarse como el paso de un sistema existente hacia un nuevo sistema, siendo un proceso altamente riesgoso.

Se podría pensar en una migración total o parcial, cada una de ellas puede ofrecer algunas ventajas y desventajas.

Migración parcial, solo una parte del sistema es transformada, generalmente se hace con aquella que tiene mayores problemas, mayores costos de mantenimiento, menor flexibilidad etc. Esta suele ser una solución factible con un riesgo normal.

Migración completa, el riesgo es bastante alto, se hace una migración total del sistema y es necesario redefinir arquitectura, diseño y todo lo que involucra crear un sistema.

Migración de interfaz cliente, la gran mayoría de los sistemas legados, aun hoy en día son accedidos mediante terminales brutas, una típica son las terminales 3270.

El mercado actual ofrece algunas herramientas, que pueden ser usadas como instrumento, para generar interfaces GUI (Interface Gráfica de Usuario) o una interacción vía Web. Esta tecnología es conocida como *MiddleWare de Screen Scraping* basados en lenguajes de scripting. [4]

Revisando estos puntos podemos deducir que actualmente existen empresas que ya cuentan con el problema de tener un sistema legado, siendo una de las soluciones migrar el sistema, pero como se indica en [4], estas suelen poseer un riesgo relativamente elevado, es por esta razón que en esta presente tesis realizaremos una metodología para realizar una migración exitosa y con menor riesgo.

## 2.2. Bases Teóricas

### 2.2.1. Aplicaciones legadas

Los sistemas legados se han definido como “software que es vital para la organización, pero no se sabe qué hacer con él”. [22]

Debido a esta realidad, la noción tradicional de arquitecturas corporativas controladas de forma central se torna obsoleta y surge un nuevo paradigma de soluciones basadas en servicios débilmente acoplados. La realidad es que la gran mayoría de los sistemas actuales están corriendo en plataformas legadas y, si bien existe el deseo corporativo de reemplazar estos sistemas legados, el mismo

se ve obstruido por restricciones como son las inversiones realizadas en estos sistemas, la falta de personal técnico adecuado, así como una pérdida de conocimiento en el funcionamiento interno de estos sistemas. Como consecuencia, ninguna solución empresarial estaría completa sin una solución viable de incorporar estos sistemas a sus aplicaciones compuestas. [5]

#### **a. Características de los Sistemas Legados**

Un sistema legado es una aplicación de software que posee un importante valor de negocio, aunque se ha debilitado por el constante cambio tecnológico, un pobre soporte arquitectónico, alto costo de mantenimiento y documentación inexistente, se mantiene en producción [7]. No necesariamente un sistema viejo es un sistema legado, más bien se le considera a aquel que se resiste a modificaciones y actualizaciones perdiendo competitividad [6], y aunque fue desarrollado aplicando adecuados principios de ingeniería, su arquitectura no es lo suficientemente flexible como para absorber los cambios provocados por nuevos requerimientos [6]. Además, en algunos casos no se establecieron principios de evolución y actualización del software durante el ciclo de desarrollo del software. En su mayoría los sistemas legados son monolíticos, poseen un enfoque limitado y son incapaces de crecer y evolucionar fácilmente, o su código fuente no está disponible. También se debe tener en cuenta que las continuas modificaciones de un sistema legado tienen efectos acumulativos que incrementan su complejidad [9]. Finalmente, en muy pocos casos a pesar de las exigentes necesidades de cambio, se preserva como un software inmutable, porque el costo de reemplazarlo o modificarlo es inviable [8].

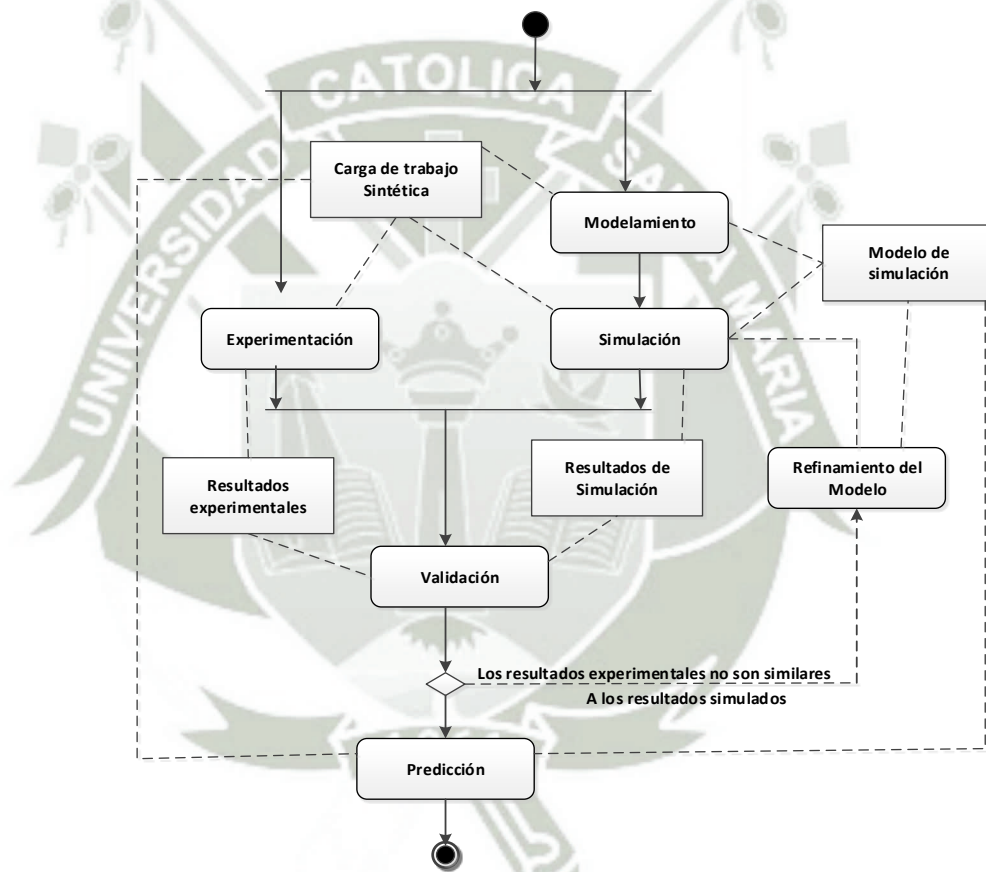
## **b. Alternativas para actualización de Sistemas Legados**

Hablar de actualizar los sistemas legados implica considerar aspectos como el tamaño del código. Un estudio realizado por Ulrich en 1990, que es referenciado por Sommerville [6], presenta cifras contundentes: 120 mil millones de líneas de código es el estimativo del tamaño de los sistemas legados, en su mayoría escritas en COBOL y FORTRAN, a las que hoy en día se suman las escritas en lenguajes estructurados como C++, Java, Visual Basic y Pascal entre otros. Veamos a continuación algunas de estas alternativas:

### **i. Abandonar el sistema legado para dar paso a uno nuevo**

Cuando se ha perdido la contribución del sistema a los procesos de negocio de la organización y el costo de hacer reingeniería resulta muy alto, y es más razonable invertir en nuevas tecnologías de hardware o software. Un aspecto esencial de esta perspectiva es planear la migración del sistema legado a un nuevo sistema. A este respecto, Pinheiro [11] propone un proceso de simulación para determinar el comportamiento antes de hacerlo operativo. El modelo propuesto fundamenta las actividades de modelamiento, experimentación y predicción sobre la base de una carga de trabajo sintética establecida a partir de la información recogida tanto del nuevo sistema como del sistema legado. Se elabora una guía de desempeño del nuevo sistema comparado con el actual, que es una tarea compleja si se tiene en cuenta que el sistema legado puede tener

no miles sino millones de líneas de código [11]. La figura 1 muestra el proceso de simulación para evaluar el desempeño de un sistema objetivo mediante la producción de un modelo valido. En la etapa de modelamiento se especifica el modelo, en la actividad de simulación se ejecuta el modelo y se producen las salidas de la simulación, que, junto a los resultados de la experimentación, permiten hacer la validación.



**Figura 1: Propuesta de simulación propuesto por Pinheiro [11]**

Si los resultados son similares el modelo se considera válido, de lo contrario se somete a una actividad de refinamiento y se vuelve a hacer la simulación, esto se repite hasta que la diferencia de resultados entre simulación y experimentación sea escasa. El modelo

validado y aceptado se pasa a la actividad de predicción del que resulta la descripción del comportamiento del nuevo sistema. [13]

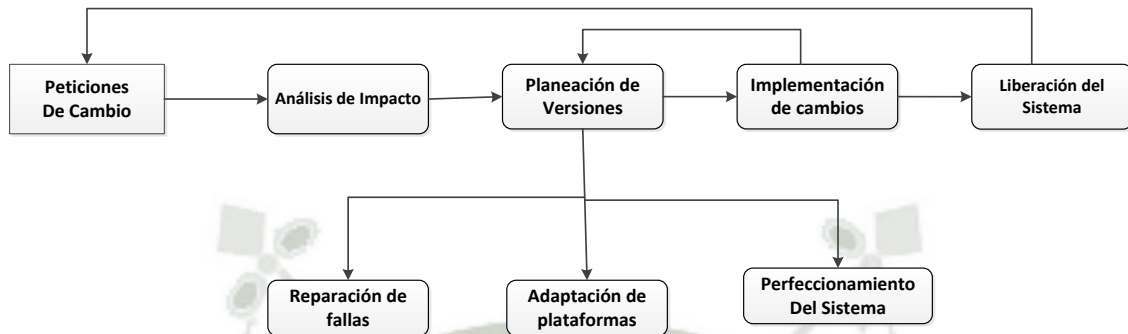
**ii. Garantizar mantenimiento del sistema actual**

Es la forma natural de hacer evolucionar un sistema de software, y consiste en realizar cambios sobre componentes ya existentes o adicionar nuevos. Los cambios se pueden hacer en varios niveles: el primero es corregir errores puntuales de codificación y solucionar errores en diseño, denominado mantenimiento correctivo en que las modificaciones no son tan costosas y pueden abarcar todo el ciclo de desarrollo de software. [13]

El siguiente nivel es hacer cambios en el entorno bien sea del hardware o software para que soporten la aplicación legada, este mantenimiento es conocido como adaptativo. A continuación, se adicionan nuevas funcionalidades al sistema que respondan a los cambios en requerimientos existentes o que se agreguen. Por lo general el mantenimiento que requiere mayor esfuerzo es el resultante de adicionar nuevas funcionalidades (65%), le siguen en una proporción similar el adaptativo (18%) y el correctivo (17%) [10]. La figura 2 describe el proceso de mantenimiento propuesto por Sommerville [10], iniciando con peticiones de cambio que hacen los interesados en el sistema (usuarios, clientes, analistas). Se evalúa el impacto que puede causar y se cuantifica el costo. Una vez se han aceptado los cambios, se planifican en el software considerando los

niveles de modificación: correctivo, adaptativo o de mejoramiento.

La siguiente tarea es implementar los cambios creando una nueva versión del producto, que finalmente es liberado.

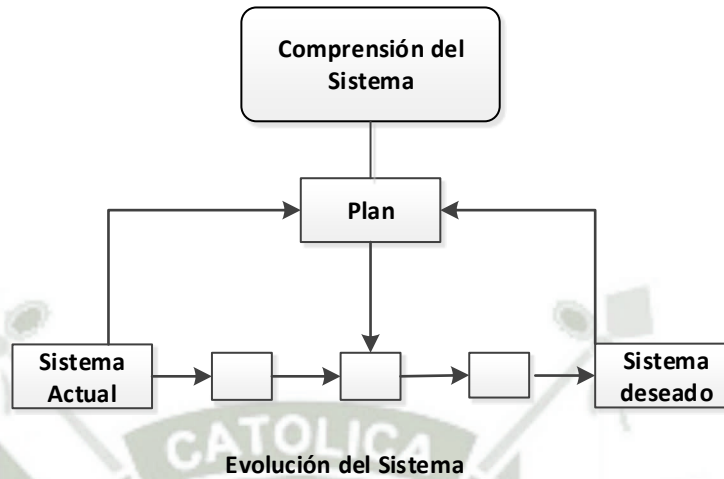


*Figura 2: Proceso de mantenimiento de software. [10]*

### c. Reingeniería del sistema actual

La reingeniería es un enfoque más amplio y drástico para evolucionar un sistema, que supone el mejoramiento de la eficiencia en el uso de los recursos disponibles (hardware y software) adicionando nuevas funcionalidades al sistema legado y reduciendo drásticamente los costos de mantenimiento. Este enfoque pretende actualizar y reestructurar los sistemas legados para garantizar mantenibilidad a largo plazo e integración con nuevas tecnologías de software y hardware, mejorando la mayor parte de elementos algorítmicos complejos. [13] Se debe considerar que no siempre es conveniente aplicar reingeniería a un sistema de software, antes se debe determinar verdadero valor de negocio que aporta en ese momento al sistema [12]. Entre las ventajas de hacer reingeniería en los sistemas legados están bajo riesgo que supone partir de una base sólida y funcional de software, y

garantizando disminución de costos. Veamos los diferentes enfoques que orientan la reingeniería. [13]



*Figura 3: Propuesta de reingeniería como solución de un problema delimitado.*

### 2.2.2. Reingeniería de Software

Reingeniería del software se puede definir como: “modificación de un producto software, o de ciertos componentes, usando para el análisis del sistema existente técnicas de Ingeniería Inversa y, para la etapa de reconstrucción, herramientas de Ingeniería Directa, de tal manera que se oriente este cambio hacia mayores niveles de facilidad en cuanto a mantenimiento, reutilización, comprensión o evaluación.” [14]

Nosotros estamos en un mundo en constante cambio donde las demandas de funciones de negocios y de tecnología de información que las soportan están cambiando a un ritmo que impone mucha presión competitiva en todas las organizaciones comerciales. Tanto los negocios como el software que soportan, deberán diseñarse una vez más para mantener el ritmo. [26]

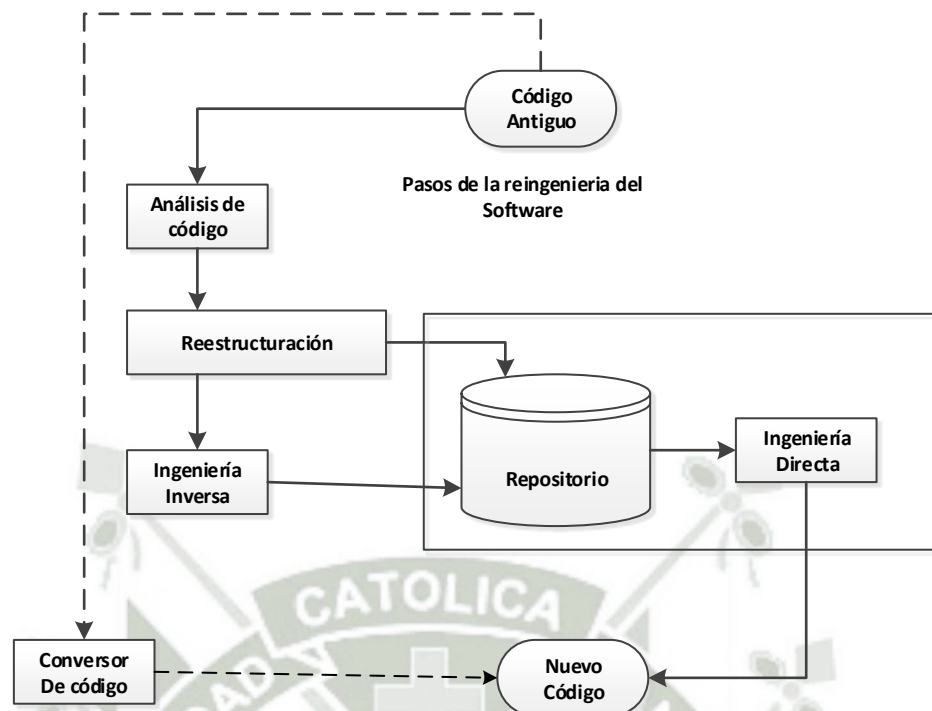
Entre los beneficios de aplicar reingeniería a un producto existente se puede incluir:

- Pueden reducir los riesgos evolutivos de una organización.
- Puede ayudar a las organizaciones a recuperar sus inversiones en software.
- Puede hacer el software más fácilmente modificable
- Amplía las capacidades de las herramientas CASE
- Es un catalizador para la automatización del mantenimiento del software
- Puede actuar como catalizador para la aplicación de técnicas de inteligencia artificial para resolver problemas de reingeniería

La reingeniería del software involucra diferentes actividades como son:

- Análisis de inventarios
- Reestructuración de documentos
- Ingeniería inversa
- Reestructuración de programas y datos
- Ingeniería directa

con la finalidad de crear versiones de programas ya existentes que sean de mejor calidad y los mismos tengan una mayor facilidad de mantenimiento.



**Figura 4: Pasos de la Reingeniería de Software**

**a. Análisis de Inventario**

Todas las organizaciones de *software* deberían tener un inventario de todas sus aplicaciones. El inventario tal vez no sea más que un modelo en una hoja de cálculo que contenga información que proporcione una descripción detallada (tamaño, edad, importancia para el negocio) de las aplicaciones activas.

Los candidatos a la reingeniería aparecen cuando se ordena esta información en función de su importancia para el negocio, longevidad, mantenibilidad actual y otros criterios localmente importantes. Es entonces cuando es posible asignar recursos a las aplicaciones candidatas para el trabajo de reingeniería.

Es importante señalar que el inventario deberá visitarse con regularidad, el estado de las aplicaciones puede cambiar en función del tiempo y, como resultado, cambiarán las prioridades para la reingeniería. [14]

#### **b. Reestructuración de Documentos**

La documentación débil es la marca de muchos sistemas heredados. ¿Pero que se hace acerca de ellos? ¿Cuáles son las opciones? Crear documentación consume mucho tiempo, si el sistema funciona vivirá con lo que tenga. La documentación debe actualizarse, pero se tiene recursos limitados. Se utiliza un enfoque de “documentar cuando se toque”. El sistema es crucial para el negocio y debe volver a documentarse por completo incluso en este caso un enfoque inteligente es recortar la documentación a un mínimo esencial. Cada una de estas opciones es viable. Una organización de software debe elegir la más apropiada para cada caso. [14]

#### **c. Ingeniería Inversa**

La ingeniería inversa es el proceso de descubrir los principios tecnológicos de un objeto, herramienta, dispositivo o sistema, mediante el razonamiento abductivo (haciendo conjeturas) de su estructura, función y operación. Dicho de otra manera, se trata de tomar algo, por ejemplo, un dispositivo mecánico o electrónico, para analizar su funcionamiento en detalle, con el objetivo de crear un dispositivo que haga la misma tarea o una similar sin copiar los detalles del original. [30]

#### d. Reestructuración de Código

La Reestructuración del Código es una Técnica de Ingeniería de Software y principalmente modifica el código fuente y/o los datos en un intento de adecuarlos a futuros cambios. Actualmente el término Reestructuración de Código se vincula mejor como Refactorización (del inglés *Refactoring* que propone la *extreme Programming*) lo cual implica, realizar modificaciones en el código, con el objetivo de mejorar su estructura interna sin alterar su comportamiento externo y es el que manejaremos dentro de este texto. Refactorizar no es una técnica para encontrar y corregir errores en una aplicación, de hecho, no modificar el comportamiento externo de la aplicación es uno de los pilares de cualquiera de las prácticas que forman parte de la técnica, por lo que es indispensable hacer uso de pruebas unitarias para asegurarse de no haber realizado alguna modificación. El término se creó como analogía con la factorización de números y polinomios. [31]

#### e. Reestructuración de Datos

La reestructuración de datos es una actividad de reingeniería a gran escala. En la mayoría de los casos, la reestructuración de datos comienza con una actividad de ingeniería inversa. La arquitectura de datos actual se analiza con minuciosidad y se define los modelos de datos necesarios, se identifican los objetivos de datos y los atributos, y después se revisa la calidad de las estructuras de datos existentes.

Cuando la estructura de datos es débil (por ejemplo, actualmente se implementan archivos planos, cuando un enfoque relacional simplificaría muchísimo el procesamiento), se aplica una reingeniería a los datos.

Dado que la arquitectura de datos tiene una gran influencia sobre la arquitectura del programa, y también sobre los algoritmos que lo pueblan, los cambios en datos darán lugar invariablemente a cambios o bien de arquitectura o bien de código. [14]

#### f. Reingeniería Directa

En un mundo ideal, las aplicaciones se reconstruyen utilizando un “motor de reingeniería” automatizado. En el motor se insertaría el programa viejo, que lo analizaría, reestructuraría y después regeneraría la forma de exhibir los mejores aspectos de la calidad del *software*. La ingeniería directa no solo recupera la información de diseño a partir del *software* existente, también utiliza esta información para alterar o reconstruir el sistema existente con la finalidad de mejorar su calidad global. En la mayoría de los casos el software sometido a reingeniería vuelve a implementar la función del sistema existente y también añade nuevas funciones o mejoras. [14]

#### ¿Cuándo es necesaria la reingeniería?

Los candidatos a la reingeniería aparecen usualmente si cumplen estas condiciones [25]:

- Frecuentes fallas de producción (fiabilidad cuestionable).
- Problemas de rendimiento.

- Tecnología obsoleta.
- Problemas de integración del sistema.
- Código de calidad pobre.
- Dificultad (peligroso) al cambio.
- Dificultad para probar.
- Mantenimiento caro.
- Incremento de problemas del sistema.

### 2.2.3. Reingeniería de Procesos

La Reingeniería de Procesos es una de las herramientas de gestión más recientes que surge a finales de la década de los 80 de la mano de los autores Michael Hammer y James Champy. Debe entenderse como un estímulo al cambio de las realidades empresariales. Pretende proporcionar soluciones que permitan a las organizaciones enfrentarse a los retos que exigen los clientes, al obstáculo que representa la competencia y, por último, al riesgo que supone un importante cambio en la empresa. Como partes fundamentales de la implementación de esta metodología citamos las personas, las actividades y las técnicas y herramientas.

[15]

#### a. Conceptos de Reingeniería

La Reingeniería es el rediseño de un proceso en un negocio o un cambio drástico de un proceso, es comenzar de nuevo desde cero, es un cambio de todo o nada. Pero no sólo esto, la reingeniería también significa el abandono de viejos procedimientos, y la búsqueda de trabajo que agregue

valor al consumidor, así como organizar la empresa alrededor de los procesos. [15]

La reingeniería analiza el proceso, lo observa completo desde que se reciben los primeros insumos hasta una salida que, por principio básico, debe tener valor para el cliente. Aunque las tareas individuales son importantes para el desarrollo del proceso completo, ninguna de ellas es importante si el proceso global no funciona; ninguna de ellas importa si el cliente no recibe el producto final esperado en tiempo y forma. [27]

La reingeniería de un proceso cuenta con la dificultad de que el proceso que se quiere cambiar ya existe, y este hecho complica su propia transformación porque se inclina a basarse en gran medida al proceso inicial. Para que la reingeniería funcione debe empezar sin ningún preconcepto, sin dar nada por sentado, debe determinar primero lo que una empresa debe hacer y después cómo hacerlo. La reingeniería debe olvidarse por completo de lo que es el proceso actual y concentrarse en cómo debe ser, es decir, reinventar como podemos hacer de nuevo nuestro trabajo. El rediseño radical implica ir a la raíz de las cosas. Por tanto, también es un aliciente para el personal, por el aspecto creativo e innovador que introduce en las tareas de un profesional dedicado al diseño de procesos. Esta herramienta tiene en su planteamiento lograr que se produzca una vinculación entre el cambio que promueve, el enfrentamiento con la competencia y la búsqueda de la satisfacción de los clientes. [15] En la figura siguiente se trata de mostrar el grado de compromiso de la reingeniería con los agentes esenciales del éxito:



*Figura 5: Las Tres C. [15]*

### **b. Metodología de implementación de Reingeniería**

Las empresas suelen plantearse la reingeniería de procesos en muchas ocasiones demasiado tarde, lo que implica que la mejora no sea lo suficientemente efectiva, además conduce a que las personas se acostumbren a una manera particular de ver el proceso y sean más reticentes a los cambios. La implantación de una metodología que indique la manera de actuación, y además involucre y prepare a toda la organización a visualizar los aspectos y procesos mejorables, es la manera de resolver los problemas que suceden en empresas que requieren reingeniería en sus procesos, y que sin ella no están preparados para incorporarla rápida y eficazmente a sus procesos. En la siguiente figura se muestran la estructura de la metodología, desde los niveles iniciales hasta fusionar todas las condiciones necesarias: herramientas, actividades y personas, de menor a mayor dificultad, poniendo de relieve que la mayor complejidad y se encuentra en las personas donde radica el éxito, ya que son las que conducen la reingeniería. El conjunto de herramientas o actividades no obtienen ninguna ventaja competitiva. Si no se realiza

desde la concienciación y motivación de las personas que lo que hacen.

[15]



*Figura 6: Tres niveles estructurales. [15]*

En relación a las herramientas de Reingeniería, el objetivo del cambio es mejorar y la mejora se debe emprender sobre una base de conocimiento del estatus actual del proceso. Se parte de una evaluación utilizando una serie de medidas relevantes para el proceso, siendo éste el planteamiento de mejora, el cual debe ser ambicioso, pero a su vez posible y controlado. Para llevar dicho control se establece un conjunto de indicadores que actuarán como medidores de la evolución del cambio. [15]

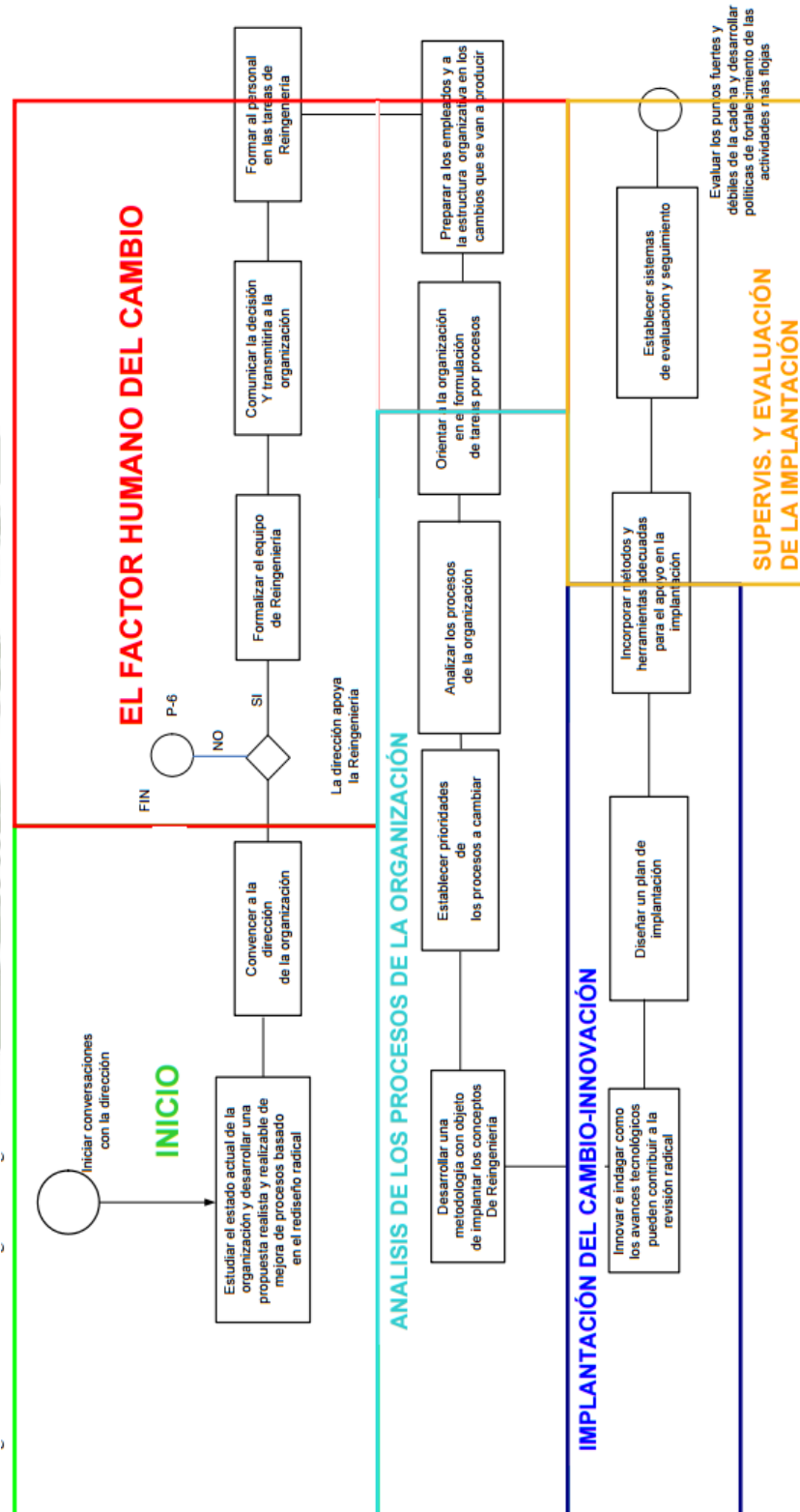


Figura 7: Proceso metodológico de reingeniería. [15]

### c. Etapas de la Reingeniería

Las etapas por las que evoluciona la metodología de reingeniería están desglosadas siguiendo un circuito progresivo, tal y como muestra la figura, a medida del cual la organización avanza para completar un conjunto de buenas prácticas en lo que se refiere a la mentalización y organización empresarial, preparada a evolucionar con los cambios que ella misma promueve de manera programática. [15]



*Figura 8: Evolución de la reingeniería. [15]*

#### **Arranque de la implantación**

Solo desde el convencimiento, el conocimiento exhaustivo y la aceptación de las implicaciones que tiene para la organización, pueden dar comienzo los planes de implantación que supongan el rediseño fundamental de los procesos susceptibles de ser cambiados. La reingeniería supondrá para las organizaciones grandes y profundos cambios en la estructura y en el personal de la compañía. Suele convenir analizar la repercusión del cambio y como está preparada, o las necesidades de preparar a la organización para que el personal se sienta cómodo en las nuevas funciones que le asigne la reingeniería. Para ello, se deben iniciar conversaciones, Estudiar el estado actual de la organización con respecto a la reingeniería y Plantear un plan-calendario de actuaciones. [15]

### **El factor humano del cambio**

La preparación del personal a afrontar los cambios de manera optimista es decisiva para avanzar siguiendo la metodología propuesta, siendo habitual que ésta sea la etapa sobre la que recaen los riesgos más determinantes, y propulsores del fracaso de la implantación de cualquier proyecto de reingeniería, y mucho más de su metodología. Dentro de esta etapa se tiene que Formalizar un equipo de trabajo, Comunicar e involucrar a toda la organización, Formar al personal en las tareas de Reingeniería y Preparar a los empleados y a la estructura organizativa en los cambios que se van a producir. [15]

### **Análisis de los procesos**

En esta fase se da comienzo y se realizan los preparativos que permiten contemplar el estado de los procesos. Para ello se describen y desarrollan los planes de actuación en cuanto a los análisis cualitativos y cuantitativos del estado actual su evolución y su relevancia en la organización. Además, se desarrolla la metodología donde se describe las pautas de evaluación y supervisión de los procesos los diferentes estadios por los que pasa un proceso desde su implantación y cuando es necesario su relevo. La metodología incorpora las técnicas y herramientas y la descripción de las áreas involucradas y los recursos y cómo se realiza la incorporación de éstos en la implantación de un cambio radical. Lógicamente estas circunstancias tratadas en la metodología deben ir evolucionando de la misma manera en que lo hacen los procesos. Las

etapas para el Análisis de los procesos son: Establecer prioridades de los procesos a cambiar, Analizar los procesos de la organización, Orientar a la Análisis de los procesos de la organización Arranque Implantación del cambio Factor humano del cambio. Supervisión y evaluación de la implantación 6 organización en la formulación de tareas por procesos y Desarrollar una metodología con objeto de implantar los conceptos de Reingeniería. [15]

### **Implantación del cambio / innovación**

En esta fase se entra de lleno en la parte más operativa del cambio se evalúan los procesos de la organización y se realizan los preparativos necesarios para desarrollar el cambio. Por ello es en este apartado donde prestamos especial atención en el trabajo, debido a que es la esencia de este, aunque sin las otras etapas no podría desarrollarse ésta. La implantación del cambio-innovación se realiza de la siguiente manera: Innovar e indagar como los avances tecnológicos pueden contribuir a la revisión radical, Diseñar un plan de implantación e Incorporar métodos y herramientas adecuadas para el apoyo en la implantación. [15]

### **Supervisión y evaluación de la implantación**

Se estructura en Establecer sistemas de evaluación y seguimiento, ser consciente de que el cambio es dinámico y Comentar y transmitir los logros alcanzados y la necesidad de continuar. [15]

La Reingeniería es una solución muy radical que no puede confundirse con ninguna otra, ya que implica la reinvención de los procesos y no su

mejora o reestructuración. Esto puede llegar a ser una gran ventaja competitiva para las empresas que buscan mejorar sus procesos de negocios.

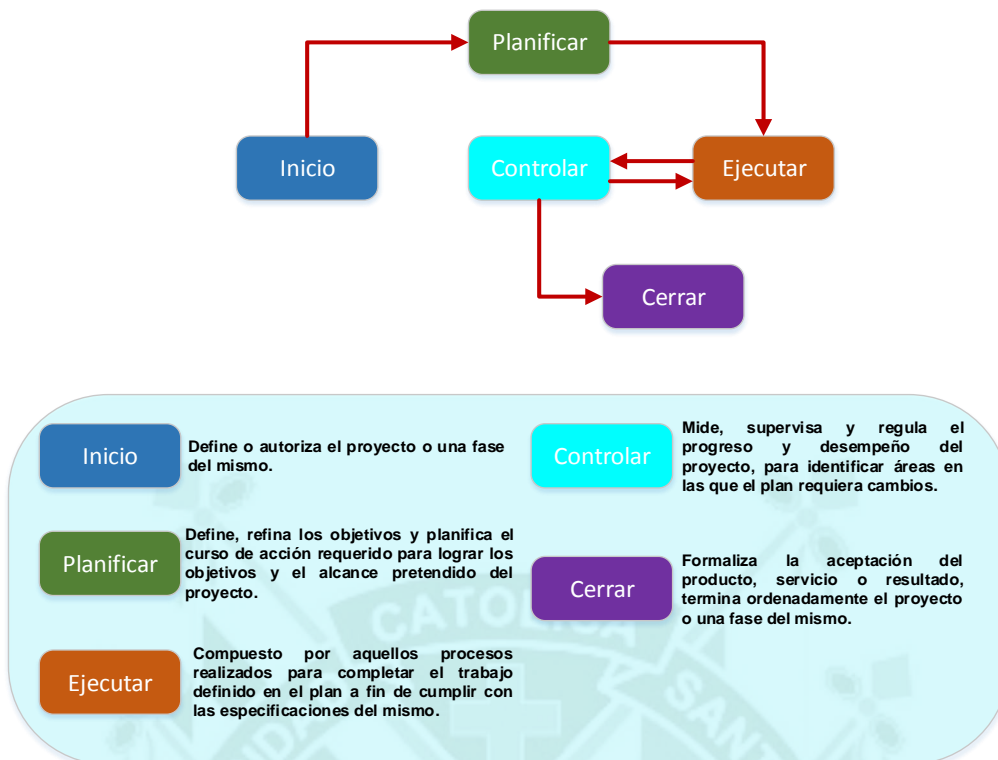
#### 2.2.4. Pmbok

El PMBOK® GUIDE es un estándar en la Administración de proyectos desarrollado por el *Project Management Institute* (PMI). La misma comprende dos grandes secciones, la primera sobre los procesos y contextos de un proyecto, la segunda sobre las áreas de conocimiento específico para la gestión de un proyecto. [16]

Este estándar reconoce 5 grupos de procesos básicos y 10 áreas de conocimiento comunes a casi todos los proyectos. Los procesos se traslapan e interactúan a través de un proyecto o fase y son descritos en términos de:

- Entradas (documentos, planes, diseños, etc.)
- Herramientas y Técnicas (mecanismos aplicados a las entradas)
- Salidas (documentos, productos, etc.).

##### a. Grupo Básicos de Procesos



*Figura 9: Grupos básicos de proceso*

## b. Áreas de Conocimiento

Las 10 áreas del conocimiento mencionadas en el PMBOK® GUIDE son [16]:

**1. Gestión de la Integración del Proyecto:** Incluye los procesos y actividades necesarios para identificar, definir, combinar, unificar y coordinar los diversos procesos y actividades de la dirección de proyectos dentro de los grupos de procesos de dirección de proyectos.

**2. Gestión del Alcance del Proyecto:** Incluye los procesos necesarios para garantizar que el proyecto incluya todo (y únicamente todo) el trabajo requerido para completarla con éxito.

**3. Gestión del Tiempo del Proyecto:** Incluye los procesos requeridos para administrar la finalización del proyecto a tiempo.

**4. Gestión de los Costos del Proyecto:** Incluye los procesos involucrados en estimar, presupuestar y controlar los costos de modo que se complete el proyecto dentro del presupuesto aprobado.

**5. Gestión de la Calidad del Proyecto:** Incluye los procesos y actividades de la organización ejecutante que determinan responsabilidades, objetivos y políticas de calidad a fin de que el proyecto satisfaga las necesidades por la cuales fue emprendido.

**6. Gestión de los Recursos Humanos del Proyecto:** Incluye los procesos que organizan, gestionan y conducen el equipo del proyecto.

**7. Gestión de las Comunicaciones del Proyecto:** Incluye los procesos requeridos para garantizar que la generación, la recopilación, la distribución, el almacenamiento, la recuperación y la disposición final de la información del proyecto sean adecuados, oportunos y entregada a quien corresponda (interesados del proyecto o *stakeholders*).

**8. Gestión de los Riesgos del Proyecto:** Incluye los procesos relacionados con llevar a cabo la planificación de la gestión, identificación, el análisis, la planificación de respuesta a los riesgos, así como su monitoreo y control en un proyecto.

**9. Gestión de las Adquisiciones del Proyecto:** Incluye los procesos de compra o adquisición de los productos, servicios o resultados que es necesario obtener fuera del equipo del proyecto.

**10. Gestión de Interesados del Proyecto:** Incluye los procesos relacionados con la identificación, análisis y manejo proactivo de los interesados.

**c. Estimación de proyectos**

Actualmente se dispone de técnicas para estimación de proyectos que permiten la realización de evaluaciones más precisas que las obtenidas a través de métodos tradicionales (orientadas a calcular individualmente el esfuerzo correspondiente a cada una de las actividades del mismo). [17]

En la mayoría de las empresas donde se produce software para apoyar el negocio, las prácticas de estimación y planificación son débiles. En general, los administradores estiman el costo y la duración del proyecto a desarrollar utilizando solamente el juicio de un experto, lo que produce cronogramas y presupuestos poco acertados.

**1. Método tradicional**

Las distintas metodologías tradicionales de estimación de *Boehm B.W., Software Engineering Economics*, proporcionan un dato básico (horas hombre) a aplicar al proyecto, entendiéndose por ésta a cantidad de horas a utilizar, a partir de lo cual y mediante la utilización de diversos ratios tales como pesos/hora u horas/persona pueden deducirse otros valores de estimación como el costo, personal involucrado, etc. Basándonos en la experiencia recogida en la actividad profesional (ITBA-CAPIS. Carpetas de la Carrera de Posgrado en Ingeniería del Software) se plantean básicamente dos situaciones (que son

ejemplificadas en la figura 10, en la cual se muestra el proceso de estimación de acuerdo al enfoque tradicional que determina las horas hombre necesarias a partir de las que se calculan los costos y en función de los plazos de entrega acordados, se llega finalmente a establecer la cantidad de personal a utilizar). [17]

**Primera Situación:** Tiene que ver con el costo estimado del proyecto (a partir de estimar horas hombre) el cual está directamente relacionado al ratio utilizado y considerando los plazos de entrega se converge a un valor que satisfaga a las partes involucradas [17]



**Figura 10: Enfoque tradicional para estimación de proyectos. [17]**

**Segunda Situación:** Está íntimamente relacionado con la estimación del tiempo total del proyecto ya que es determinado por dos criterios que normalmente se oponen según nos refiramos al punto de vista del usuario o del desarrollador. Para explicar mejor la idea y simplificándola podemos sintetizar que el desarrollador tratará de extender el proyecto lo máximo posible a fin de asegurar su cumplimiento y el usuario, por el contrario, pretenderá reducirlo. Dado que los métodos tradicionales proporcionan una medida de las horas hombre necesarias para aplicar al proyecto, la discusión se centra en la asignación de personal de manera

de acortar proporcionalmente el tiempo total en función de aumentar aquella variable. Si bien para los desarrolladores es claro que ello no es así (normalmente se recurre a analogías del estilo de “... si una persona puede pintar una habitación en diez días no puede pensarse que diez personas pudieran hacerlo en un día ...”), éstos no disponen de argumentos con base científica que puedan explicar esta situación cuando se trata de proyectos de desarrollo de *software*. Esto lleva invariablemente a la utilización de explicaciones que tienen más que ver con el sentido común y la experiencia que con justificaciones elaboradas con fundamento. Dicha situación acarrea un inconveniente adicional producido por el establecimiento de fechas de entrega que pueden resultar imposibles de cumplir cuando se acuerdan bajo la presión del usuario o bien por la necesidad del desarrollador de cumplir anticipadamente un proyecto. Como síntesis de lo expresado sería deseable utilizar técnicas que no solamente posibilitaran calcular las horas hombre a aplicar al desarrollo de un proyecto sino estimar horas hombre estimar el costo determinar plazos de entrega determinar el personal involucrado también estimar un valor de la duración del mismo dado por sus características intrínsecas independientemente de los recursos a emplear. [17]

## 2. Métodos Alternativos

La utilización de dos métodos, Puntos de Función (Medición del tamaño de software) y COCOMO (Modelo constructivo de costos) en forma conjunta permitiría mejorar la situación descrita en la sección

precedente. El método COCOMO permite determinar los valores de las siguientes dos variables:

- Meses/hombre a aplicar al proyecto
- Meses totales del proyecto (dependiendo de factores tales como los atributos de fiabilidad requerida del *software*, tamaño de la base de datos, complejidad del producto, limitaciones en el tiempo de ejecución, limitaciones de memoria principal, volatilidad de la máquina virtual, frecuencia de cambio en el modelo de explotación del ordenador, capacitación de los analistas, experiencia en aplicaciones, capacitación de los programadores, experiencia en la máquina virtual, experiencia en el lenguaje de programación, prácticas modernas de programación, uso de herramientas para el desarrollo del software y limitaciones en la planificación). [17]

En la figura 11 se presenta un esquema de estimación que proporciona además de las horas hombre a emplear el tiempo total del proyecto (basándose para ello en el conocimiento previo de la cantidad de sentencias de código del proyecto) lo que permite determinar el plazo de entrega. Mostrándose, además, como a partir de estos dos valores (horas hombre y tiempo total) y simplemente por el cociente de ambos se obtiene la cantidad de recursos (personas) para llevarlo a cabo. A partir de allí se puede elaborar el costo mediante la aplicación de ratios, de igual forma que en las metodologías tradicionales. [17]



*Figura 11: Estimación de proyectos por método COCOMO [17]*

Como pudimos apreciar hay dos factores fundamentales a examinar en la estimación de un proyecto software, su duración y costo, la información que toma cada vez mayor importancia como factor estratégico ha determinado que sea la duración de un proyecto uno de los aspectos más prioritarios en su realización.

### 2.2.5. Juicio de Expertos

Una pregunta que surge cuando se intenta medir el comportamiento es qué tan válida y confiable es la medición. El presente escrito se centra en la validez de contenido, específicamente, en la utilización del juicio de expertos como parte del proceso para su estimación. Esta técnica debe realizarse de manera adecuada, ya que muchas veces constituye el único indicador de la validez de contenido. Actualmente el juicio de expertos es una práctica generalizada que requiere interpretar y aplicar sus resultados de manera acertada, eficiente y con toda la rigurosidad metodológica y estadística, para permitir que la evaluación basada en la información obtenida de la prueba pueda ser utilizada con los propósitos para la cual fue diseñada. [18]

Estos métodos se utilizan más cuando no existe un conjunto de datos históricos útiles en los cuales pueda basarse un análisis. Los análisis cualitativos son cada vez más importantes y comienzan a formar parte de las investigaciones en múltiples esferas. Los expertos pueden ser especialistas internos o externos, pues muchas veces la práctica se convierte en un método seguro, pero cuando la búsqueda es completamente teórica ¿cómo demostrar la fidelidad de nuestra propuesta? [19]

### **2.2.6. Modelo Vista Controlador – MVC**

Modelo Vista Controlador es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. MVC consiste en tres tipos de objetos. Modelo es el objeto de la aplicación, la Vista es su representación en pantalla y el Controlador define el modo en que la interfaz reacciona a la entrada del usuario. MVC desacopla las vistas de los modelos estableciendo entre ellos un protocolo de suscripción/notificación. [20]

#### **a. Flujo de Control**

En el flujo de control podemos apreciar [20]:

- El usuario realiza una acción en la interfaz.
- El controlador trata el evento de entrada. Previamente se ha registrado.
- El controlador notifica al modelo la acción del usuario, lo que puede implicar un cambio del estado del modelo (si no es una mera consulta)

- Se genera una nueva vista. La vista toma los datos del modelo. El modelo no tiene conocimiento directo de la vista.
- La interfaz de usuario espera otra interacción, que comenzará otro nuevo ciclo.

#### **b. Framework**

Un *framework*, en el desarrollo de software, es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado.

Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otro software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. [20]

No es más que una base de programación que atiende a sus descendientes (manejado de una forma estructural y/o en cascada) posibilitando cualquier respuesta ante las necesidades de sus miembros, o en secciones de una aplicación (Web), satisfaciendo así las necesidades más comunes del programador. [20]

#### **c. Arquitectura**

Dentro de este aspecto, podemos basarnos en el modelo MVC (Controlador = Modelo = Vista), ya que debemos fragmentar nuestra programación. Tenemos que contemplar estos aspectos básicos en cuanto a la implementación de nuestro sistema. [20]

#### d. Estructura

Los datos dependen de nosotros como interpretar y manejar estos 'datos'. Ahora, sabemos que el único dato de una dirección estática web es conseguir un archivo físico y sea interpretado o no, el servidor responde. El modelo, al igual que el controlador y la vista, maneja todos los datos que se relacionen consigo (solo es el proceso medio de la separación por capas que ofrece la arquitectura MVC). Y solo la vista, puede demostrar dicha información. Con lo cual ya hemos generado jerarquía de nuestro programa: Controlador, Modelo y Vista. [20]

#### e. Lógica

Al parecer, debemos inyectar ciertos objetos dentro de sus parientes en esta aplicación solo así se compartirán herencia y coherencia en su aplicación. [20]

#### 2.2.7. Code First

Según Sergio León, nos indica que *Code First* (lo primero es el código) es un enfoque más de *Entity Framework* (hay otros dos enfoques que son *Database First* y *Model First*) que plantea lo siguiente: Tú creas N clases con tu lenguaje favorito (C#, VB.NET, etc.) y crea relaciones entre las mismas, después despreocúpate que ya veré yo como me las gasto para persistir tu modelo en una base de datos. [21]

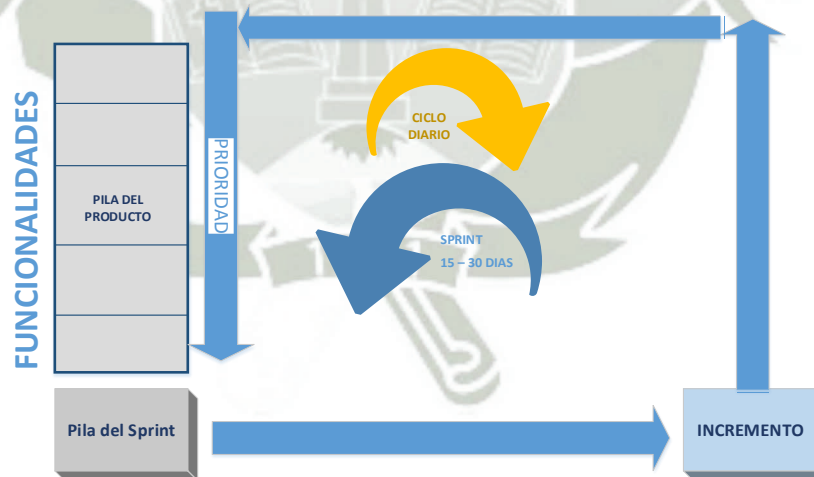
Lo importante es entender que con *Code First*, En vez de comenzar creando la base de datos y después con ingeniería inversa generar las N clases (como

hacíamos con *Database First*), con *Code First* primero creamos el modelo con código y después se genera automáticamente la base de datos. [21]

### 2.2.8. Scrum

*Scrum* es un marco de trabajo en el que equipos multifuncionales pueden crear productos o desarrollar proyectos de una forma iterativa e incremental. El desarrollo se estructura en ciclos de trabajo llamados *Sprints* (también conocidos como iteraciones). Estas iteraciones no deben durar más de cuatro semanas cada una (siendo dos semanas la duración más habitual) y tienen lugar una tras otra sin pausa entre ellas. [24]

*Scrum* es una metodología ágil de desarrollo de proyectos, se denomina “*sprint*” a cada iteración de desarrollo y recomienda realizarlas con duraciones de 30 días. El *sprint* es por tanto el núcleo central que proporciona la base de desarrollo iterativo e incremental. [23]

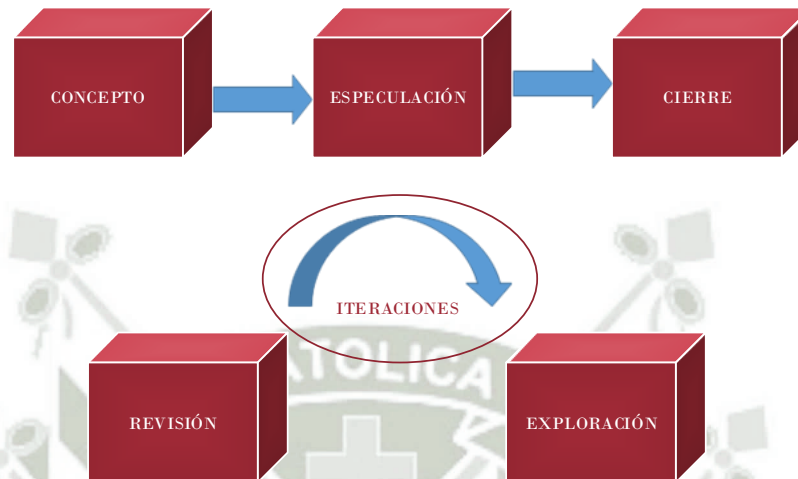


**Figura 12: Visión general del proceso [23]**

*Scrum* es una metodología ágil, y como tal [23]:

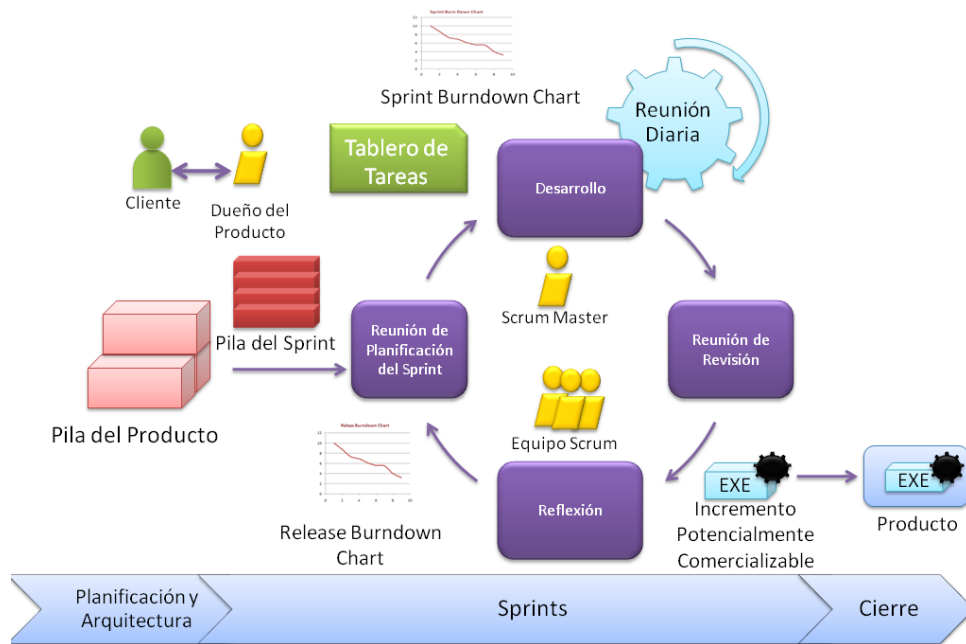
- Es un modo de desarrollo de carácter adaptable más que predictivo.

- Orientado a las personas más que a los procesos.
- Emplea la estructura de desarrollo ágil: incremental basada en iteraciones y revisiones.



**Figura 13: Estructura del desarrollo ágil – Adaptado [23]**

Todos los días el Equipo se reúne brevemente para inspeccionar su progreso y ajustar los siguientes pasos necesarios para completar el trabajo pendiente. Al final del Sprint, el Equipo revisa el Sprint con los diferentes *Stakeholders* (interesados e involucrados en el producto) y realiza una demostración de lo que han desarrollado. Se obtiene *feedback* que podrá ser incorporado en el siguiente *Sprint*. *Scrum* enfatiza un producto “funcionando” al final del *Sprint* que esté realmente “terminado”. En el caso del software, esto significa un sistema que está integrado, testado, con la documentación de usuario generada y potencialmente entregable. Los principales roles, artefactos y eventos están resumidos en Figura 14. [24]



**Figura 14: Visión general de Scrum [24]**

Nota: no existe ningún tipo de rol de jefe de proyecto en *Scrum*. Esto se debe a que no es necesario en absoluto: las responsabilidades tradicionales del jefe de proyecto han sido divididas y reasignadas entre los tres roles de *Scrum*, sobre todo entre el Equipo y el Dueño de Producto más que en el *ScrumMaster*. La práctica de *Scrum* con jefes de proyecto añadidos demuestra un desconocimiento fundamental de *Scrum*, y típicamente desemboca en conflictos de responsabilidad, confusión respecto a la autoridad y resultados por debajo de lo esperado. A veces, un (ex) jefe de proyecto puede tomar el rol de *ScrumMaster*, pero el éxito de esta iniciativa depende fuertemente de la persona y de hasta qué punto comprende bien las diferencias fundamentales entre ambos puestos, tanto en las responsabilidades del día a día como en el set mental necesario para tener éxito. Una buena forma de comprender el rol de *ScrumMaster* a conciencia y comenzar a desarrollar las habilidades clave necesarias para dicho éxito son los cursos de “*CertifiedScrumMaster*” de la *Scrum Alliance*. [24]

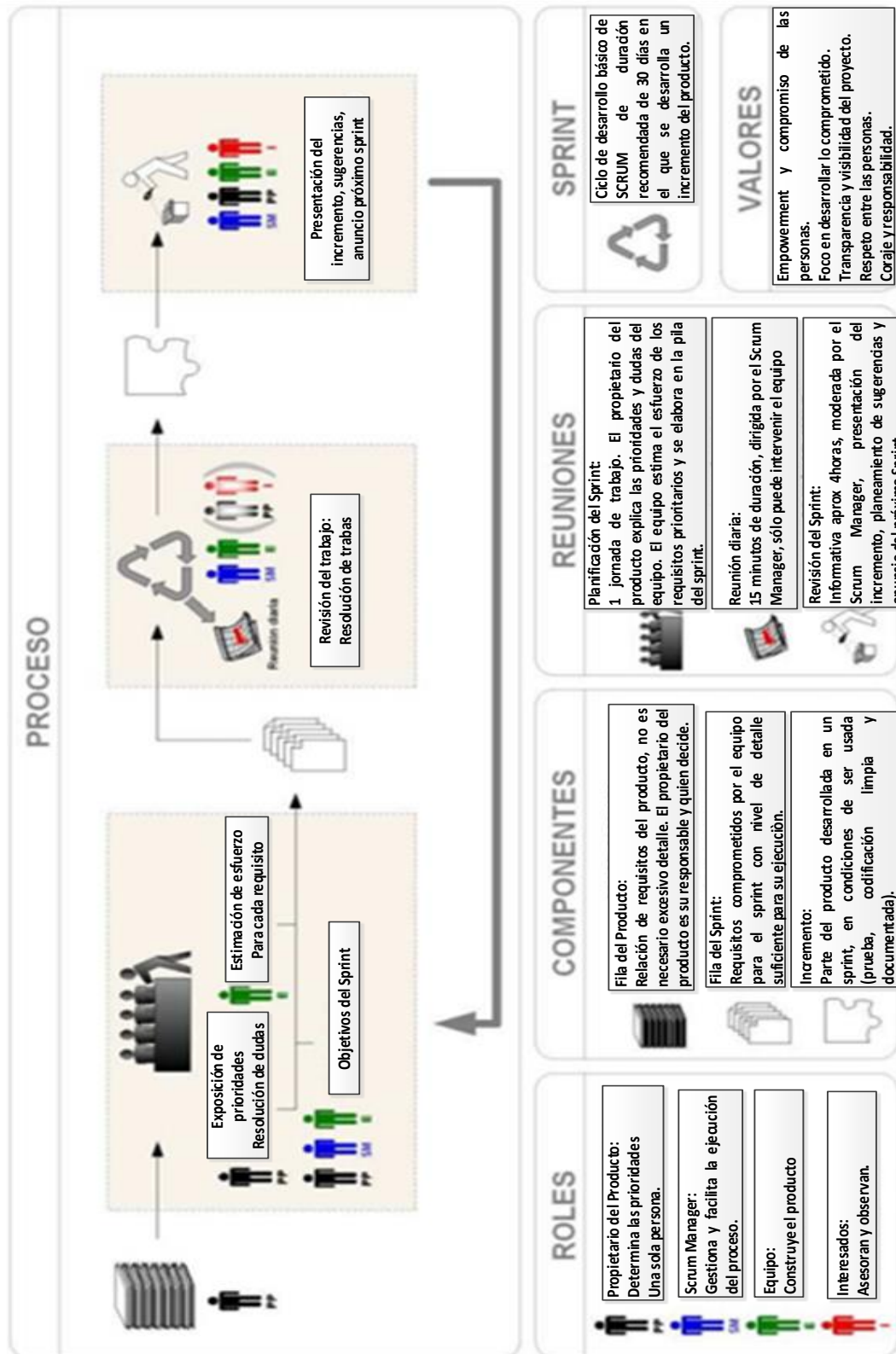


Figura 15: Visión general del modelo Scrum [23]

## CAPÍTULO III: MARCO METODOLOGICO

### 3.1. Alcances y Limitaciones

El alcance de la presente tesis va dirigida a todas las organizaciones que cuenten con un sistema de información, en donde manejen aplicativos con código fuente legado, y no se brinde el soporte adecuado ya que no se cuenta con expertos para dicho soporte.

Las limitaciones se darán principalmente a aplicativos legados que necesiten ser migrados a un entorno web, excluyendo la parte de base de datos. Las migraciones serán realizadas en tecnologías que cuenten con una licencia.

### 3.2. Aporte

La presente tesis ayudara a instituciones que cuenten con aplicativos legados, a usar tecnologías modernas y contar con las licencias debidas para recibir un soporte confiable.

### 3.3. Tipo y Nivel de la Investigación

- **Tipo de investigación:** Aplicada.
- **Nivel de investigación:** Experimental.

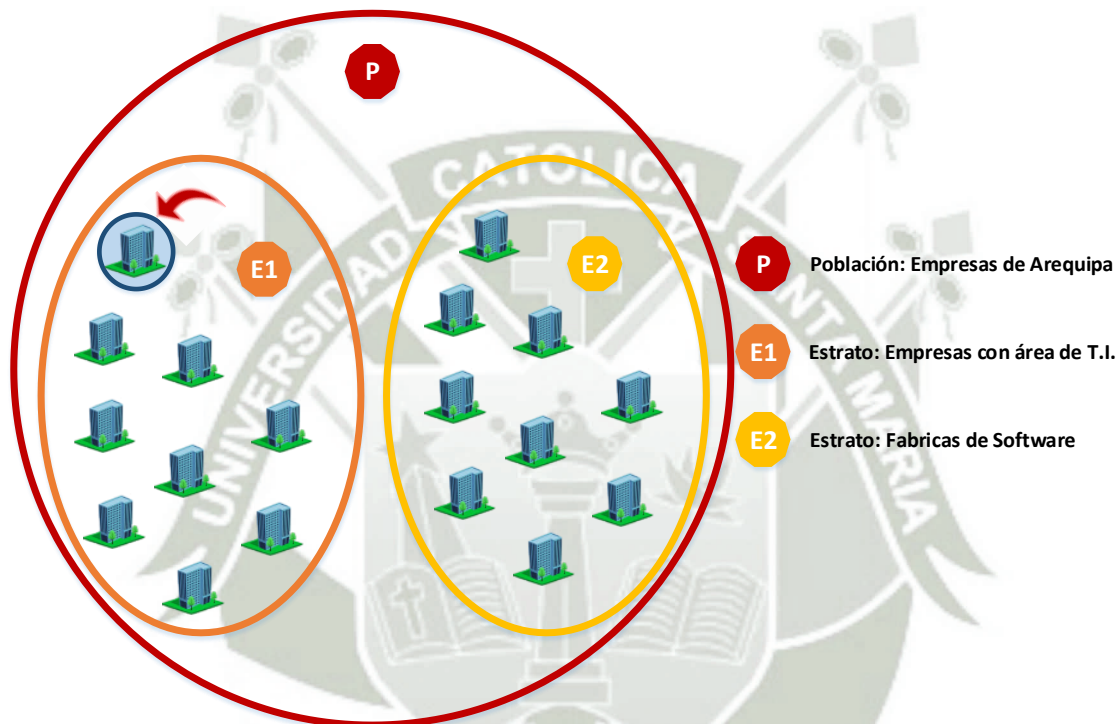
### 3.4. Población y Muestra o Universo

El proyecto está dirigido a fábricas de software, o empresas que tengan el área de tecnologías de información, de la ciudad de Arequipa, en donde cuenten con clientes que necesiten migración de aplicativos legados.

Para el cálculo de la muestra, decidimos utilizar el método no probabilístico del muestreo por cuotas, en donde nuestro universo serán las empresas del departamento de

Arequipa, dicha población se dividirá en dos estratos: Empresas que cuenten con el área de tecnologías de información y Fábricas de software.

Para este proyecto de tesis, se decidió tomar como muestra una de las empresas que se encuentra en el estrato de “Empresas con área de T.I.”, como se muestra en la siguiente figura. Se optó por esta decisión debido a que se cuenta con acceso a dicha empresa y cuentan con un aplicativo que necesita ser migrado.



*Figura 16 Población y muestra del proyecto*

### 3.5. Técnicas e Instrumentos de Recolección de Datos

Técnica	Instrumento
Observación	Lista de Chequeo Matriz de Análisis
Encuesta	Cuestionario

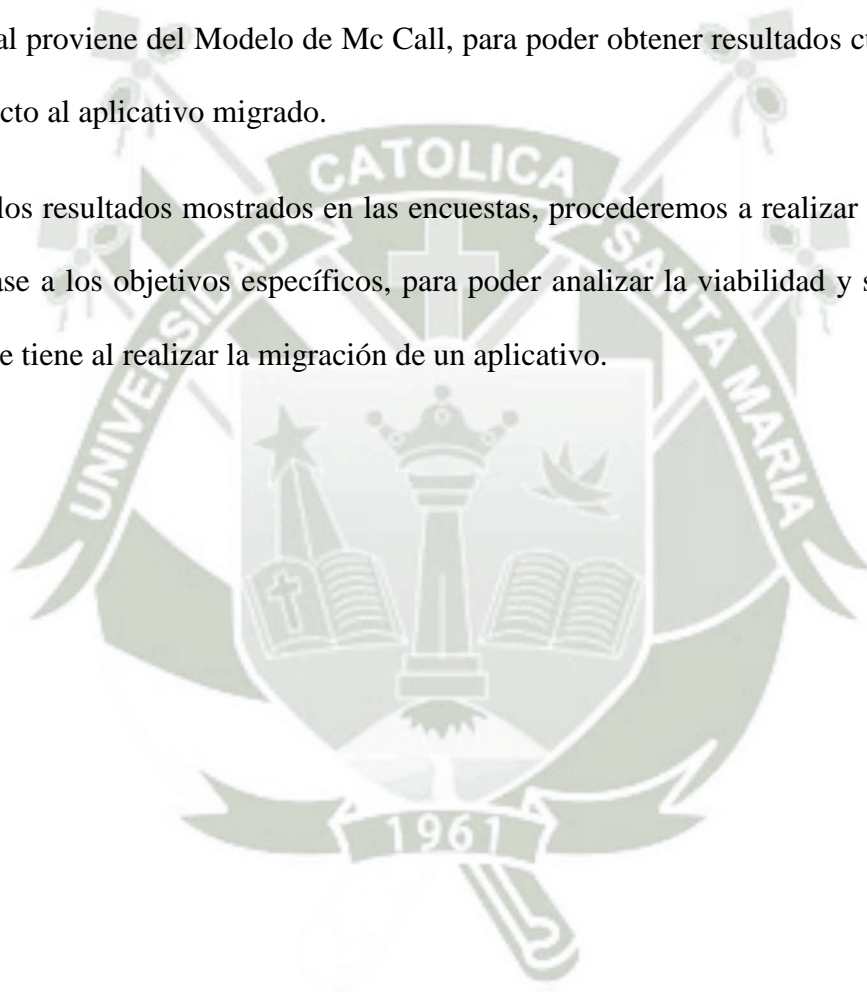
*Tabla 2: Técnica e instrumento de recolección de datos.*

### 3.6. Plan de análisis estadístico de los datos

De acuerdo a los resultados obtenidos al realizar la migración un aplicativo legado, se procederá a realizar una comparación del aplicativo migrado con el aplicativo legado, para luego proceder a realizar la lista de chequeo de acuerdo a los objetivos específicos.

Seguidamente se procederá a realizar el análisis de los resultados con el ISO 9126, el cual proviene del Modelo de Mc Call, para poder obtener resultados cuantitativos respecto al aplicativo migrado.

Con los resultados mostrados en las encuestas, procederemos a realizar estadísticas en base a los objetivos específicos, para poder analizar la viabilidad y satisfacción que se tiene al realizar la migración de un aplicativo.



## CAPÍTULO IV: PLAN DE TRABAJO

### 4.1. Proyecto de Desarrollo de Tesis

Nombre de Tarea	Duración	Fecha Inicio	Fecha Fin	% Completado
<b>Proyecto de Tesis</b>	<b>123.5 días</b>	<b>19/11/2015</b>	<b>18/05/2016</b>	<b>100%</b>
<b>Plan de Tesis</b>	<b>12 días</b>	<b>19/11/2015</b>	<b>04/12/2015</b>	<b>100%</b>
Desarrollo	7 días	19/11/2015	27/11/2015	100%
Presentación	2 días	30/11/2015	01/12/2015	100%
Correcciones	2 días	02/12/2015	03/12/2015	100%
Aprobación	1 día	04/12/2015	04/12/2015	100%
<b>Desarrollo de Tesis</b>	<b>104.5 días</b>	<b>07/12/2015</b>	<b>09/05/2016</b>	<b>100%</b>
Planteamiento de la Investigación	4 días	07/12/2015	10/12/2015	100%
<b>Fundamentos Teóricos</b>	<b>4 días</b>	<b>11/12/2015</b>	<b>16/12/2015</b>	<b>100%</b>
Estado del Arte	2 días	11/12/2015	14/12/2015	100%
Bases Teóricas	2 días	15/12/2015	16/12/2015	100%
Marco Metodológico	1 día	17/12/2015	17/12/2015	100%
Plan de Trabajo	0.5 días	18/12/2015	18/12/2015	100%
<b>Propuesta Metodológica</b>	<b>54 días</b>	<b>18/12/2015</b>	<b>09/03/2016</b>	<b>100%</b>
Planificación de Migración	8 días	18/12/2015	05/01/2016	100%

Análisis y Diseño	14 días	05/01/2016	25/01/2016	100%
Construcción	12 días	25/01/2016	10/02/2016	100%
Certificación	12 días	10/02/2016	26/02/2016	100%
Implementación	8 días	26/02/2016	09/03/2016	100%
Prueba de Metodología	25 días	09/03/2016	15/04/2016	100%
<b>Análisis y Discusión de Resultados</b>	9 días	15/04/2016	28/04/2016	100%
Evaluación de Metodología	4 días	15/04/2016	21/04/2016	100%
Evaluación de Migración	5 días	21/04/2016	28/04/2016	100%
Anexo	5 días	28/04/2016	05/05/2016	100%
Conclusión y Recomendaciones	1 día	05/05/2016	06/05/2016	100%
Introducción y Resumen	1 día	06/05/2016	09/05/2016	100%
Correcciones y otros	7 días	09/05/2016	18/05/2016	100%

*Tabla 3: Proyecto de Desarrollo de la Presente Tesis*

## 4.2. Diagrama de Gantt del Desarrollo de Tesis

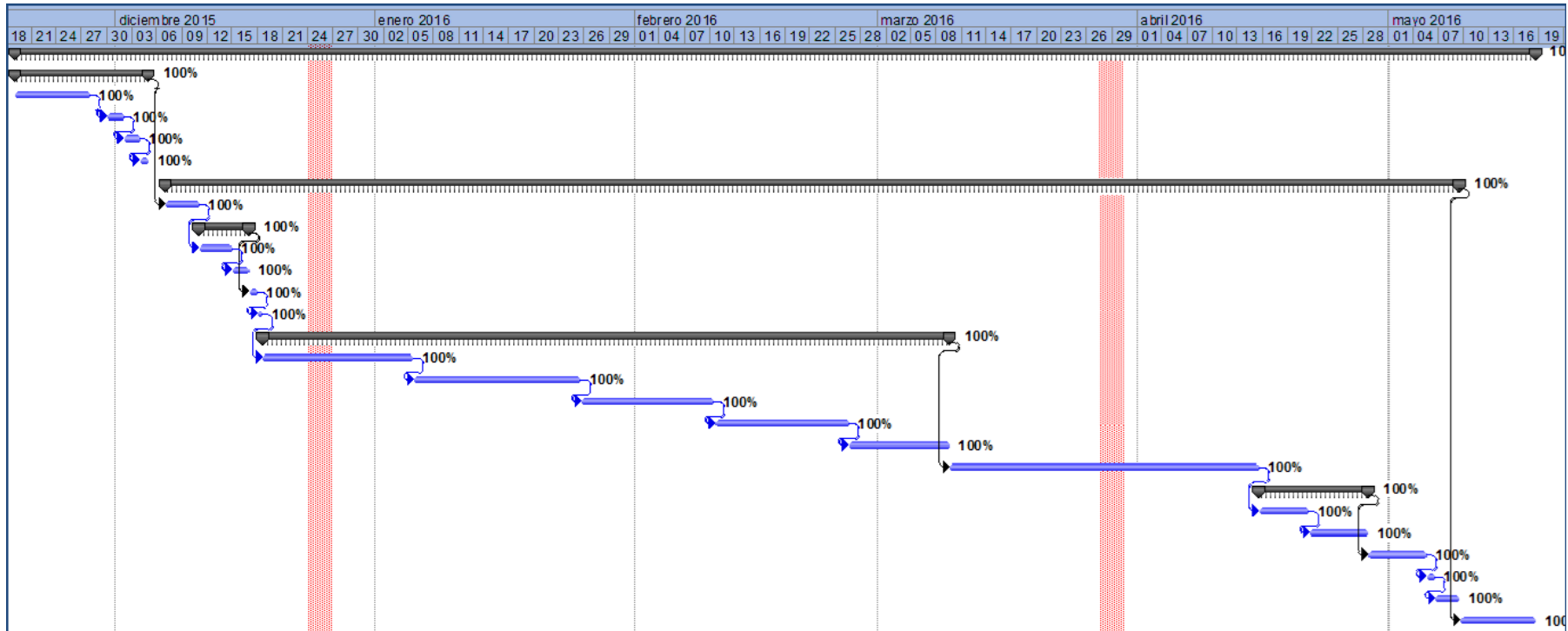


Figura 17: Diagrama de Gantt de la Presente Tesis

## CAPÍTULO V: PROPUESTA METODOLÓGICA

La metodología que se empleará en la presente tesis es la siguiente:

- **Fase de planificación:** Se trata de establecer el equipo de trabajo que deberá satisfacer las restricciones de prestaciones, planificación temporal.  
Una planificación detallada dará consistencia al proyecto y evitará sorpresas que generalmente no son bien recibidas.
- **Análisis:** Se extraen los requisitos del aplicativo legado. En esta etapa la habilidad y experiencia en la migración de aplicativos es crítica para reconocer requisitos incompletos, ambiguos o contradictorios. Usualmente el cliente tiene una visión incompleta/inexacta de lo que necesita y es necesario ayudarlo para obtener la visión completa de los requerimientos. El contenido de comunicación en esta etapa es muy intenso ya que el objetivo es eliminar la ambigüedad en la medida de lo posible.
- **Construcción:** Se traduce el análisis a código. Es la parte más obvia del trabajo de migraciones y la primera en que se obtienen resultados tangibles. No necesariamente es la etapa más larga, aunque una especificación o diseño incompletos/ambiguos pueden exigir que, tareas propias de las etapas anteriores se tengan que realizarse en esta.
- **Certificación:** Consiste en comprobar que la migración responda/realice correctamente las tareas indicadas en la especificación. Es una buena praxis realizar pruebas a distintos niveles (por ejemplo, primero a nivel unitario y después de forma integrada de cada componente) y por equipos diferenciados del de desarrollo, en este caso el validador de contenidos (pruebas cruzadas entre los programadores o realizadas por un área de test independiente).

- **Implementación:** Última fase de la metodología, donde se pondrá en marcha el aplicativo migrado en el ambiente de producción, ingresando primero a una etapa de plan piloto para luego concluir con la migración.



*Figura 18: Metodología General de la Migración*

### 5.1. Planificación Inicial de la Migración

Esta es la primera fase de la metodología planteada para la presente tesis.



*Figura 19: Metodología de la migración - Planificación*

Para realizar la migración de aplicativos legados, se sugiere comenzar por dar respuesta a un grupo muy concreto de interrogantes, que permita aclarar y definir de manera básica las expectativas de la gerencia.

- ¿El aplicativo actual cumple con los expectativas de la organización?
- ¿Qué tan factible es realizar el mantenimiento al aplicativo?
- ¿Cuánto invierten en agregar mejoras al aplicativo?
- ¿Con que frecuencia se agregan mejoras al aplicativo?

Las respuestas de estas preguntas permitirán dar forma a los siguientes elementos de análisis.

**Objetivo:**

Definir las metas del proyecto; estas deben ser detalladas y específicas, con el fin de contar con mayor detalle posible de lo que se desea conseguir.

**Audiencia:**

Grupo de personas que involucradas que tienen la finalidad de migrar un aplicativo legado.

En cuanto más claras sean las respuestas a estas preguntas, se obtendrá los resultados esperados de la migración, que son básicamente replicar los procesos que se tiene en el aplicativo legado, al aplicativo migrado.

**5.1.1. Planificación del proyecto**

En esta etapa se debe definir la forma en que se desarrollara la migración del aplicativo en la organización que lo requiera y para ello se debe tomar en cuenta:

- Formar equipo de trabajo por parte de TI y la organización.
- Realizar el levantamiento de necesidades de la organización.

**5.1.1.1. Equipo de Trabajo**

Para poder desarrollar correctamente una migración, es necesario contar con un equipo de trabajo que realice las tareas propuestas y afronte los desafíos que se irán presentando durante el desarrollo del proyecto en sus diferentes etapas hasta la puesta en marcha del mismo.

Para ello se recomienda la conformar los siguientes grupos, tomando en cuenta que una misma persona puede desempeñar diferentes papeles en el desarrollo del proyecto, es decir, ser multifuncional.

#### **a. Creación de Equipo Legado**

Este es un equipo de trabajo multidisciplinario que integra las diferentes visiones del aplicativo legado, generando un grupo de personas que interpreta los procesos existentes.

Los integrantes del presente comité provienen generalmente de las áreas de gestión, informática, comunicaciones, y en conjunto su trabajo es conseguir varias metas, entre las que resaltamos las más importantes:

- Conseguir el compromiso de todas las áreas en la organización con la migración que se desarrolla.
- Los procesos principales que realiza actualmente el aplicativo legado.
- Los procesos del aplicativo legado que cuenta con carencias.
- Identificar los procesos que no son usados.

#### **i. Funciones**

Para que el comité de expertos legados funcione adecuadamente debe cumplir con lo siguiente:

Contar con un jefe operacional que se encargue de que el comité funcione dentro de los márgenes establecidos.

Efectuar reuniones diarias para brindar el conocimiento que se tiene del aplicativo legado, se recomienda que cada reunión sea para instruir un módulo distinto.

Desarrollar una política de redacción donde se plasme todas las funciones que realiza el aplicativo legado.

## ii. **Integrantes**

**Gerente General:** Es el responsable superior de la organización. Aporta con el objetivo general de la institución, la cual es el inicio para poder otorgar el visto bueno, y de esta manera poder realizar la migración. Su presencia no es necesaria para todas las reuniones.

**Líder experto – Legado:** Es la persona que conoce más el aplicativo legado, generalmente es la que lleva más tiempo en la organización usando dicho aplicativo. Asignará el responsable de cada módulo para brindar toda la información posible.

**Expertos de Contenidos:** Grupo de personas tienen mayor experiencia en el uso de cada módulo, respecto al aplicativo legado.

**Validadores de contenidos:** Por lo general son las personas que conforman el grupo de Experto de Contenido, siendo la tarea a desempeñar probar el aplicativo migrado, y asegurar que el funcionamiento sea igual al aplicativo legado, a menos que

exista una mejora de por medio, y de esta manera poder asegurar la calidad al desplegar la migración.

### **b. Creación del Equipo de Migración**

Este equipo de trabajo lo integran personas con más experiencia en migraciones del área de TI, y su trabajo es conseguir varias metas, entre las que resaltamos las más importantes:

- Alertar posibles problemas técnicos que se presenten al realizar la migración.
- Identificar el riesgo que se tiene al migrar el aplicativo.
- Proponer alternativas de solución a los problemas técnicos que presente el aplicativo legado.

#### **i. Funciones**

Para que el comité de expertos de migración funcione adecuadamente debe cumplir con lo siguiente:

- Contar con un jefe operacional que se encargue de que el comité funcione dentro de los márgenes establecidos.
- Efectuar reuniones diarias con el fin de extraer todo el conocimiento del aplicativo legado.
- Efectuar reuniones semanales con la finalidad de verificar el avance y alertar los riesgos y problemas que se tenga al migrar el aplicativo.
- Desarrollar una política de redacción donde se plasme todas las funciones que realiza el aplicativo legado

como requerimientos para cada módulo [Anexo 1], esto debe ser aprobado por el comité de expertos - legados.

## ii. **Integrantes**

**Líder de proyecto:** Es el administrador del proyecto y responsable frente a la migración del aplicativo. Sus tareas principales son:

- Traspasar la visión estratégica organizacional en un modelo conceptual sólido consensado para ser implementado en el aplicativo migrado.
- Responsable de detectar las necesidades de los usuarios y gestionar los recursos humanos y técnicos, para obtener los resultados esperados en los plazos previstos y con la calidad necesaria.
- Dirigir y coordinar el proyecto de migración incluyendo las mejoras que se pueda realizar, supervisando las funciones y los recursos de análisis funcional, técnico y programación, con el fin de satisfacer las necesidades de los usuarios y asegurando la adecuada explotación de la migración.
- Realizar la planificación del proyecto de migración.

**Líder experto – Migración:** Es la persona que tiene mayor experiencia realizando migraciones. Responsable de la parte técnica de la migración.

**Analistas:** Grupo de personas encargadas de examinar y evaluar el aplicativo legado a través de un examen del procesamiento, así como de la entrada y salida de datos, y la producción de dicha información; proporcionando una solución de migración.

También puede realizar roles de consultor, agente de cambio.

**Desarrolladores:** Grupo de personas que realizarán la construcción del aplicativo migrado, basándose en las soluciones que colocó el analista.

**Validadores Técnicos:** Desempeñan esta función los profesionales informáticos, realizando las pruebas internas para asegurar la calidad del proyecto.

**Soporte Técnico:** Resolver los problemas que puedan presentárseles a los usuarios, mientras hacen uso de determinado servicio, programa, equipo o dispositivo.

### c. Rol de las autoridades superiores

El objetivo es realizar la migración de forma satisfactoria, por ende, debe haber la decidida intensión en la organización de otorgarle prioridad al proyecto en cuanto a recursos humanos, financieros y el tiempo adecuado para su correcta ejecución, puesto que la migración será exitosa en la medida en la que cumpla con los objetivos planteados.

### 5.1.1.2. Levantamiento de Necesidades del Aplicativo Migrado

Está profundamente relacionado con la definición de los objetivos que deberá cumplir el proyecto de migración que se desea desarrollar.

Nos permitirá definir de manera clara y comprensible para todos los involucrados, lo que se espera obtener del proyecto que se va a migrar, y que puedan contar con la misma información para actuar de manera coordinada a lo largo del desarrollo del mismo.

### 5.1.1.3. Instrumentos de seguimiento

La etapa de seguimiento tiene un rol muy importante, debido a que esta etapa se encarga de supervisar continuamente la ejecución de las tareas incluidas en el proyecto, con el propósito de asegurar que este se va desarrollando de acuerdo al plan previsto.

Se debe de supervisar los siguientes planos simultáneos:

- **Desarrollo de tareas:** Son todas aquellas revisiones de las actividades técnicas y operativas que están contempladas en el proyecto.
- **Revisión de impacto:** Define en qué áreas el proyecto causa problemas o si existen cambios que necesitan ser revisados.
- **Factores externos:** Son para tener la atención en todos aquellos elementos los cuales no están vinculados al proyecto, pero que en cierta parte puedan estar afectando.

Gracias al resultado obtenido en el proceso de desarrollo de estos 3 planos mencionados, se podrá tener información actualizada que nos ayude con

las detecciones de desviaciones respecto a la planificación prevista; redefinir la estrategia además de la dirección del proyecto y poder tomar decisiones de forma más rápida y adecuada para hacer las correcciones que se estimen necesarias.

Para llevar a cabo el seguimiento correspondiente en estos tres planos, necesitaremos de las siguientes herramientas tecnológicas:

#### **a. Metodología y software de seguimiento**

Para realizar el seguimiento del proyecto debemos contar las herramientas que puedan facilitarnos el control y nos permitan relacionar cada una de las etapas con el resto de las existentes, esto nos ayudará a descubrir qué etapa está más atrasada y afecta a las siguientes y con ello al conjunto del trabajo.

Para esto surgen dos técnicas conocidas que son la Carta Gantt y la Malla Pert, que son útiles y a través de ellas conseguimos objetivos antes mencionados.

Se recomienda usar Microsoft Project, y este será parte de uno de los entregables de esta fase, será registrado como SPM (Seguimiento de Proyecto de Migración). [Anexo 2]

#### **b. Reuniones de Control, minutas e informes de avance**

Para hacer el seguimiento de un proyecto es necesario contar con alguna herramienta que facilite el control y permita establecer la relación de cada una de las etapas con el resto de las existentes,

para la presente tesis usaremos el control en el acta de reuniones (ACR) [Anexo 3]; éstas se realizarán semanalmente y deben de estar basadas en la carta de Gantt, con la finalidad de corregir los problemas que se vayan presentando; de esta manera evitar algún tipo de retraso y hacer peligrar el proyecto.

#### 5.1.1.4. Recepción del Proyecto

Para la recepción se debe solicitar al menos lo siguiente:

##### a. Documentación

Es importante hacer notar que además de recibir el Sitio Web funcionando y cumpliendo con las tareas que se le habían solicitado, es necesario que los desarrolladores entreguen al menos la siguiente documentación que debe haber sido solicitada en los Términos de Referencia:

- Código fuente: Corresponde a los programas de todas las aplicaciones construidas; debe ser entregado en formato digital y debidamente documentado para entender de qué manera funciona.
- Documentación: Corresponde a informes completos respecto de las características de todos los elementos del Sitio Web; si se trata de sistemas, debe incluir los manuales de usuario y administrador que permita entender cómo funciona y qué hacer para resolver problemas; debe ser entregado en formato digital, además de impreso.

- Elementos gráficos: Corresponde a todas las imágenes del Sitio Web, para las cuales se pide que sean entregadas en el formato original en que fueron construidas y que puedan ser editadas, es decir, cambiadas usando el mismo programa. Lo anterior significa que no debe aceptarse que las imágenes sean entregadas en formato GIF o JPG usado en el Sitio Web, ya que eso impide su edición.

#### **b. Capacitación**

También es importante considerar que, si el desarrollo de una aplicación o Sitio Web lo realizó una empresa externa a la institución, será necesario recibir capacitación para entender cómo funciona. Para ello es indispensable contar con horas de enseñanza de parte de los desarrolladores en dos ámbitos:

- Capacitación Administradores: Es la que recibe el personal informático de la institución y que le permite entender cómo ha sido construido el Sitio Web y la forma de mantenerlo, respaldarlo e incluso, mejorarlo. Debe recibir información en estos tres aspectos, con el fin de que pueda operar de manera autónoma una vez que haya concluido la Garantía Técnica que haya contemplado el proyecto.
- Capacitación Usuarios: Es la que reciben los usuarios avanzados que estarán a cargo de la mantención del Sitio Web y que les permite familiarizarse con todos los

elementos que se deban conocer para utilizar el sitio en toda su potencialidad.

### 5.1.2. Entregables

- **SPM – Seguimiento de Proyecto de Migración:** Se registra las tareas que se realizarán en todo el proyecto, desde el análisis hasta la implementación. En caso de certificación se tomarán en cuenta un máximo de 2 ciclos de prueba.
- **Check List de Planificación:** Es la información general que se tiene del aplicativo legado, como información entrante y saliente. [Anexo 4]
- **Check List de Migración:** Es el documento que registra el aspecto técnico del aplicativo legado. [Anexo 5]
- **Check List de SPM:** Lista de criterios que se toman en cuenta para poder validar las tareas que se están tomando en cuenta para el proyecto de migración. [Anexo 6]
- **Bloque de Preguntas:** Se registran las evidencias de consultas al equipo legado, evidenciando con los correos correspondientes. [Anexo 7]
- **ACR - Acta de Reunión:** Es el acta donde se archivan todas las reuniones que se realizaron en la fase de planificación, quienes asistieron, duración, etc. Para tener archivadas dichas reuniones a manera de control interno.

### 5.2. Análisis y Diseño

Esta es la segunda fase de la metodología, en donde se realiza el análisis para la migración respecto al aplicativo legado, incluyendo todos los procesos

necesarios. Se aplicarán mejoras simples al aplicativo, donde no varíe la lógica ni los procesos.



*Figura 20: Metodología de Migración - Análisis*

### 5.2.1. Involucrados

Para el desarrollo de esta etapa es necesario contar con los siguientes involucrados:

#### a. Líder de proyecto

- Establecer los horarios de trabajo del equipo de desarrollo.
- Planificar y realizar las reuniones de control del equipo de analistas en el tiempo establecido.
- Mantener el control del resultado de estas reuniones.
- Informar sobre el estado actual del proyecto.
- Suministrar al equipo el informe del ciclo de migración

#### b. Experto Migraciones

- Dará a conocer los alcances necesarios para poder completar con los objetivos de la etapa de análisis, gracias a la experiencia con la que cuenta realizando proyectos de migración.
- Brindará soluciones a los procesos más complicados que se tenga al momento de proponer una solución del aplicativo legado al aplicativo migrado.

**c. Experto Legado**

- Brindará el conocimiento del aplicativo legado, desde el código fuente hasta el funcionamiento.
- Dará a conocer los alcances de la migración, desde el funcionamiento hasta el negocio del aplicativo legado.

**d. Analistas**

- Capturar los requisitos que se necesite para poder realizar la migración.
- Con el conocimiento que brinde el experto legado, los analistas deberán de proveer una solución para el aplicativo migrado, para la arquitectura, diseño y negocio.

**5.2.2. Captura de Requisitos**

Es necesario averiguar que se debe construir, para ello se deben recoger todos los requisitos posibles, por lo tanto, es la etapa más complicada y poco común, pero a la vez muy necesaria, ya que ayuda a guiar el desarrollo del proyecto hacia una migración correcta y así llegar a un acuerdo entre el cliente y el comité de migración. Ya que cada proyecto es diferente, las singularidades del mismo provienen de los clientes.

El método que se propone para la captura de requisitos es mediante casos de uso, ya que con ellos los analistas se ven obligados a pensar en términos de quienes son los usuarios y que necesidades u objetos de la empresa pueden cumplir.

### 5.2.3. Inicio de Análisis – Arquitectura

La arquitectura de información estará cumpliendo sus objetivos cuando un usuario entre por primera vez al aplicativo migrado y pueda reconocer que es similar al aplicativo legado; lo pueda entender en forma rápida y sin esfuerzo y encontrar la información ofrecida fácilmente y hacer uso de los servicios que este le brinda. Adicionalmente eso entregará el beneficio de que quienes producen el aplicativo podrán ubicar la nueva información sin tener que crear nuevas estructuras y al mismo tiempo tendrán la libertad de incorporar nuevas iniciativas al aplicativo sin tener que partir de cero.

Los elementos que se muestran a continuación constituyen una metodología, mediante la cual es posible conseguir las metas de organización, objetivos del aplicativo y visibilidad de los contenidos:

- Definición de objetivos del aplicativo migrado
- Definición de Audiencia
- Definición de contenidos del aplicativo migrado
- Definición de la estructura del aplicativo migrado
- Definición del diseño visual

#### 5.2.3.1. Definición de Objetivos

La finalidad de esta etapa es buscar definir cuáles serán los objetivos centrales que deberá tener el proyecto de migración y establecer la mejor forma de cumplirlos.

Se recomienda comenzar por generar un objetivo central y luego definir varios objetivos secundarios; no obstante, es importante que estos

objetivos sean explicados adecuadamente al comienzo del desarrollo, con el fin de que todo el equipo de profesionales involucrados, tanto el comité legado como el comité de migración, tenga claro el horizonte del proyecto.

Para generar objetivos que sean válidos y comprensibles, se aconseja escribirlos a través de frases que se inicien con verbos que expresen las acciones a realizar. Por ejemplo: Recibir preguntas de los usuarios sobre las actividades que realizan en el aplicativo legado.

Una forma de llegar más concretamente a los objetivos, es revisar la visión y la misión que tenga actualmente el aplicativo legado, desde la cual se pueden obtener las claves que permitan definirlos. Los objetivos que se creen para el aplicativo migrado deberán estar en concordancia con las necesidades y planificación que haya hecho la institución con el aplicativo legado.

Idealmente el listado de objetivos de la migración debe estar acotado y no llegar a más de cinco proposiciones. Este número, si bien es arbitrario, se indica como una forma de señalar que los objetivos no pueden ser tantos como para impedir su cumplimiento; ni tan pocos como para que la migración sea poco ambiciosa.

De cada uno de estos objetivos se pueden desprender tareas concretas que permitan avanzar en el cumplimiento de ellos.

#### **5.2.3.2. Definición de Audiencia**

Al culminar adecuadamente la etapa anterior con la generación de los objetivos de la migración, se debe dar un siguiente paso, que es el de

determinar las principales audiencias (públicos) hacia las cuales se orientará la migración.

Está claro que el aplicativo migrado será atendido por el mismo público que atiende actualmente el aplicativo legado. Sin embargo, hay varias consideraciones que tener en cuenta, al ser una herramienta nueva.

A continuación, se hacen algunas definiciones de audiencia, las que, sin importar las características del aplicativo migrado, siempre deberán estar presentes, y por lo mismo, deben traducirse en acciones para poder atenderlas adecuadamente.

**Por capacidad técnica:** La audiencia que llegue al aplicativo migrado se dividirá de acuerdo a la experiencia técnica que tenga; por ello se deben plantear accesos simples mediante enlaces y otros más complejos, por ejemplo, mediante el uso de buscador. Es importante notar que en Perú desde hace algunos años más del 50 % de la población accede a la Internet [28], por lo que cada año, se puede presentar audiencia que tiene poca o nula experiencia en el uso de Internet.

**Por conocimiento de la institución:** Los usuarios del aplicativo migrado son divididos entre quienes conocen la institución y quienes no la conocen. Por lo anterior, los primeros siempre sabrán dónde buscar lo que necesitan usando la terminología, siglas y nombres de departamentos internos; los segundos, en tanto, no entenderán nada de la nomenclatura interna y les será muy difícil acceder a la información que se les ofrezca de esa manera.

**Por necesidades de información:** Los usuarios del aplicativo migrado también se dividirán entre quienes llegan a buscar contenidos determinados y quienes sólo llegan a ver si existe algo que les pueda servir en lo que estén realizando.

**Por ubicación geográfica:** El aplicativo migrado al ser un sitio web, provocara una demanda de audiencia que ingresen al aplicativo desde lugares diferentes a la ciudad de Arequipa o incluso Perú, por lo que los contenidos deben responder también a esta diversidad.

### 5.2.3.3. Definición de Contenidos de la Migración

Una vez que se han identificado los objetivos del sitio y la audiencia, se debe proceder a hacer las definiciones más concretas que permitan decidir qué contenidos son los que va a tener el aplicativo migrado que se desarrolle.

Para identificar contenidos, se deben utilizar como insumo los materiales que se hayan obtenido en la etapa de identificación de Objetivos y de Audiencias, ya que en ambos la búsqueda giró en torno a las necesidades que tenían los usuarios del aplicativo legado.

Para cumplir con una norma general respecto de qué debería contener el aplicativo migrado, se pueden anotar las siguientes, como las más importantes:

**Acerca de la Institución:** Entregar la información completa referida a Autoridades, Organigrama, Normativa legal asociada, Oficinas, Horas de Atención, Teléfonos, etc.

**Productos / Servicios:** Destacar las actividades principales que el usuario puede hacer en el aplicativo legado; puede incluir una guía de trámites que facilite las acciones de las personas que acudirán a la institución y que considere servicios interactivos para hacerlos desde el aplicativo migrado.

**Novedades de la Institución:** Últimas actividades, noticias, etc.

#### a. Agrupar y etiquetar contenido

Con las definiciones hechas hasta ahora, llega el momento de poner en práctica las metodologías que permiten ordenar los contenidos, agrupándolos en conjuntos coherentes y dándoles nombres que los identifiquen.

Probablemente la mejor técnica para hacerlo es hacer unas pequeñas tarjetas de papel, en las que se anotan las principales áreas de contenido que se hayan detectado y revisado en el paso anterior. Una vez hecho, las cartas se ponen sobre una mesa y se van agrupando, hasta formar conjuntos de elementos coherentes entre ellos. Luego, a cada conjunto se le pone un nombre (idealmente una sola palabra) que identifique a todos sus contenidos.

Con esas agrupaciones hechas, ya tendremos los elementos adecuados para generar posteriormente el árbol de contenidos que, a su vez, permitirá hacer el sistema de navegación.

Una vez que el proceso de Etiquetado ha concluido, es bueno hacer comprobaciones empíricas de la validez de los nombres escogidos.

Para ello, se requiere que los elegidos sean mostrados a personas de diverso origen y que conozcan el aplicativo legado, como también a quienes la desconozcan por entero. Ellos deben responder las siguientes preguntas:

- ¿Qué significa este nombre?
- ¿Qué tipos de contenidos esperaría encontrar en esta área?

Con las respuestas obtenidas se podrá juzgar si los nombres que se han usado son los más adecuados o, bien, hay que introducir modificaciones.

Por ejemplo, dentro de los nombres más usados para una de las secciones habituales de un Sitio Web como es la de información corporativa, se tiene la sub-sección de *Acerca de, Quiénes Somos* e Información Corporativa.

#### **b. Identificar requerimientos funcionales**

Junto con la búsqueda de las áreas de contenido que deberá tener el aplicativo migrado, se debe trabajar también en la definición de lo nuevo que se requiere, es decir, los tipos de interacción que se busca incluir.

Dentro de las funcionalidades de mayor complejidad, pero a las que todo sitio debería aspirar, se cuenta:

- Buscador interno del Sitio Web
- Área de acceso privado para usuarios registrados

- Sistema de envío de boletines de noticias del sitio a usuarios registrados

Será importante que el aplicativo migrado cuente con todos los servicios interactivos descritos como mínimos, respecto al aplicativo legado, para ofrecer una mejor experiencia al usuario que lo visita.

#### **c. Análisis de requerimientos funcionales similares**

Con ayuda del experto de migración se buscarán los requerimientos funcionales que tengan alguna semejanza, tanto en el diseño como en lo codificable, para evitar el re-trabajo al momento de realizar la construcción de la migración.

Es importante que se haga una pauta previa de comparación, con el fin de saber de antemano cuáles son los parámetros que se medirán y gracias a eso, optimizar la revisión que se haga.

#### **d. Análisis de aplicativos migrados similares**

El último elemento que se debe desarrollar en esta etapa es la búsqueda de otras migraciones en nuestro almacén de software que sean similares a nuestro aplicativo migrado, con el fin de revisar de qué manera han resuelto los mismos problemas que deberemos atender.

Esta actividad comparativa permitirá llegar a las buenas prácticas que es posible adaptar a las necesidades de nuestro sitio, con el fin

de asegurarnos métodos de funcionamiento y despliegue de contenidos, que sean coherentes con los objetivos que se han planteado inicialmente.

#### 5.2.3.4. Definición de Estructura de la Migración

Una vez que se ha hecho el trabajo de identificación de contenidos señalado en las etapas anteriores, se debe avanzar hacia las definiciones relacionadas con la forma que tendrá la migración que se está desarrollando. Ello implicará trabajar en tres áreas concretas, a través de las cuales se definirá la estructura del aplicativo migrado, el árbol de contenidos y los sistemas de navegación que se ofrecerá a los usuarios para que avancen a través de sus contenidos.

##### a. Creación de la estructura

Se refiere al proceso de identificar la forma que tendrá el aplicativo migrado que se está desarrollando. En este sentido es importante hacer una diferencia entre estructura y diseño.

**Estructura:** Se refiere a la forma que tendrá el aplicativo migrado en términos generales con sus secciones, funcionalidades y sistemas de navegación. No considera ni incluye elementos gráficos (logotipos, viñetas, etc.).

**Diseño:** Se refiere a la solución gráfica que se creará para el sitio, en la cual aparecen colores, logotipos, viñetas, y otros elementos de diseño que permiten identificar visualmente al sitio.

Dado lo anterior, cuando hablamos de la estructura nos estamos refiriendo básicamente a cuál será la experiencia que tendrá un usuario cuando accede al aplicativo migrado. De esta manera podremos determinar dónde estarán ubicados los servicios interactivos (buscador, sistemas de encuestas, áreas de contenidos).

Básicamente la estructura estará basada en el aplicativo legados esto nos permitirá discutir en términos muy prácticos cuál será la oferta de elementos de información e interacción que tendrá el usuario. Al no incluir elementos de diseño, se permite que la discusión sobre la estructura se desarrolle en aspectos concretos, sin que intervengan aún consideraciones estéticas que habitualmente atrasan la aprobación de esta etapa del desarrollo.

#### **b. Mapas permanentes del aplicativo migrado**

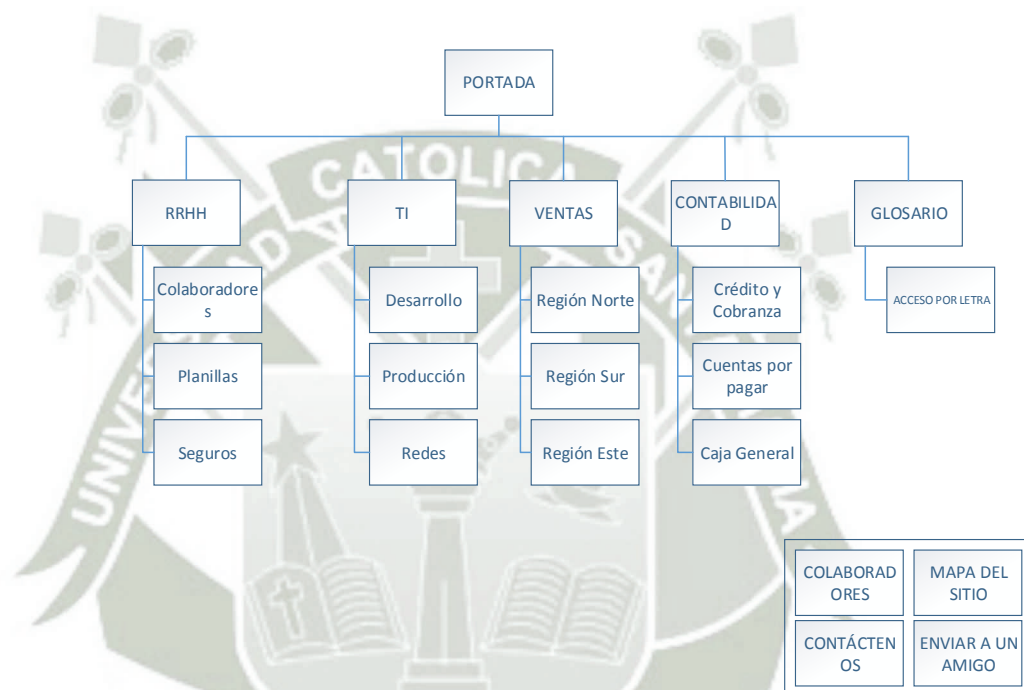
Se refiere al proceso de crear un árbol de contenido en el que se muestre de manera práctica cuántas secciones tendrán la migración en desarrollo y cuántos niveles habrá dentro de cada uno.

Cuando se usa la idea de crear un árbol, se refiere exactamente a generar un diagrama que cuente con un tronco, ramas y hojas, para mostrar las zonas principales, secundarias y contenidos finales que se irán incorporando.

En este sentido se debe evitar a toda costa que el árbol de contenidos represente la estructura de la organización, dado que ésta es conocida y comprendida internamente, pero constituye una

barrera de entrada para usuarios externos. Si ellos llegan a buscar en una estructura de contenidos basada en la forma en que funciona la organización, primero deberán comprender cómo funciona el aplicativo legado para luego encontrar lo que nos interesa.

En este sentido se sugiere el uso del Árbol Funcional, como se muestra en la siguiente figura.



**Figura 21: Árbol Funcional**

Dado lo anterior, las recomendaciones para la generación de este árbol son las siguientes:

**Secciones:** En caso las secciones del aplicativo legado estén correctamente ordenados y cumplan con lo indicado anteriormente, se mantendrán las secciones en el aplicativo migrado.

En caso no cumplan con lo indicado se debe intentar que sean las menos posibles, con el fin de concentrar las acciones del usuario en pocas áreas; hay que considerar que cada una de las áreas a integrar en el árbol requerirá de mantenimiento posterior en contenidos, gráfica y funcionalidad, lo que encarecerá el costo final de operación del aplicativo migrado.

**Niveles:** Se debe intentar que el usuario esté siempre a menos de tres clicks del contenido que anda buscando. Por ello no se debería crear más de tres niveles de acceso; se recomienda una Portada, una Portadilla de Sección y los Contenidos propiamente tales.

**Contenidos relacionados:** Se debe considerar que habrá funcionalidades que estén presentes en todo el sitio. Entre ellas se incluyen elementos como Buscador, Preguntas Frecuentes y Formularios de Contacto. Se recomienda que este tipo de elementos quede fuera del árbol y floten sobre éste, con el fin de indicar que desde todas las páginas habrá enlaces a ellos.

### c. Definición de los sistemas de navegación

Una vez que se cuenta con los árboles de contenidos en el paso anterior, la tarea siguiente consiste en generar los sistemas de acceso a dichos contenidos en el aplicativo migrado. A través de estos, los usuarios podrán avanzar por sus diferentes áreas, sin perderse.

En la generación de dichos sistemas se debe atender a dos elementos que serán muy importantes:

**Textual:** Se refiere a que la navegación se hará a través de elementos concretos, tales como menús, guías, botones y otros elementos que deben ser claramente distinguibles dentro de la interfaz. Para generarlos se debe conseguir que cada uno de ellos represente claramente la función para la que fueron designados y no dejar lugar a dudas sobre su función ni sobre la acción que desarrollarán al ser usados. Es decir, un botón debe parecer tal y no sólo un parche de color sobre la pantalla. Adicionalmente, es muy importante que las palabras escogidas para indicar acciones, sean claras y precisas. En este sentido, si un botón necesita ser explicado, es mejor desecharlo y buscar otra solución.

**Contextual:** Es todo lo referido a cómo se presenta la información, utilizando para ellos elementos basados en texto, gráficos o bien de entorno. Los elementos relevantes en este caso, serán todos aquellos que permiten mostrar la navegación en la pantalla. Entre ellos, la gráfica utilizada, la redacción de los textos que se muestran e incluso el nombre del dominio (URL) que permitirá que el usuario sienta que está en el lugar indicado.

Al generar el sistema de navegación, se deben tener en cuenta las siguientes características:

**Consistente:** El aplicativo migrado debe ser similar al aplicativo, en lo referido a su ubicación y disposición en las páginas.

**Uniforme:** El aplicativo migrado debe utilizar similares términos con el fin de que el usuario que lo vea en las páginas, confíe en que sus opciones llevan siempre hacia los mismos lugares dentro del sitio.

**Visible:** El sistema debe distinguirse claramente, con el fin de que el usuario cuente con él, como si se tratara de una guía permanente en el área en que se encuentre.

#### **d. Elementos de los sistemas de navegación**

Entre los elementos más relevantes que conforman el sistema de navegación se cuentan los siguientes:

**Menú General:** Siempre presente en todo el aplicativo migrado, permite el acceso a cada una de las áreas del aplicativo.

**Pie de Página:** Ubicado en la parte inferior de cada página, indica el nombre de la institución, teléfonos, dirección física y de correo electrónico.

**Barra Corporativa:** Ofrece diversas opciones de información respecto del sitio y tal como el anterior, se muestra en todas las páginas.

**Ruta de Acceso:** Listado que aparece en la parte superior de cada página y que muestra el trazado de páginas que hay entre la Portada del aplicativo hasta la página actual que se esté revisando; cada una de ellas debe tener un enlace, para acceder al área de la cual depende la página. Cada uno de los elementos que conforman este camino debe tener un enlace que permita el acceso a esas áreas. En la literatura internacional, se llama a este elemento como breadcrumbs [29].

**Fecha de publicación:** Para saber la vigencia de publicación del contenido desplegado.

**Botón Home:** Para ir a la portada

**Botón Mapa del sitio:** Para ver el mapa del Sitio Web

**Botón Contacto:** Para enviar un mensaje al encargado del sitio.

**Buscador:** Presente en cada página si es que la funcionalidad existe en el sitio.

**Botón Ayuda:** Para recibir ayuda sobre qué hacer en cada pantalla del sitio.

**Botón Imprimir:** Para imprimir el contenido de la página; se espera que el formato de impresión del documento que se muestra

en pantalla sea más simple que la página normal del aplicativo, para dar la impresión al usuario de que hay una preocupación por ayudarlo en la tarea de llevar impreso el contenido.

#### **5.2.3.5. Definición de Diseño Visual**

Una vez que se ha terminado el trabajo en la estructura, ya se cuenta con los insumos suficientes como para avanzar hacia la generación del diseño visual de las pantallas del sitio, momento en que se utilizarán todos los insumos que se han ido generando en las etapas anteriores.

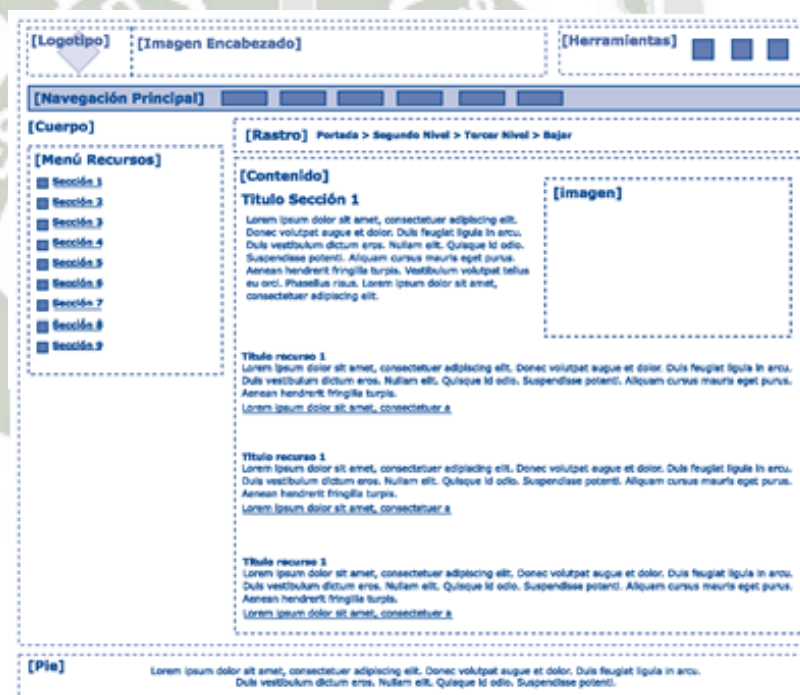
Para ello la recomendación es trabajar en cuatro etapas sucesivas e incrementales, que se describen a continuación:

##### **a. Diseño de estructuras de pagina**

Esta etapa considera la generación de dibujos sólo lineales que describen los componentes de cada una de las pantallas de la migración, con el objetivo de verificar la ubicación de cada uno de ellos.

El ideal es que se dibujen diagramas con todas las pantallas que tendrá el sitio, ya que de esta manera será posible que los desarrolladores tengan un documento concreto de trabajo, a través del cual resuelvan todas las dudas de los elementos que componen esta página. Si bien este trabajo es largo y puede resultar tedioso, su ventaja es que ningún elemento en las páginas queda puesto al azar, sino que responde a necesidades puntuales que se han detectado y que se resuelven por esta vía.

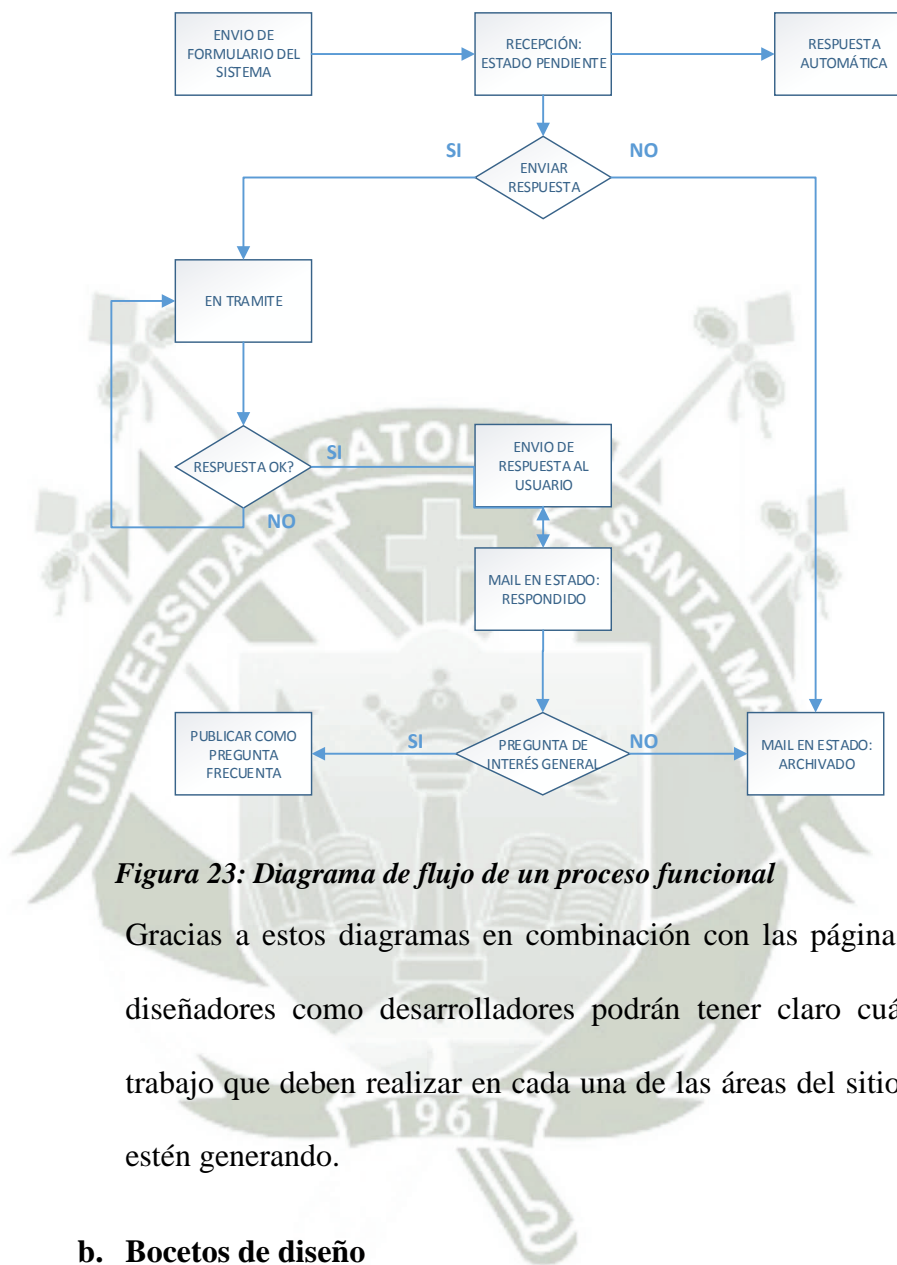
Nuevamente, tal como se planteó cuando se estaba presentando el tema de Estructura en las páginas anteriores, lo ideal es que estos dibujos no tengan ningún elemento gráfico o visual concreto, sino que sólo incluyan líneas y bloques que representen objetos de contenido (como logos, viñetas o fotos). Nuevamente, como en ese caso, el uso de estas imágenes ayudará a que la discusión sobre cada pantalla se centre en la funcionalidad y no en temas más subjetivos como colores o calidad de los elementos de diseño en la página.



**Figura 22: Estructura de la página**

Un elemento que se debe considerar junto con estos dibujos de estructura, es que en las pantallas que representen transacciones, se debe incluir un diagrama de flujo sencillo, mediante el cual se ejemplifique cuáles son las interacciones posibles y sus resultados.

Naturalmente se deberá incluir las pantallas correspondientes cuando sea adecuado.



**Figura 23: Diagrama de flujo de un proceso funcional**

Gracias a estos diagramas en combinación con las páginas, tanto diseñadores como desarrolladores podrán tener claro cuál es el trabajo que deben realizar en cada una de las áreas del sitio que se estén generando.

**b. Bocetos de diseño**

Esta etapa consiste en la generación de dibujos digitales acabados de la forma que tendrán las páginas principales del sitio que se desarrolla, considerando como tales la Portada, Portada de Sección y Página de despliegue de contenidos.



*Figura 24: Boceto de diseño*

Para desarrollar los elementos gráficos se utilizan como insumos los dibujos de estructura que se han generado en la etapa anterior.

La idea es que en esta etapa se trabaje en software gráfico para facilitar el proceso de corrección, ya que habitualmente habrá mucha interacción con los usuarios.

### c. Borradores del aplicativo migrado

Una vez que se ha aprobado la etapa anterior, se toman los bocetos de diseño que hayan sido aprobados y se genera un prototipo (páginas clickeables) mediante el cual se pueda comprobar directamente la forma en que se desempeñan.

La intención de esta etapa es usar el diseño de pantalla que se ha creado, contando con enlaces reales que permitan ver la forma de usar sus atributos (cada enlace tiene los estados de Enlace, Flotante, Activo y Visitado) y revisar la forma en que se despliegan las páginas que se van a desarrollar.

Adicionalmente, en esta etapa será posible hacer la comprobación efectiva del peso de las páginas una vez construidas y se podrán optimizar los elementos que queden fuera de norma por peso o tamaño.

#### **d. Maqueta del aplicativo migrado**

Es la etapa final y consiste en generar todo el sitio en tecnología HTML utilizando imágenes y contenidos reales.

Es importante considerar que todas las tareas y actividades incluidas en esta etapa pueden ser abordadas por un diseñador que tenga experiencia en diseño de migraciones, es decir al experto de migración.

#### **5.2.4. Entregables**

- **DAM - Documento de Análisis de la Migración:** Esta será realizada por los analistas, con este documento darán por finalizada la etapa de análisis y diseño. Este documento tendrá el contenido de todo el análisis, proponiendo más de 2 alternativas de migración, y como deberá ser desarrollada en la fase de construcción. [Anexo 8]
- **PUE – Pruebas Unitarias Específicas:** Se adjuntan en el DAM, estas pruebas son para corroborar el funcionamiento del módulo en construcción. [Anexo 9]
- **ACR - Acta de Reunión:** Es el acta donde se archivan todas las reuniones que se realizaron en el proceso, quienes asistieron, cuanto duró, etc. Para tener archivadas dichas reuniones a manera de control interno.

- **Bloque de Preguntas:** Se actualizan las evidencias de consultas al equipo legado, evidenciando con los correos correspondientes.

### 5.2.5. Control de Calidad – Análisis y Diseño

- **CCI – Control de calidad interno:** Para el control de calidad interna es necesario que los entregables estén conformes, luego se elegirá a un analista del proyecto para que revise si lo desarrollado cumple con los requisitos solicitados en el DAM, los entregables cumplan con lo necesario y este control será registrado en un documento CCI. (Anexo 10).
- **CCE – Control de calidad externo:** Para el control de calidad externa es necesario que el documento CCI se encuentre en estado conforme, posteriormente se elegirá a un analista externo que esté a cargo de otro proyecto, para que revise si lo desarrollado cumple con los requisitos solicitados en el DAM, y este control será registrado en un documento. (Anexo 11)

### 5.3. Construcción

Es la tercera fase de la metodología, la cual se enfoca en la codificación del aplicativo, siguiendo los requerimientos del DAM (Documento de análisis de la migración), documento el cual se obtiene en la fase anterior de Análisis y Diseño.



*Figura 25: Metodología de Migración - Construcción*

A partir de esta fase, se cuentan con ambientes para cada una de las etapas siendo las siguientes:

- Ambiente de desarrollo para construcción
- Ambiente de Certificación para realizar las pruebas
- Ambiente de Producción para la implementación



**Figura 26: Esquema del ambiente de desarrollo**

En este caso tenemos el primer ambiente (Desarrollo) donde será necesario el uso de computadores para codificar el aplicativo, y luego pasar a los siguientes procesos que también contarán con sus respectivos ambientes.

En esta etapa se tomarán en cuenta todas las medidas para poder ejecutar el desarrollo de la migración, para ello tendremos que contar con el DAM (Documento de Análisis de Migración), el grupo de involucrados y el plan del proyecto para realizar una migración exitosa, además se corroborará que la transferencia de información del análisis sea la correcta.

## Objetivos

- Realizar la codificación de migración.
- Realizar las mejoras indicadas por los expertos legados, respecto al aplicativo legado.
- Corregir errores hallados en la migración pos-certificación.

## Audiencia

- Es el grupo de personas implicadas en el desarrollo de la migración, los cuales serán los encargados de colaborar con sus conocimientos a nivel de codificación, en el proceso de migración.

### 5.3.1. Involucrados

#### a. Líder de proyecto.

- Seleccionar el equipo de desarrollo que estará involucrado en la migración.
- Establecer los horarios de trabajo para el equipo de desarrollo.
- Supervisar las reuniones de control del equipo de desarrollo en el tiempo establecido.
- Conservar el control del resultado de estas reuniones.
- Informar sobre el estado actual del proyecto.
- Suministrar al equipo el informe del ciclo de migración
- Dar el conforme de la construcción, para poder avanzar a la etapa de certificación.

**b. Experto Migración.**

- Es la persona con mayor experiencia en procesos de migración que facilitará la toma de decisiones para la realización del proyecto.
- Apoyar en los procesos de codificación del aplicativo de migración.

**c. Desarrolladores.**

- Son los encargados de codificar la migración, basándose en el DAM, para luego realizar las pruebas integrales, preparación de ambiente de certificación y producción.

**d. Analista**

- Apoyará y resolverá dudas de los desarrolladores respecto al DAM.

**5.3.2. Pre-Desarrollo****5.3.2.1. Transferencia de conocimientos**

Esta parte es fundamental en el inicio del desarrollo, debido a que el analista es el que estudia todo el aplicativo legado y realiza una propuesta para el aplicativo migrado, posteriormente se comunicará con el desarrollador, y le brindará los conocimientos que contiene el DAM, además absolverá las dudas que pueda tener el desarrollador con el proyecto.

**5.3.2.2. Tecnologías de Desarrollo**

Se propone usar las siguientes tecnologías, para un mejor funcionamiento:

- **Framework:** Se usará Entity Framework (EF) porque nos permitirá trabajar con datos relacionales usando objetos específicos del dominio. Adicionalmente, tiene la ventaja de eliminar de la mayor parte del código de acceso a datos, que los desarrolladores suelen tener que escribir.
- **Servidor:** Usaremos el servidor web IIS (Internet Information Services) porque es confiable, seguro y administrable en internet. Además, que se adapta con aplicaciones de Visual Studio de Microsoft.
- **Lenguaje:** El lenguaje a utilizar será el C# de Microsoft, debido a que este ofrece soporte a comparación de los Open Source.

### 5.3.2.3. Preparación de Ambientes para Desarrollo

Disponer de herramientas de:

- Hardware (Pc con las condiciones necesarias)
- Desarrollo (Visual Studio .NET, Bases de datos SQL Server, Servidor IIS), instalado.
- Control de versiones automatizado. (TFS)

### 5.3.3. Desarrollo

#### 5.3.3.1. Diseño para Acceso Rápido

La tecnología web es muy conocida debido a su facilidad para mostrar contenidos de manera gráfica y para vincular de manera fácil documentos de diferentes ubicaciones. Cabe recalcar que, para lograr que esta tecnología sea efectiva, es necesario que los desarrolladores cumplan con

ciertos estándares para que se pueda cumplir con dos objetivos muy importantes.

- Que el aplicativo migrado trabaje rápidamente y sin dificultades técnicas en los computadores de los usuarios.
- Que el aplicativo migrado pueda ser visualizado por los usuarios de la misma manera en que sus autores las han construido.

Para lograr ambos objetivos, es necesario que quienes construyan el aplicativo de migración hagan uso de un conjunto de buenas prácticas que se han obtenido de la experiencia en la construcción de este tipo de aplicativos, y también, asegurarse de cumplir con estándares.

#### **5.3.3.2. Normas Mínimas para Facilidad el Acceso - vía conexión telefónica**

Para tener acceso a recursos Web dependemos de la transmisión de datos entre dos computadores, por lo que vemos como prioridad optimizar la cantidad de información que se envía entre ambos, de tal manera que quien la recibe pueda verla satisfactoriamente.

Lo anterior se expresa en seis áreas de recomendaciones muy concretas:

##### **a. Peso de Páginas**

El aplicativo migrado debe tener un peso máximo permitido por página que no supere una cantidad razonable de kilobytes (kb) que hagan compleja su visualización. Por lo tanto, lo razonable dependerá directamente del aplicativo que se esté desarrollando y de la conexión con la que cuente los usuarios.

Ante esto el desarrollador del aplicativo de migración debería de preguntarse si sus usuarios estarán dispuestos a esperar todo el tiempo que se demora el aplicativo en cargar completamente.

- 5 segundos para que aparezca algo visible en la pantalla.
- 10 segundos para que aparezca algo legible en la pantalla.
- 30 segundos hasta hacer un clic hacia otra parte del sitio o hacia otro sitio.

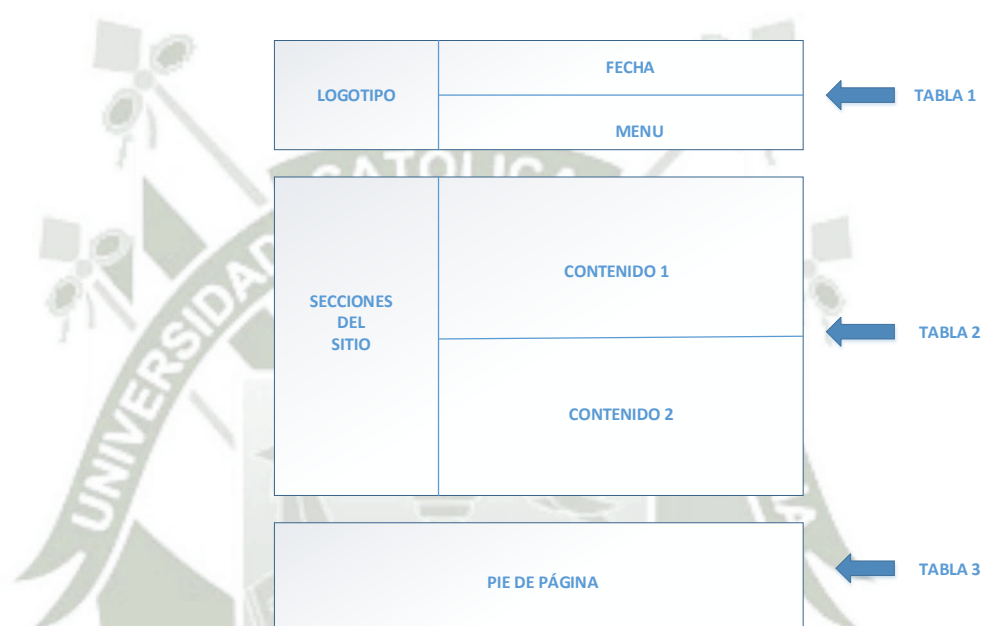
#### **b. Diagramación de las Páginas**

En la actualidad, contamos con nuevas tecnologías para la diagramación de las páginas web (como las Hojas de Cascadas de Estilo o CSS), lo normal es que los contenidos que se muestran se dispongan en tablas con el fin de que cada elemento ocupe el lugar que se le ha asignado dentro del contenido.

Recomendamos construir una estructura de presentación de los contenidos que se pueda fragmentar en varias tablas. Así, cuando el aplicativo migrado se presente en el programa visualizador del cliente, siempre mostrará la primera tabla (que normalmente llevará el logotipo y la identificación del aplicativo) de manera rápida, Brindando al usuario la sensación de haber llegado al destino elegido. Luego en las siguientes tablas se van poniendo los restantes elementos del sitio.

En la siguiente figura se puede ver que la interfaz del aplicativo está construida en tres tablas, de acuerdo al siguiente orden:

- Tabla 1: Muestra el logotipo de la institución, la fecha y el menú del sitio.
- Tabla 2: Muestra las Secciones del aplicativo además de los contenidos de diferente nivel.
- Tabla 3: Muestra el *footer* (pie de la página) con la identificación corporativa.



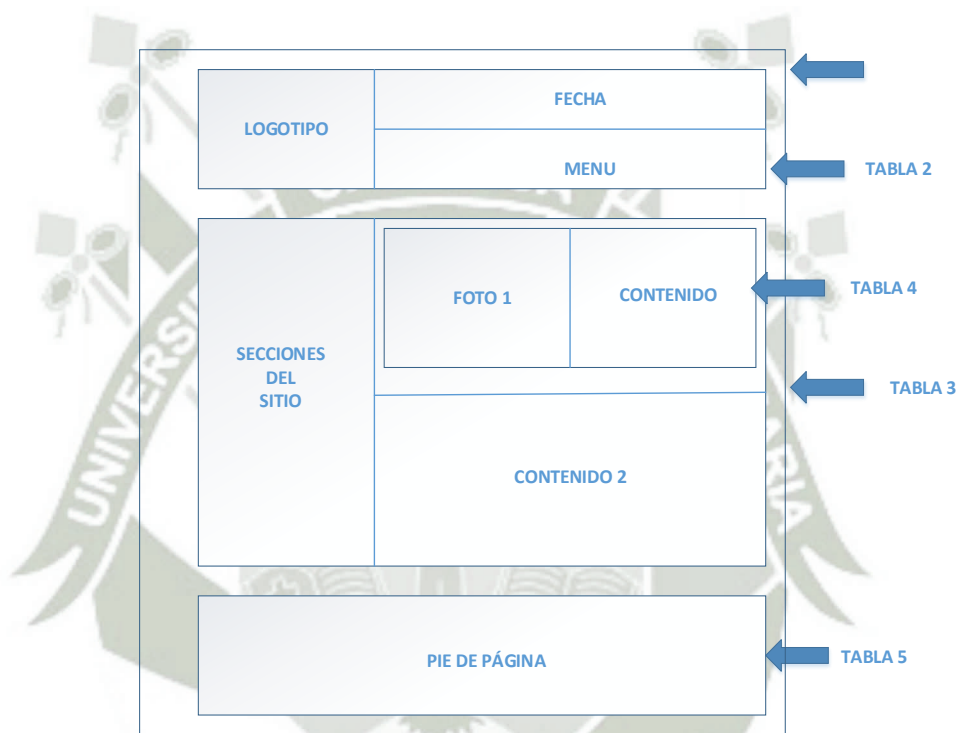
**Figura 27: Ejemplo de tablas para separar contenidos y conseguir despliegue rápido**

Estudios sobre acceso a sistemas web indican que el usuario espera que al primer segundo después de haber hecho clic sobre un enlace, éste tenga alguna reacción y note que algo está ocurriendo.

Para el caso anterior, recomendamos evitar las tablas generales que incluyen en sí mismas a otras (tablas anidadas), ya que se usará una parte del tiempo en calcular esa relación de dependencia entre las

tablas, antes de mostrar algo útil en la pantalla, lo que hará que el tiempo de carga sea más largo.

En la siguiente figura se puede ver que el aplicativo se construye en base a tres tablas interiores, que son agrupadas por una tabla general; también en la zona de Contenido 1 se dispuso una tabla que permite incluir una foto junto al contenido:



*Figura 28: Ejemplo de tablas que dificultan el despliegue rápido.*

### c. Uso de Presentaciones en Flash®

La tecnología flash era muy buena para presentar contenido multimedia interactivo, eso sí nos retomamos hace 5 años, pero en los años 2015 y 2016, podemos observar que esta tecnología es obsoleta, debido a su peso, pues para cargar cierto contenido,

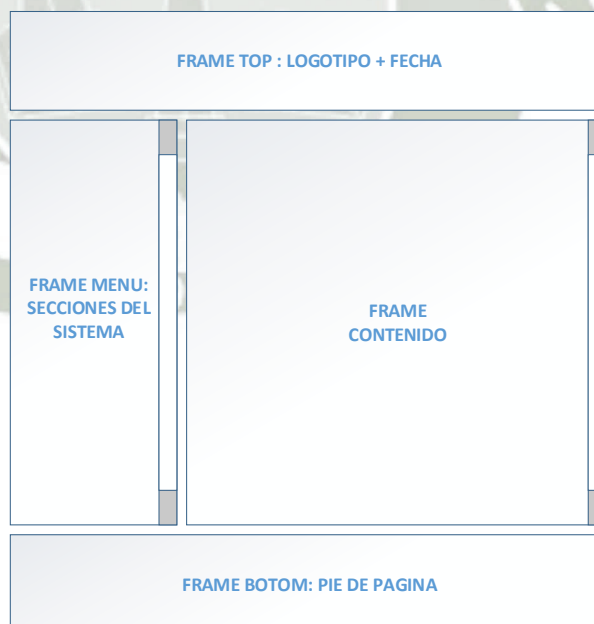
debíamos esperar demasiado y eso incomodaba al usuario, además de ciertas incompatibilidades con sistemas operativos actuales.

Ante esto, es que recomendamos usar tecnología como JQuery y Ajax, tecnologías que permiten hacer contenidos interactivos a través de código, que a su vez son más ligeros y se muestran en un tiempo más rápido.

#### d. Uso de Marcos o Frames

Los marcos o *frames* son agrupaciones de varios archivos para que se desplieguen de manera simultánea, permitiendo ver varios contenidos al mismo tiempo en la misma ventana.

Gráficamente podemos ver cómo se hace el despliegue de dichos archivos:



*Figura 29: Ejemplo de uso de frames para desplegar contenidos simultáneos*

El uso de marcos tiene sus positivos y negativos, que detallamos brevemente:

**Positivos:**

- Permite tener varios contenidos presentes todo el tiempo, como un cabezal o menú.
- Es de fácil navegación ya que el usuario nunca pierde de vista dónde se encuentra.

**Negativos:**

- Evita el poder marcar una página como favorita porque nunca se le muestra cuál es su dirección web.
- Cuando se llega a un contenido desde un enlace brindado por el buscador, verá el sitio sin los otros marcos y no sabrá cómo navegar en él.
- La visualización masiva genera una carga mayor para el usuario que se encuentra en el aplicativo; eso lo obliga a esperar a que aparezcan todos los contenidos de los archivos para poder usarlo.

Esta forma debe desecharse para pasar a sitios de interfaz contenida en un solo archivo.

#### e. Uso de Imágenes Background

Esta tecnología se puso de moda en el año 1996 cuando el navegador Netscape lo implementó, fue el uso de imágenes como fondos o *backgrounds* de las páginas web.

Esta práctica debe ser dejada de lado porque agrega un paso excesivo, afectando el tiempo de descarga y acceso a la información.

#### 5.3.3.3. Normas para Incorporar Elementos Gráficos y Multimedia

Al incorporar en el aplicativo, elementos gráficos y multimedia, debemos de seguir normas muy concretas para evitar que su peso afecte el desempeño de la página cuando sea solicitada por los usuarios.

Estas son algunas recomendaciones para asegurar la correcta inclusión de dichos elementos:

- Optimizar el peso de las imágenes: Debemos reducir el número de colores disponibles y la resolución (72 dpi es la norma).
- Elegir el formato adecuado: Si la imagen tiene el mismo tamaño, el peso varía dependiendo de si están en formato GIF o formato JPG.

Es común que una imagen con colores planos (como un icono) tendrá un peso menor si se guarda en GIF respecto de si es guardada en JPG. Lo contrario ocurrirá con una imagen con muchos colores diversos (como una foto). Se deberá de probar ambos formatos para determinar cuál es el mejor a usar.

- Ubicación de imágenes: Recomendamos tener un directorio para almacenar las imágenes repetidas, tales como los iconos y otros elementos gráficos usados en el aplicativo. Al ubicarlos en un directorio único podemos aprovechar la función de caché del programa visualizador para mejorar el rendimiento de las páginas.

#### 5.3.3.4. Interpolaría

Debido a que el aplicativo migrado se puede acceder sin problemas desde computadores que utilizan diferentes sistemas operativos, debemos de cuidar el aspecto de la diversidad de navegadores para que el contenido pueda verse sin mayores contratiempos.

Es por eso que nuestras recomendaciones son:

- Usar código HTML estándar, no mejorado para un visualizador en especial.
- Probar el sitio con las versiones para los sistemas operativos con diversos visualizadores de páginas (browsers); especialmente hacerlo con versiones de Internet Explorer, Chrome, Mozilla, Opera y Safari.
- Asegurarse de que el sitio puede ser visualizado de alguna forma cuando no se cumplen ciertas condiciones mínimas debido a la versión desactualizada del navegador del usuario.

### 5.3.3.5. Estándares Internacionales

En la web, sus tecnologías tienen un conjunto de estándares que deben ser respetados para lograr la certificación que acredita al sitio respecto de su conformidad con ellos.

La World Wide Web Consortium (<http://www.w3c.org/>) es la encargada de verificar la tecnología web y así logra que los navegadores cumplan efectivamente el lenguaje HTML.

Dentro de estos estándares, tenemos dos que van de la mano con la interfaz del contenido, los cuales son:

- **Validación de HTML:** Esta validación es presentada por la W3C (<http://validator.w3.org/>) y que permite detectar errores en la forma de usar el lenguaje HTML y XML en la construcción del sistema Web.
- **Validación de CSS:** Esta validación también es presentada por la W3C (<http://jigsaw.w3.org/css-validator/>) mediante el cual se puede validar la sintaxis de una Hoja de Estilo en Cascada (Cascade Style Sheet o CSS, en inglés), que describe la forma de presentar contenidos en una web. La tecnología CSS es que facilita la mantención de un sitio mediante la separación de la presentación (diseño) del contenido.

### 5.3.3.6. Diseño para la Accesibilidad

Este diseño va referido a la capacidad del aplicativo migrado para personas con discapacidades físicas, que les impiden usar la información disponible de una manera tradicional y por ello emplean ayudas técnicas.

Para poder verificar que nuestro Sistema de Migración cumple con las normas de accesibilidad, la iniciativa WAI (*Web Accessibility Initiative*) de la W3C (*World Wide Web Consortium*) propone la realización de las siguientes pruebas:

- Comprobar la accesibilidad utilizando herramientas automáticas y revisión humana. Los métodos automáticos son generalmente rápidos y convenientes, pero no pueden identificar todos los problemas de accesibilidad. La revisión humana puede ayudar a garantizar la claridad del lenguaje y la facilidad de navegación.
- Usar validaciones desde el inicio del desarrollo. Los problemas de accesibilidad que se identifican a tiempo son fáciles de corregir y de evitar. Entre estas validaciones tenemos:
  - Utilizar una herramienta automatizada de validación de la accesibilidad y la navegación. Se debe tener en cuenta que las herramientas o programas de revisión no contemplan todos los problemas de accesibilidad, como lo comprensible que puede ser un enlace de texto, o el contenido de un texto alternativo, etc.
  - Comprobar la sintaxis de programación del contenido con las herramientas ofrecidas por el W3C.

- Validar las hojas de estilo.
- Usar varios navegadores, antiguos y nuevos.
- Revisar la gramática y ortografía. Una persona que lee una página con un sintetizador de voz puede no ser capaz de descifrar la pronunciación que emite ese dispositivo de una palabra que tiene un error ortográfico. Eliminando los problemas gramaticales se aumenta la comprensión.
- Invitar a personas con discapacidad a revisar los documentos debido a que proporcionan valiosa información sobre la existencia de problemas de accesibilidad o usabilidad y la seriedad de la falla.

#### **5.3.3.7. Diseño de la Experiencia del Usuario**

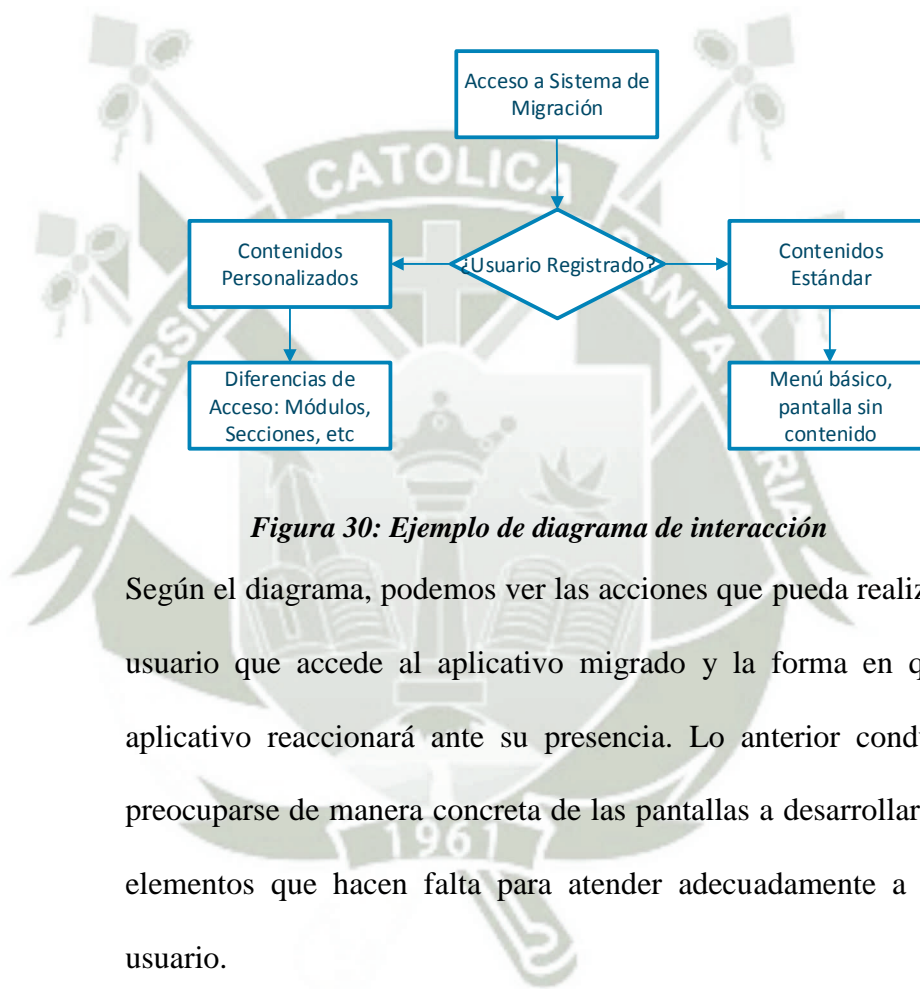
Este diseño sirve para tener en cuenta la experiencia que tendrá el usuario al navegar por el aplicativo. A esto se le denomina experiencia del usuario y el objetivo siempre será el de que cada persona que visite el aplicativo encuentre lo que está buscando de manera simple, de tal manera que regrese al sitio y les cuente a otros sobre su contenido y funcionalidades. Para conseguirlo es necesario hacer un trabajo muy acabado en lo que se refiere a la planificación y organización de los contenidos, como también definir cómo se van a mostrar y operar las funcionalidades.

##### **a. Desarrollo de diagramas de iteración**

Para poder asegurar que la experiencia del usuario se está resguardando adecuadamente, es recomendable la generación

los diagramas de interacción mediante los cuales podemos representar gráficamente las posibilidades de acción que tiene un usuario enfrentado a tomar una decisión en un aplicativo.

Por ejemplo, el siguiente diagrama muestra las posibilidades de reacción que tiene un Sitio Web ante el ingreso de un usuario registrado en un sitio:



**Figura 30: Ejemplo de diagrama de interacción**

Según el diagrama, podemos ver las acciones que pueda realizar un usuario que accede al aplicativo migrado y la forma en que el aplicativo reaccionará ante su presencia. Lo anterior conduce a preocuparse de manera concreta de las pantallas a desarrollar y los elementos que hacen falta para atender adecuadamente a dicho usuario.

#### **b. Prueba de sistemas de interfaces**

Antes de iniciar la denominada puesta en marcha del aplicativo, es muy conveniente hacer una serie de pruebas que permita asegurarse, antes de la construcción del código, que los usuarios

van a entender la forma en que está organizada la información y los contenidos y funcionalidades que se están ofreciendo a través del aplicativo. Es por eso que recomendamos las siguientes pruebas:

### ***Pruebas Heurísticas***

Son análisis realizados por expertos respecto a las interfaces que se está ofreciendo el aplicativo.

Las más usadas fueron creadas por Jakob Nielsen, son diez pautas de evaluación que las nombramos a continuación:

- **Visibilidad del estado del aplicativo:** Permite saber si el usuario conoce el comportamiento del aplicativo. Se revisa si existen los diferentes elementos que ayudan a esto:
  - Indicación gráfica de donde se encuentra (ruta de acceso desde portada)
  - Indicación de que ha visto (marcar los enlaces visitados)
  - Indicación de que hay un proceso en marcha (anunciando estado de avance...)
  - Indicación de cuántos pasos faltan para terminar (como en el caso de que ya a un proceso de registro en el Sitio Web)
- **Similitud entre el aplicativo y el mundo real:** Permite medir si el aplicativo es comprensible para el usuario.

- **Control y libertad del usuario:** Permite medir si los usuarios que se equivocan al hacer algo tienen forma de recuperarse de esos errores. Se revisa si existen formas de hacerlo. Por ejemplo: ¿Se puede deshacer una operación? ¿Se puede rehacer una operación?
- **Consistencia y cumplimiento de estándares:** Permite medir el cumplimiento de los estándares web que se usan en el aplicativo. Para ello se debe validar y revisar el sitio con las herramientas que se ofrecen en <http://www.w3c.org> para HTML y CSS.
- **Prevención de errores:** Asegura el ingreso de la información por parte del usuario, permite evitarle errores. Para ello, se verifica si en las áreas en que los usuarios deben interactuar con el aplicativo, se les explica claramente lo que se espera de ellos. Por ejemplo:
  - Uso de Javascript para validar formularios: para que todos los campos obligatorios sean llenados.
  - Uso de elementos destacados en los formularios: indicar los campos obligatorios con asteriscos (\*) o, bien, campos obligatorios marcados con color.
- **Preferencia al reconocimiento que a la memorización:** Verifica si el aplicativo ayuda al usuario a recordar cómo se hacía una operación, o bien le obliga a aprenderse los pasos cada vez que ingresa a través de una navegación coherente.

- **Flexibilidad y eficiencia de uso:** Revisa si se brindan soluciones diferentes de acceso a los contenidos, a los usuarios novatos respecto de los expertos. Por ejemplo, se puede contar con botones para los primeros y atajos de teclado para el experto.
- **Estética y diseño minimalista:** Cada elemento que se muestre en pantalla debe tener su razón del porqué debe estar ahí. Se comprueba la existencia de elementos irrelevantes (texto, sonido e imagen), que no aportan ni ayudan a que el usuario distinga lo importante de lo superfluo. Para ello se verifica la existencia de:
  - Jerarquías visuales: que permiten determinar lo importante con una sola mirada.
  - Tamaño de imágenes: que no afectan la visión general de la información del Sitio Web; se verifica tanto tamaño como peso.
- **Ayuda ante errores:** Va más enfocado a ayudar en problemas a través una página tanto *online* como *offline*; entre los elementos que se miden se cuentan:
  - Mensaje 404 personalizado, con el fin de ofrecer una información y navegación alternativa cuando una página no es encontrada.

- Mensaje de falla ofrece una alternativa offline (teléfono, mesa de ayuda) que permite que el usuario mantenga su confianza en la institución.
- **Ayuda y documentación:** El aplicativo debe brindar ayuda de acuerdo al lugar en que el usuario esté visitando; también se revisa la existencia de módulos de búsqueda que permita al usuario encontrar los elementos de ayuda que sean relevantes de ofrecer (preguntas frecuentes; páginas de ayuda).

### *Pruebas de Usabilidad*

Encierran una serie de elementos que sirven para poder establecer si cumplen con las necesidades de los usuarios del aplicativo.

Son pruebas realizadas con usuarios, para poder de determinar si la organización de los contenidos y las funcionalidades que ofrece el aplicativo son entendidas y usadas por los usuarios de manera simple y directa.

Las pruebas más conocidas son:

- **Prueba Inicial:** Sirve para determinar el funcionamiento de la organización de contenidos y elementos iniciales de diseño (botones, interfaces). El material con que se prueba es una imagen dibujada del Sitio Web.
- **Prueba de Boceto Web:** Sirve para determinar el entendimiento la navegación, si se pueden cumplir tareas y

si el usuario entiende todos los elementos que se le ofrecen.

El material con que se prueba es una maqueta web semi-funcional.

Los resultados de cada una de esas etapas permitirán adecuar los elementos con los que se esté trabajando en esos momentos, con el fin de atender a los usuarios y ofrecerle una experiencia a la altura de sus expectativas.

Es de gran importancia la realización de estas pruebas, ya que generan gastos que serán muy útiles y permitirán darse cuenta a tiempo de errores conceptuales en la entrega de la información, que puedan ser enmendados tempranamente y sin afectar el desarrollo total del proyecto.

#### **5.3.3.8. Como Atender a los Usuarios**

El aplicativo migrado permitirá la interacción y comunicación entre los usuarios, es por eso que se ha diseñado esta sección para tratar directamente el tema. Cabe recalcar, que la tecnología web se orienta a brindar niveles de comunicación muy avanzados, de los cuales se espera respuestas rápidas e interacción permanente.

Para eso es que recomendamos brindar *feedback* o retroalimentación para el usuario, usando las siguientes herramientas:

- Sistemas de Correo Electrónico.
- Sistemas de Encuestas o Votaciones.
- Sistemas de Chat.

- Sistemas de Simulación.

#### a. Sistemas para Generar Feedback

Como se planteó antes, un sitio web tiene diversos aplicativos para recibir *feedback* o retroalimentación de parte del usuario, entre los que destacamos:

- Sistemas de Correo Electrónico
- Sistemas de Encuestas o Votaciones
- Sistemas de Simulación
- Sistemas de Búsqueda

#### b. Formularios del Sitio

Al crear nuestro aplicativo, es muy relevante cumplir con ciertas prácticas de usabilidad, que hagan más sencilla la operación de los formularios. Por ejemplo:

- Validación de campos obligatorios
- Validación del e-mail
- Valores por defecto.
- Claridad de botones

#### 5.3.4. Aplicar Pruebas Unitarias Especificas

Estas pruebas unitarias ayudarán al desarrollador a percatarse de sus errores y tratar de enmendarlos antes de enviarlos al ambiente de certificación. Además de las pruebas mencionadas deben de realizar los siguientes tipos de prueba:

- **Compatibilidad:** Permite verificar si el aplicativo se adapta bien al Sistema Operativo al igual que sus componentes.
- **Prueba de validaciones:** Es aquella prueba que consiste en verificar si las lógicas efectuadas en la codificación del aplicativo son correctas.
- **Tiempo de ejecución:** Es el tiempo que tarda en ejecutar cada proceso del aplicativo.
- **Tiempo de respuesta:** Es el tiempo que tarda en recibir respuesta a las consultas por parte del aplicativo.
- **Casos de Prueba de analista:** Realizar las pruebas propuestas por el analista en el DAM, estas deberán estar en estado conforme.

#### 5.3.5. Entregables

- **MIM - Manual de instalación de Migración:** El manual de instalación de migración está dirigido al personal de soporte técnico responsable de la instalación y configuración del aplicativo migrado, este manual contiene todas las indicaciones para la instalación del aplicativo y configuraciones necesarias. Este Manual se encuentra en el [Anexo 12].
- **PUE – Pruebas Unitarias Específicas:** El PUE, es generado en la fase de análisis, y estos son ejecutados en la fase de construcción; este documento tiene que ser actualizado adjuntando la evidencia de que las pruebas unitarias fueron ejecutados correctamente.
- **PDI: Plan de implementación:** Es la agenda de planeamiento para el pase a producción a nivel administrativo, donde se indicará los

involucrados, procesos y responsables para realizar la implementación que se indica en el MIM (Manual de Instalación de Migración).

En este documento vemos que para hacer una prueba de cómo va el aplicativo se realizará primeramente un plan piloto, es decir lanzar la aplicación en un sector que no afecte mucho al negocio para que luego de ser usado, se pueda ver si este aplicativo tiene fallas o no. En caso de tenerlas se pone la aplicación legada nuevamente mientras que el nuevo aplicativo es corregido.

Realizado este plan piloto con resultados exitosos, se realizará la implementación del aplicativo con tecnología moderna en todas las partes donde éste se utilice, para poder ejecutarlo con normalidad. El PDI se encuentra en el [anexo 13].

- **ACR - Acta de Reunión:** Es el acta donde se archivan todas las reuniones que se realizaron en el proceso, quienes asistieron, cuanto duró, etc. Para tener archivadas dichas reuniones a manera de control interno.
- **Código fuente:** Paquete completo de toda la codificación de la migración.
- **Publicación de la Migración:** Archivos que contemplan el aplicativo migrado para que pueda ser usado al momento de realizar la instalación.
- **Bloque de Preguntas:** Se actualizan las evidencias de consultas al equipo legado, evidenciando con los correos correspondientes.
- **Check List de MIM:** Documento donde se encuentran criterios de validación para una correcta ejecución posteriormente. [Anexo 14]
- **Check List de Programación:** Lista de criterios que se toman en cuenta para poder validar el código fuente usado para la migración. [Anexo 15]

### 5.3.6. Control de Calidad

- **CCI – Control de calidad interno:** Para el control de calidad interna es necesario que los entregables estén conformes, luego se elegirá a un analista del proyecto para que revise si lo desarrollado cumple con los requisitos solicitados en el DAM, los entregables cumplan con lo necesario y este control será registrado en un documento CCI.
- **CCE – Control de calidad externo:** Para el control de calidad externa es necesario que el documento CCI se encuentre en estado conforme, posteriormente se elegirá a un analista externo que esté a cargo de otro proyecto, para que revise si lo desarrollado cumple con los requisitos solicitados en el DAM, y este control será registrado en un documento.

### 5.3.7. Proceso de Congelamiento

Una vez que se concluya con el control de calidad en desarrollo, se procede a realizar el proceso de congelamiento, que consiste en realizar la copia de todos los entregables a una carpeta del ambiente de desarrollo, para que luego en la fase de certificación se realice la copia de todos los entregables al ambiente de certificación.



**Figura 31: Congelar entregables en el servidor de desarrollo**

#### 5.4. Certificación

Es la cuarta fase del proceso de migración, la cual se enfoca en certificar correctamente el aplicativo hasta que este se encuentre sin ocurrencias y sin errores, en caso de tener errores, estos se informaran a los encargados de construcción para que sean enmendados.



**Figura 32: Metodología de migración - Certificación**

Como ya lo mencionamos anteriormente, esta etapa también cuenta con su propio ambiente de trabajo, para la realización de la certificación.



**Figura 33: Esquema del ambiente de certificación**

Esta acción será llevada básicamente a cabo por el validador de contenido y desarrolladores para la verificación del aplicativo migrado, para ver si cumple los requisitos definidos en las especificaciones técnicas.

#### 5.4.1. Involucrados

##### a. Líder de proyecto:

- Es el responsable de verificar conjuntamente con los desarrolladores el proceso migratorio, los registros de errores serán almacenados en el CDD (Control de defecto).
- Conservar el control del resultado de estas reuniones.
- Informar sobre el estado actual del proyecto en certificación.
- Suministrar al equipo el informe de la certificación.

##### b. Líder experto – Legado:

- Es el encargado de supervisar la labor del validador de contenidos.
- Comunicar al líder de proyecto los casos atípicos que se presenten en la certificación

- c. **Experto de Contenidos:** Es el responsable conjuntamente con el validador de contenidos, que los casos de pruebas se lleven con normalidad.
- d. **Validador de contenidos:**
- Grupo de personas que realizarán la certificación del aplicativo migrado.
  - Reportar las ocurrencias y errores que se presenten en la certificación plasmando en el documento CDD (Control de defecto). [Anexo 16]
- e. **Desarrollador:**
- Corrige a la brevedad las ocurrencias indicadas por el validador de contenidos.
  - Realizar el proceso de congelamiento de las correcciones.
- f. **Soporte técnico:** Es el encargado de llevar a cabo el proceso de instalación siguiendo los procedimientos indicados en el MIM en el ambiente de certificación.

#### 5.4.2. Pre-Certificación

Antes de iniciar la fase de certificación, es necesario la creación del documento de pruebas legados (DPL); este documento será realizado por el experto legado, en compañía del validador de contenidos, este contendrá las pruebas funcionales que se aplicaran al aplicativo migrado. [Anexo 17]

En esta fase, el personal de soporte técnico legado recibe el MIM (Manual de Instalación de Migración) por parte de los desarrolladores. Luego se realizará

la instalación del aplicativo en base a los pasos indicados en este manual, para así dar pase a la certificación que realizará el validador de contenidos.

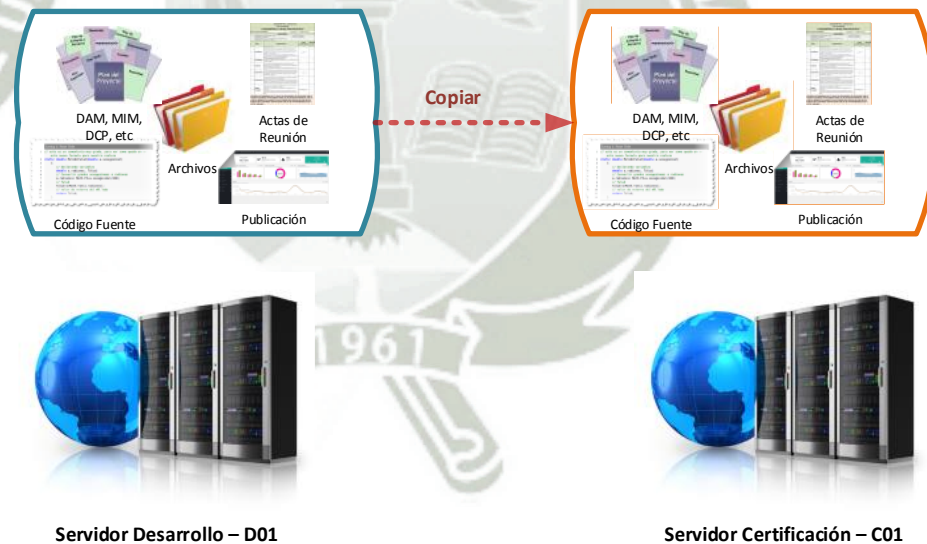
**a. Tráferencia de Conocimientos**

Los desarrolladores darán a conocer la ruta en donde se encuentran congelados los entregables.

Los desarrolladores serán los encargados de reunirse con el personal técnico legado para brindarles el contenido paso a paso el MIM.

**b. Ejecución de MIM**

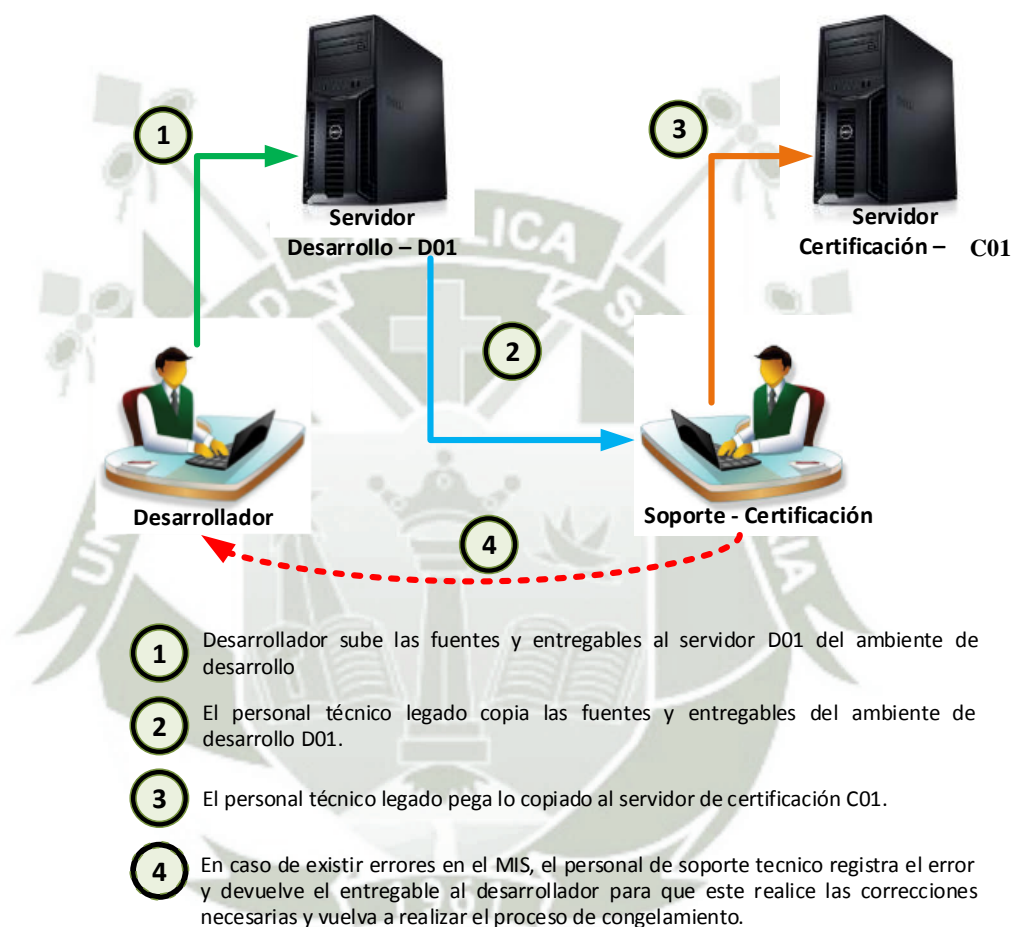
Lo primero que se debe realizar, es el descongelamiento de los entregables congelados en el ambiente de desarrollo, al ambiente de certificación.



**Figura 34: Proceso de descongelamiento en certificación**

Para realizar la instalación del aplicativo, el personal técnico deberá contar con el IIS instalado en el servidor del ambiente de certificación.

Luego realizar todos los puntos descritos en el MIM, en caso de existir algún error, describir dicho error presentado en el MIM e indicar a los desarrolladores, para que luego los desarrolladores vuelvan a congelar el MIM con las correcciones necesarias versionando el documento y volver a realizar la instalación del MIM.



*Figura 35: Proceso de congelamiento y reversión*

#### 5.4.3. Inicio de Certificación

Para realizar la certificación, segmentaremos este proceso en 5 áreas:

- Pruebas de Interfaces y Contenidos
- Pruebas de Funcionalidades y Operación

- Pruebas de Carga
- Pruebas de Seguridad
- Pruebas de Respaldo y Recuperación

#### 5.4.3.1. Prueba de Interfaces y Contenidos

Esta etapa consiste en realizar revisiones precisas de la forma en que se despliegan las ventanas del sitio y ver si cumplen con los estándares mínimos que se hayan definido como meta a ser cumplida.

En esta etapa consideraremos las siguientes acciones:

##### a. Verificación de Contenidos

Esta prueba revisa si el aplicativo migrado desarrollado incluye todos los contenidos que se han especificado en el DPL (Documento de Pruebas Legado). Se puede hacer en forma manual o automática, de acuerdo a las siguientes orientaciones:

**Sistema Manual:** Hace referencia a una revisión manual de los contenidos del aplicativo migrado a través de la navegación de sus ventanas. Para ello se recomienda primero construir un índice de contenidos y luego verificar cada uno de los ítems que contiene, y realizar un recorrido exhaustivo del aplicativo. Los elementos que deben probarse obligatoriamente son:

- Verificación de ortografía y redacción
- Verificación de enlaces principales
- Verificación de imágenes en páginas

- Verificación de existencia de archivos adjuntos
- Verificación de la Lista de Chequeo de Accesibilidad

**Sistema Automático:** Especialmente orientado a la verificación de enlaces rotos, lo cual se puede hacer utilizando software especializado.

**Sistemas Basados en Internet:** Se recomienda usar el servicio Check Link (<http://validator.w3.org/checklink>) del estándar W3C para asegurar la alta calidad técnica y editorial, así como obtener un mayor apoyo desde el W3C y desde la comunidad en general.

#### b. Verificación Estándar

El aplicativo migrado debe cumplir ciertas normas de organización de su código fuente (sintaxis), que permitan su visualización por software equivalente en diferentes plataformas. Dicha sintaxis está estandarizada y puede ser probada a través de herramientas públicas que están disponibles en Internet. Las dos más importantes son:

**Validación de HTML:** Esta validación es realizada por el *World Wide Web Consortium* (<http://validator.w3.org>) e indica si el código usado en el aplicativo migrado es correcto. Como resultado entrega un reporte con los eventuales errores para ayudar a su reparación.

**Validación de CSS:** Esta validación también es realizada por el *World Wide Web Consortium* (<http://jigsaw.w3.org/css-validator/>)

e indica si la Hoja de Cascada de Estilo (*Cascading Style Sheet*) cumple con la sintaxis estándar y por lo tanto podrá ser visualizada correctamente en el aplicativo migrado.

### c. Verificaciones de Interfaces

A través de esta prueba revisamos los aspectos gráficos del aplicativo migrado, para determinar si su despliegue en las páginas es correcto. Dentro de los elementos más importantes a ser verificados, se incluyen los siguientes:

- Plug-ins necesarios
- Consistencia de la Diagramación
- Ancho de la Diagramación
- Imágenes Sin Atributo ALT

### 5.4.3.2. Prueba de Funcionabilidades y Operación

Esta etapa consiste en realizar chequeos completos respecto a las funcionalidades y aplicaciones que ofrece el aplicativo de migración, ya sean de aplicaciones simples como formularios hasta más complejos, como consultas y modificaciones de registros en base de datos.

En este sentido, las pruebas se deben hacer sobre diferentes elementos, siendo algunos de los más importantes los siguientes:

#### a. Validación de Formularios

Si el aplicativo de migración tiene formularios para el envío o ingreso de datos, debemos utilizar sistemas de validación del

ingreso de datos para asegurar que éstos sean bien ingresados. En este aspecto, algunas de las validaciones más importantes deben ser las siguientes:

- Campos Obligatorios
- Sintaxis de Ingreso
- Ingreso de Datos
- Reingreso y Corrección de Datos
- Elementos de Interface
- Multiplataforma
- Sistemas de Búsqueda

#### **5.4.3.3. Prueba de Carga**

La carga de trabajo hace referencia a la capacidad máxima que tiene un servidor web (hardware y software), para atender a un conjunto de usuarios de manera simultánea.

Es por eso que, las actividades de esta etapa tienen relación con comprobar, de manera anticipada, el funcionamiento que tendrá el servidor del aplicativo migrado cuando esté en plena operación.

Estas pruebas tienen la finalidad de simular una carga de trabajo similar y superior a la que tendrá cuando el aplicativo esté funcionando, con el fin de detectar si el software instalado (programas y aplicaciones) cumple con los requerimientos de muchos usuarios simultáneos y también si el hardware (servidor y el equipamiento computacional de redes y enlace que lo conecta a Internet) puede soportar la cantidad de visitas esperadas.

Existen muchos softwares en el mercado que están orientados a este tipo de simulaciones, todos los cuales ofrecen características similares. Entre los datos más relevantes que es posible obtener se cuenta:

- Tiempo de acceso de los usuarios a los datos
- Volumen de datos y ancho de banda utilizado
- Archivos solicitados y tiempos usados en transferencia de datos
- Tiempo de espera de los usuarios tras hacer un clic
- Tiempo de respuesta a clics de usuarios
- Niveles de error existentes tras clics de usuarios

Como se puede apreciar del listado anterior, los reportes que se obtienen a través de esta vía se refieren a tiempos de acceso que tienen los usuarios que acceden al aplicativo migrado y la degradación que ocurre en los servicios cuando aumenta el volumen de visitantes concurrentes.

#### **5.4.3.4. Prueba de Seguridad**

Para realizar las pruebas de seguridad en el aplicativo migrado nos orientaremos a varios ámbitos, como se describe a continuación. Los temas a tratar son los siguientes:

- Manejo de DNS
- Protección de Estructura Interna del aplicativo de migración
- Protección contra Robots
- Manejo de Privacidad
- Canales Seguros
- Mecanismos de Control de Acceso

- Protección de Código
- Roles Mínimos a Asegurar

#### 5.4.3.5. Prueba de Respaldo y Recuperación

El respaldo de la información del aplicativo migrado se refiere a copiar el contenido completo del aplicativo (datos, programación, imágenes, etc.) a un medio que sea confiable, que esté en un lugar seguro y que permita la recuperación de manera rápida y eficiente.

Ente respaldo nos servirá para contingencias que puedan presentarse, hay que preocuparse no sólo de probar la confiabilidad del aplicativo al momento de respaldar, sino también para la acción de recuperar y volver a instalar lo respaldado.

#### 5.4.4. Registro de Control de pruebas y Errores

Para validar una prueba, esta debe ser lo más documentada posible, con el fin de que, quien deba efectuar la corrección, pueda replicar el error para analizarlo y luego proceder a tomar medidas correctivas. Para ello se llevará una planilla de control de defectos – CDD en que se vayan anotando por columna los siguientes datos:

- a. Detección del Error:** Para ser anotado por quien prueba.
  - Módulo: Indica la sección en la que se produce el error.
  - URL: Dirección de la página donde ocurrió el error.
  - Acción: Indicar la secuencia de pasos que siguió para que ocurra el error.

- Lo que hace o dice: Es la explicación más detallada posible del error, en particular señalando la secuencia de pasos seguida hasta dar con el error.
  - Lo que debe hacer o decir: Se debe indicar lo que se espera que debería ocurrir cuando se hace la acción que se ha descrito.
  - Encontrado por: Nombre de quien prueba.
  - Fecha: Fecha en la que se hace la anotación.
  - Reproducible: Indicar si el error se repite al hacer nuevamente la prueba.
  - Clasificación: Permite definir el grado de complejidad del error, al señalar si afecta el funcionamiento del sitio (caso extremo) o sólo su presentación.
- b. Diagnóstico del Error:** Para ser anotado por quien corrige.
- Causa: Motivo por el cual se produce el error.
  - Efectos laterales: Indicar en qué otros módulos la existencia de este error puede estar causando impacto negativo; muchas veces errores diversos tienen una causa común, por lo que al reparar ésta se arreglan los demás.
- c. Corrección del Error:** Para ser anotado por quien corrige.
- Descripción: Acción realizada para hacer la reparación del error.
  - Archivos intervenidos: Archivos en los que se hicieron modificaciones o, al menos, el principal de ellos.
  - Corregido por: Persona que hizo la corrección.
  - Fecha corrección: Fecha en que quedó reparado el error.

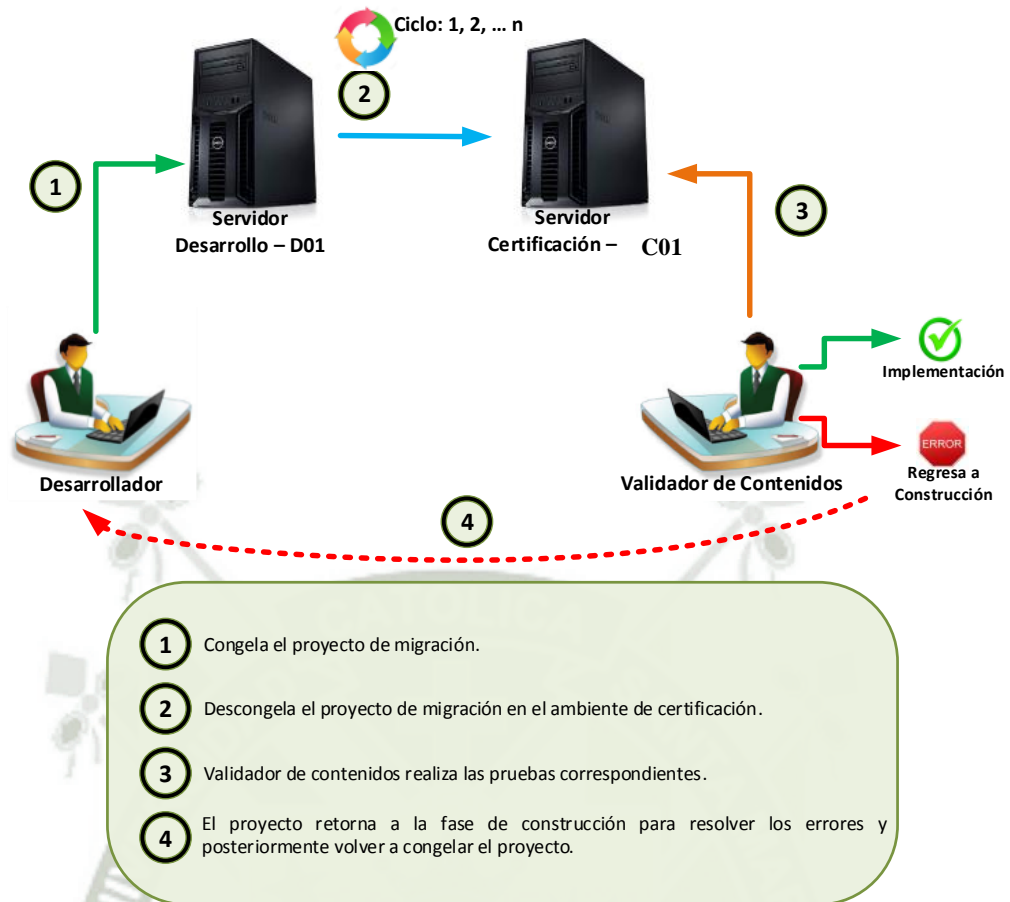
**d. Comprobación de la Corrección:** Para ser anotado por quien revisa la corrección realizada.

- Revisor: Nombre de quien revisa si el error fue efectivamente reparado.
- Fecha: Fecha en que se realiza la revisión.
- Reparado: Indicar si está reparado o no. Si no lo está, se debe copiar la línea de error en blanco en una nueva planilla, con el fin de solicitar nuevamente el proceso de corrección.

#### 5.4.5. Ciclos de Prueba

Cuando el proyecto de migración ingresa por primera vez a la fase de certificación, comienza el ciclo uno, de no existir errores, el proyecto de migración pasa a la fase de implementación con un ciclo.

En caso de existir errores, estos serán registrados en la plantilla de control de defectos (CDD). El proyecto de migración regresará a la fase de construcción para resolver los errores presentados, luego se procederá a realizar el congelamiento debido para nuevamente ingresar a la fase de certificación, en este caso ingresaría al ciclo 2. De presentarse más errores, se volverá a aplicar la misma lógica y aumentará el número de ciclos.



*Figura 36: Ciclos de prueba*

#### 5.4.6. Entregables

- **CDD- Control de defectos:** Es el control donde se tienen registrados todos los errores o fallas detectados en la migración, para su posterior solución. Este documento es realizado por parte del validador de contenidos, siendo responsable el experto del aplicativo legado.

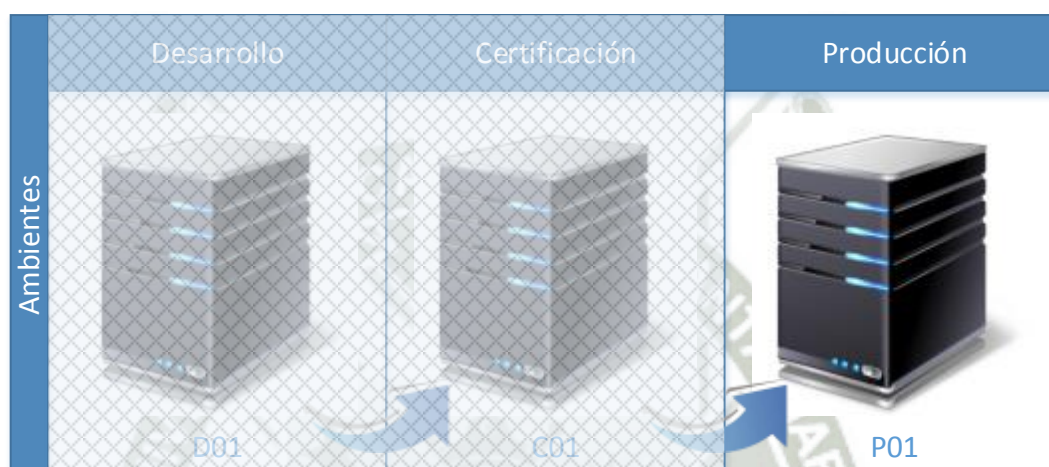
#### 5.5. Implementación

Es la quinta y última fase del proceso de migración, la cual se enfoca en el pase a producción. Para llegar a ésta fase, el aplicativo migrado ya deberá haber pasado adecuadamente el proceso de certificación, luego de ello, el aplicativo será

puesto en marcha, mediante un plan piloto para luego determinar la permanencia de al 100% del aplicativo migrado.



*Figura 37: Metodología de migración – Implementación*



*Figura 38: Esquema de ambiente de producción*

### 5.5.1. Involucrados

- **Líder de proyecto:** Es el responsable del equipo de migraciones.
- **Líder experto – Legado:** Es el encargado de supervisar las pruebas del plan piloto, para luego replicar el aplicativo migrado en toda la empresa.
- **Experto de Contenidos:** Es el responsable conjuntamente con el validador de contenidos, de que los casos de pruebas se lleven con normalidad en producción.
- **Validador de contenidos:** Grupo de personas que realizarán las pruebas más simbólicas en el aplicativo migrado reportando las ocurrencias y

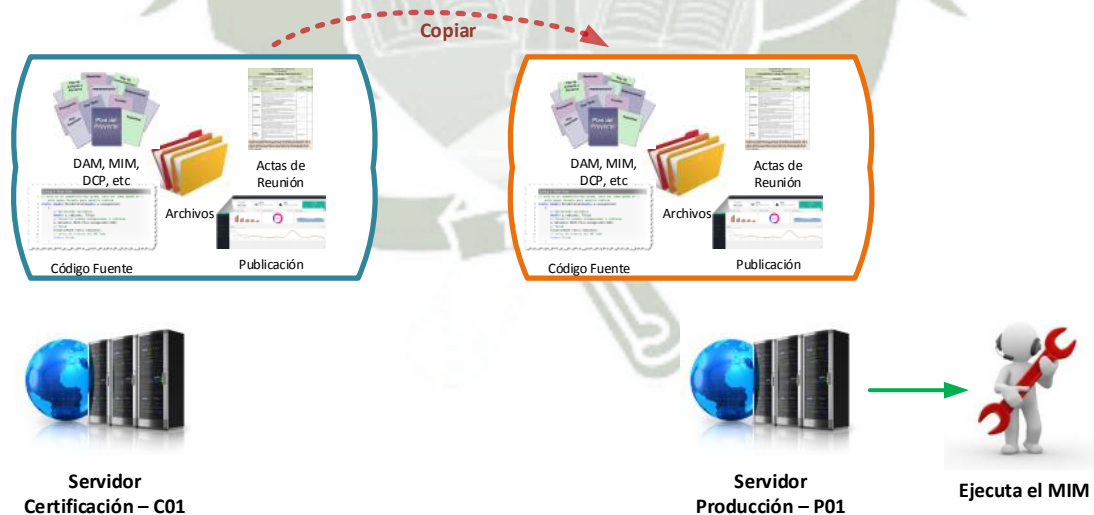
errores que se presenten, dichas pruebas se llevarán a cabo en un plazo no mayor a 2 horas.

- **Líder Experto – Migración:** Verificar que la instalación del aplicativo migrado se lleve con normalidad.
- **Soporte técnico:** Es el encargado de llevar a cabo el proceso de instalación siguiendo los procedimientos indicados en el MIM en el ambiente de producción.

### 5.5.2. Pre-Lanzamiento

#### a. Instalación

Primero se realiza el descongelamiento de los entregables del ambiente de certificación a producción. La instalación consiste únicamente en ejecutar el MIM (manual de instalación de migración) en el ambiente de producción.



*Figura 39: Proceso de descongelamiento en producción*

Para realizar la instalación del aplicativo, el personal técnico deberá contar con el IIS instalado en el servidor del ambiente de producción. Luego ejecutar todos los puntos descritos en el MIM.

#### **b. Pruebas Rápidas**

Con la experiencia que tiene el validador de contenidos en el aplicativo migrado, realizará pruebas en el ambiente de producción a la brevedad posible, por ejemplo, revisar los siguientes puntos:

**Cumplir Listas de Chequeo:** Se deben de haber cumplido las pruebas que se realizaron en la fase de certificación.

**Cumplir con la funcionalidad de los módulos:** Para esto se deben recrear ejemplos de procesos simples.

**Generación de Reportes:** Revisar que los reportes generados den como resultado las columnas y datos correctos.

**Generación de imprimibles:** Revisar que los documentos impresos se encuentren en el formato correcto.

**URL Simple:** La url del sitio debe ser simple, de tal manera que sea fácil comunicarla y además recordarla, para facilidad del usuario.

#### **5.5.3. Lanzamiento de Plan Piloto**

Antes de implementar permanentemente el aplicativo migrado, se deberá realizar el plan piloto, para esto se debe seleccionar un usuario piloto para que trabaje únicamente con el aplicativo migrado, en caso de existir un error, reportarlo al equipo de migraciones y continuar el proceso en el aplicativo

legado. Luego de concluir con el plan piloto, se procederá a eliminar el aplicativo legado para que el aplicativo migrado sea puesto en marcha para toda la institución.



**Figura 40: Esquema del Plan Piloto**

Para lograr el plan piloto del aplicativo migrado debemos de verificar que se haya cumplido adecuadamente las pruebas antes descritas, con el fin de que todos los contenidos prometidos estén bien incorporados y las funcionalidades realicen todo aquello que se describe respecto de ellas.

Si hay contenidos o funcionalidades que aún están como no disponibles para el momento en que se desea hacer el lanzamiento del aplicativo migrado, se recomienda eliminarlos en ese momento e incorporarlos cuando estén listos, en lugar de dejarlos en el sitio y que den una mala imagen sobre el mismo.

#### 5.5.4. Puesta en Marcha

Una vez concluido el plan piloto, se procederá a guardar un *backup* del aplicativo legado, para luego deshabilitarlo. Seguidamente se expandirá el

conocimiento del aplicativo migrado para ser usado por todos los usuarios, en caso existan dudas se procederá una capacitación a los usuarios.

Si existen algún tipo de mejoras al aplicativo migrado, se procederá a realizar un nuevo proyecto, que serán tomados como requerimientos.

### 5.5.5. Lecciones Aprendidas

#### a. Proponer

En este punto es que recolectamos muchas recomendaciones extraídas a lo largo del desarrollo de la migración (planificación, análisis, construcción, certificación o implementación), y de esta forma poder armar una bitácora con las mejores prácticas para realizar una migración, y de esta manera permitir asegurar una buena experiencia de los usuarios finales.

Estas lecciones aprendidas crecerán a medida de la antigüedad del aplicativo, ya que debemos tomar en cuenta que los sistemas de programación de hoy en día, no son compatibles al 100% con distintos componentes o forma de programación usados en los aplicativos legados; es por esta razón que es muy importante este punto, y registrarlo en el formato BUP (Buenas Practicas). Anexo [20]

Nota: Las lecciones aprendidas no solo son para problemas técnicos, sino también para problemas en la gestión del proyecto.

## b. Revisar

En caso se tenga algún problema al desarrollar la migración, primero se debe buscar en la bitácora de buenas prácticas para ver si existe una solución rápida, en caso no exista la solución, buscar en otros medios y seguidamente registrar dicha solución como una lección aprendida en el formato BUP.

### 5.5.6. Entregables

- **APP - Aceptación de Plan Piloto:** El líder experto legado, dará la aceptación de que el plan piloto fue realizado con éxito, y solicita expandir el aplicativo migrado a toda la institución. [Anexo 18]
- **FDP – Fin de Proyecto:** El líder proyecto junto al gerente general legado, realizarán el documento de fin de proyecto, el cual finalizara con el proyecto de migración. [Anexo 19]
- **BUP – Buenas Prácticas:** Al finalizar el proyecto, el desarrollador debe de llenar el formato BUP, donde indique las dificultades que se tuvo al momento de realizar la migración, con la solución propuesta. [Anexo 20]

## CAPÍTULO VI: ANÁLISIS Y DISCUSIÓN DE RESULTADOS

La metodología propuesta para la migración de aplicativos, se realizó en una empresa arequipeña, que se dedica a la venta de electrodomésticos, tecnología, productos para el hogar, deportes, y ropa en general; dicha empresa cuenta con más de 15 tiendas a nivel nacional.

La empresa en mención cuenta con el área de tecnologías de información, área donde se realizó la migración del aplicativo de Comisiones, aplicativo que se encarga de imponer metas para las ventas, créditos, garantías, etc. y de esta manera los vendedores pueden alcanzar un bono extra a parte del sueldo pactado con la empresa.

El aplicativo de Comisiones se encuentra en funcionamiento hace más de 10 años; al principio fue desarrollado como una ayuda para conocer las comisiones de cada vendedor, sin embargo, al transcurrir el tiempo se convirtió en un aplicativo vital para la empresa.

Para el desarrollo de la metodología propuesta, se tiene como aplicativo legado al “*aplicativo de Comisiones*”, que cuenta con las siguientes características:

- Se encuentra desarrollado en Visual Basic 6.0.
- Es un aplicativo de escritorio.
- El aplicativo solo puede ser usado en una computadora que cuente con el sistema operativo Windows y de preferencia la versión XP.
- El aplicativo solo puede ser usado en la intranet de la empresa.

Por las características mencionadas, se procedió a realizar una migración del aplicativo comisiones usando la metodología propuesta en la presente tesis.

## **6.1. Resultados por parte del Área de Tecnologías de Información**

### **6.1.1. Consideración para la Evaluación de la Metodología**

Se decidió usar la técnica de Muestreo por Conveniencia, ya que es menos costoso, se da en poco tiempo y es más conveniente para el presente caso, determinando que la muestra constará de 18 personas involucradas con el desarrollo de la migración de aplicativo comisiones.

### **6.1.2. Instrumentos para la Evaluación de la Metodología**

La encuesta será el instrumento que se utilizará para la recolección de información acerca de la aprobación de la metodología desarrollada, ésta encuesta será desarrollada por expertos en desarrollo de sistemas de información con conocimientos en migración, para verificar si la metodología es viable.

### **6.1.3. Descripción del Proceso para la Evaluación de la Metodología**

El proceso de evaluación de la metodología consta en la explicación de la misma, a cada encuestado se le otorgó una copia de la metodología desarrollada, anexándole la encuesta.

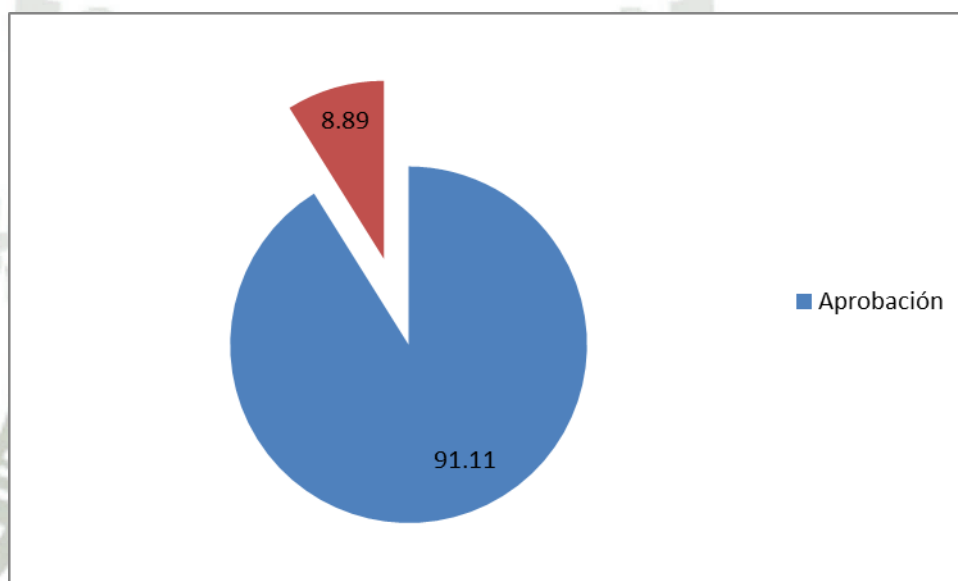
#### 6.1.4. Resultados de la Evaluación Metodológica

##### 1. ¿Aprueba usted la metodología?

Encuestados	Aprobación
18	91,11%

*Tabla 4: Resultado de la pregunta 1*

Fuente: Elaboración propia



*Figura 41: Gráfico de la pregunta 1*

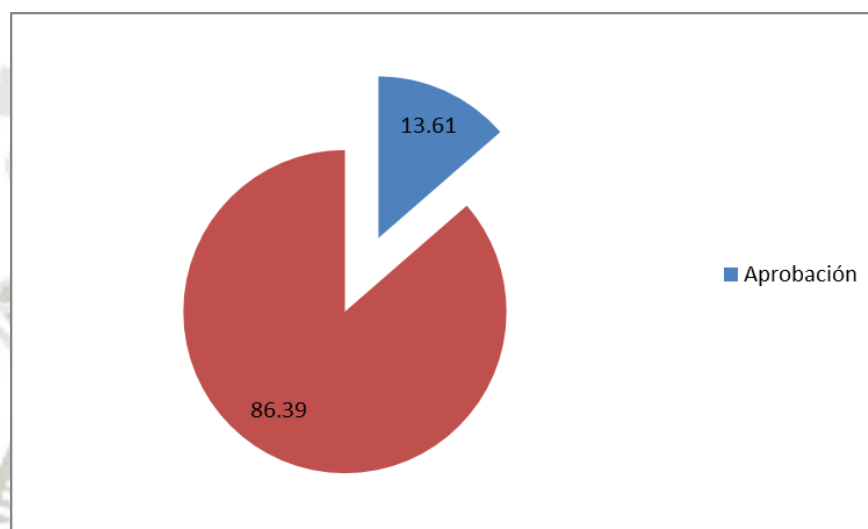
El 91,11% de los encuestados aprueba la metodología. (**Anexo 21**)

2. ¿Considera que ésta metodología es compleja?

Encuestados	Aprobación
18	13.61%

*Tabla 5: Resultado de la pregunta 2*

Fuente: Elaboración propia



*Figura 42: Gráfico de la pregunta 2*

El 13,61% de los encuestados señalo que la metodología es compleja.

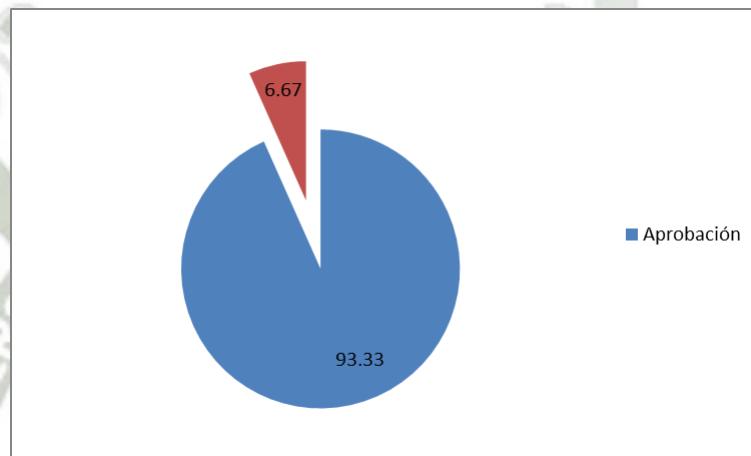
(Anexo 22)

**3. ¿Considera usted que el control de calidad propuesto (CCI, CCE) fue un buen influyente en la realización de ésta metodología?**

Encuestados	Aprobación
18	93.33%

*Tabla 6: Resultado de la pregunta 3*

Fuente: Elaboración propia



*Figura 43: Gráfico de la pregunta 3*

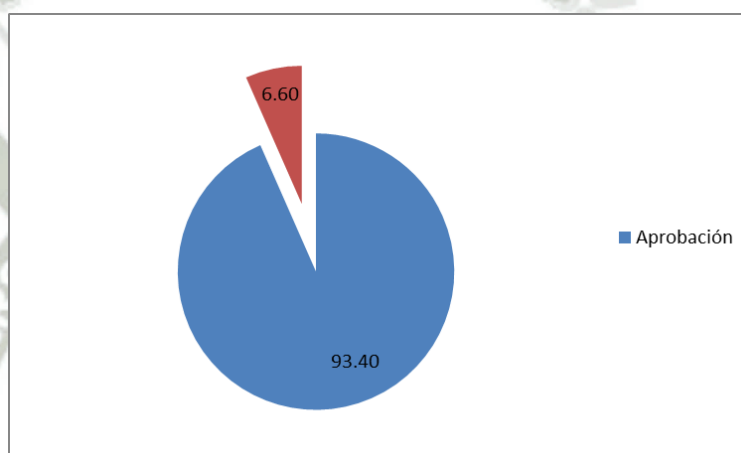
El 93.33% considera que los procesos de calidad fueron buenos influyentes en la metodología. (**Anexo 23**)

4. ¿Considera usted que el uso de las buenas prácticas fueron de gran ayuda para el modelamiento de la metodología?

Encuestados	Aprobación
18	93,40

*Tabla 7: Resultado de la pregunta 4*

Fuente: Elaboración propia



*Figura 44: Gráfico de pregunta 4*

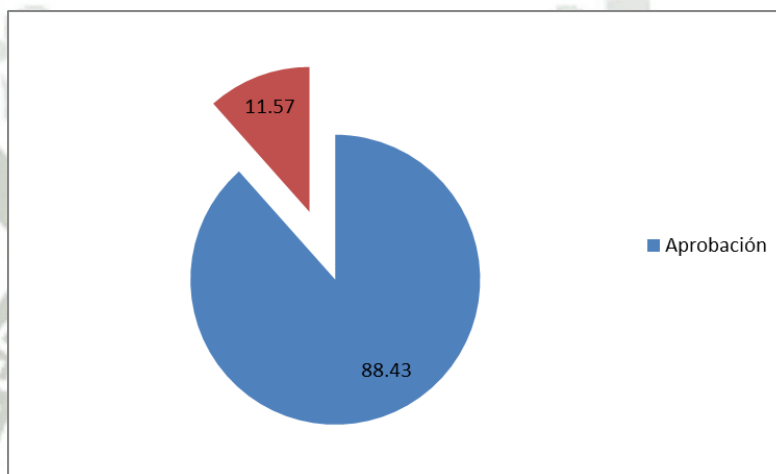
El 93,40% de los encuestados considera que las buenas prácticas fueron de gran ayuda para el modelamiento de la metodología. (**Anexo 24**)

5. ¿Considera usted las buenas prácticas como un valor agregado para la metodología?

Encuestados	Aprobación
18	88.43

*Tabla 8: Resultado de la pregunta 5*

Fuente: Elaboración propia



*Figura 45: Gráfico de la pregunta 5*

El 88.43% de los encuestados considera las buenas prácticas como un valor agregado para la metodología. (**Anexo 25**)

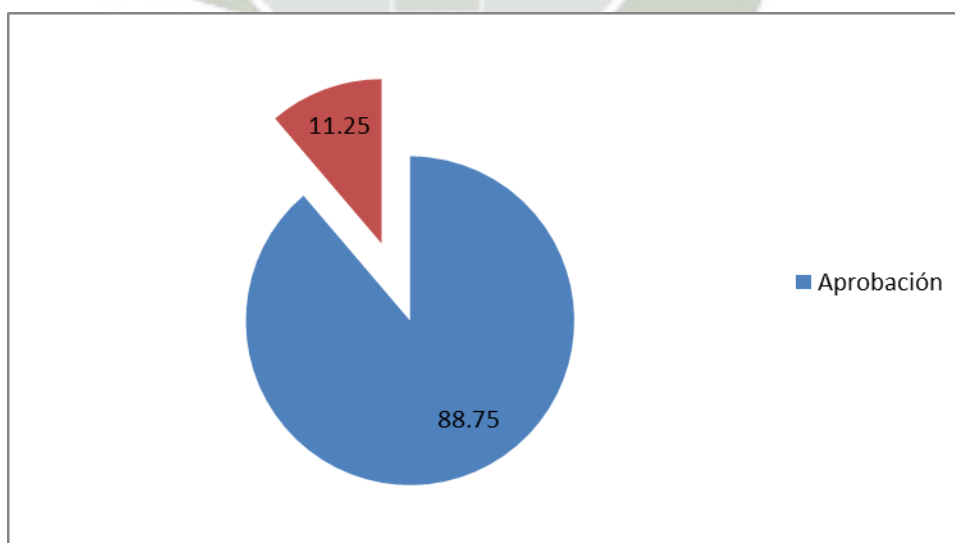
### 6.1.5. Análisis de Objetivos Específicos con la Metodología

Todas las preguntas incluidas en la encuesta que evalúa la metodología, fueron desarrolladas anteriormente, están dirigidas a probar si son válidos los objetivos específicos que se consideran para la metodología y son los siguientes:

Objetivos Específicos	Preguntas
1. Demostrar que es factible realizar una migración con la metodología propuesta.	1,2
2. Utilizar procesos calidad para la migración del aplicativo.	3
3. Demostrar que las mejores prácticas son útiles para migrar un aplicativo	4,5

*Tabla 9: Análisis Objetivos Específicos – Área de TI*

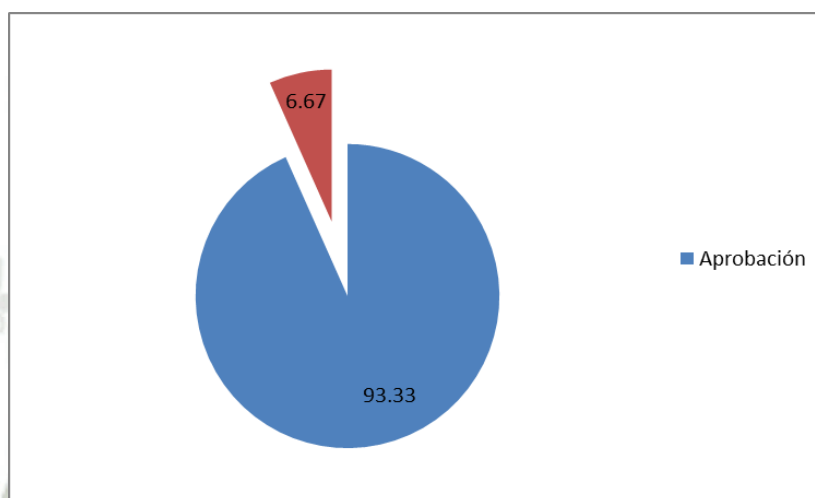
**Objetivo Específico 1:** Demostrar que es factible realizar una migración con la metodología propuesta.



*Figura 46: Gráfico de resultados del objetivo específico 1*

La factibilidad para demostrar que se puede cumplir una migración con la metodología propuesta, indica que el 88,75% de los encuestados considera que la metodología es aceptable.

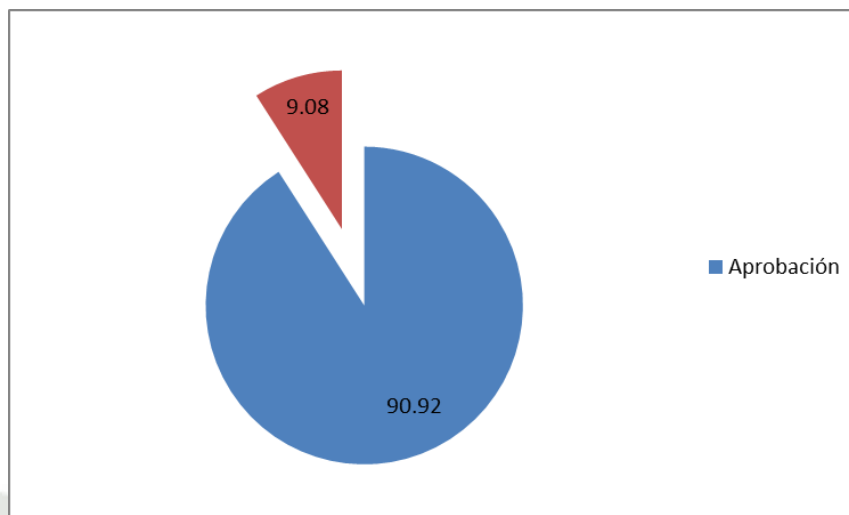
**Objetivo Especifico 2:** Utilizar procesos calidad para la migración del aplicativo.



*Figura 47: Gráfico de resultado del objetivo específico 2*

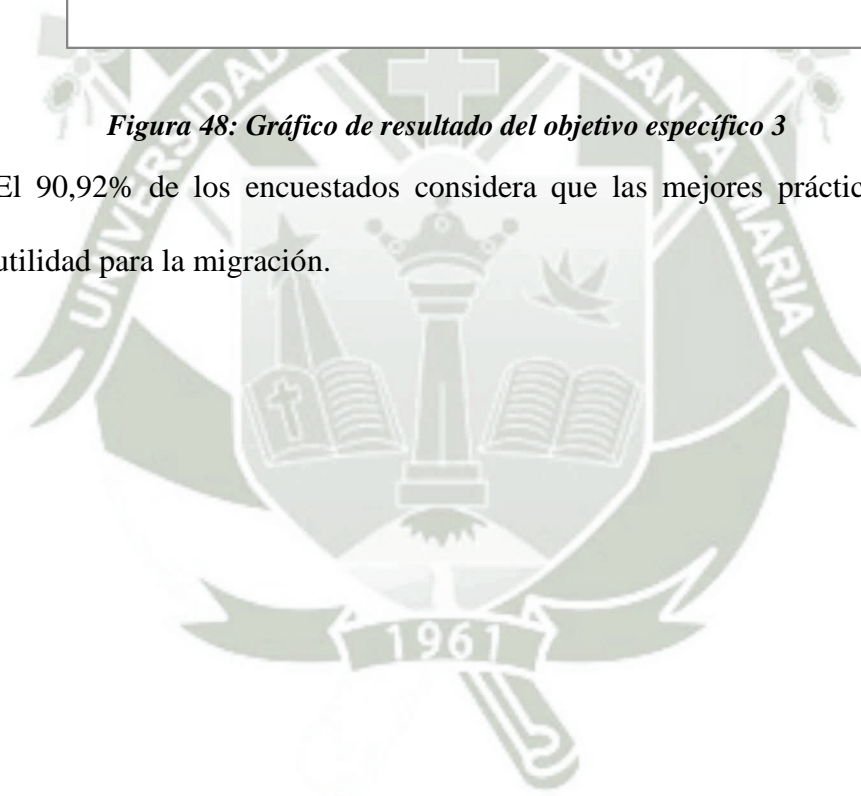
El 93,33% de los encuestados considera que el uso de procesos de calidad es favorable para la migración.

**Objetivo Especifico 3:** Demostrar que las mejores prácticas son útiles para migrar un aplicativo



*Figura 48: Gráfico de resultado del objetivo específico 3*

El 90,92% de los encuestados considera que las mejores prácticas son de utilidad para la migración.



## **6.2. Resultados por parte del Usuario**

### **6.2.1. Consideración para la Evaluación del Aplicativo Migrado**

La metodología propuesta en la presente tesis, demuestra que es posible obtener una adecuada migración de un aplicativo legado utilizando los pasos sugeridos en dicha metodología. A continuación, se intentará comprobar que el aplicativo migrado cumplió con los objetivos trazados en el proyecto.

Se han elegido objetivos específicos que ayudaran a realizar dicha evaluación y así poder demostrar lo beneficiosa que resulta esta metodología para la migración de aplicativos.

La técnica utilizada para hallar la muestra necesaria de la evaluación del aplicativo migrado, es el muestreo por juicio, donde los elementos de la población se seleccionan en base al juicio del experto, en este caso, una muestra de 10 personas que son usuarios finales del sistema, serán suficientes para determinar si el sistema funciona correctamente.

### **6.2.2. Instrumentos para la Evaluación del Aplicativo Migrado**

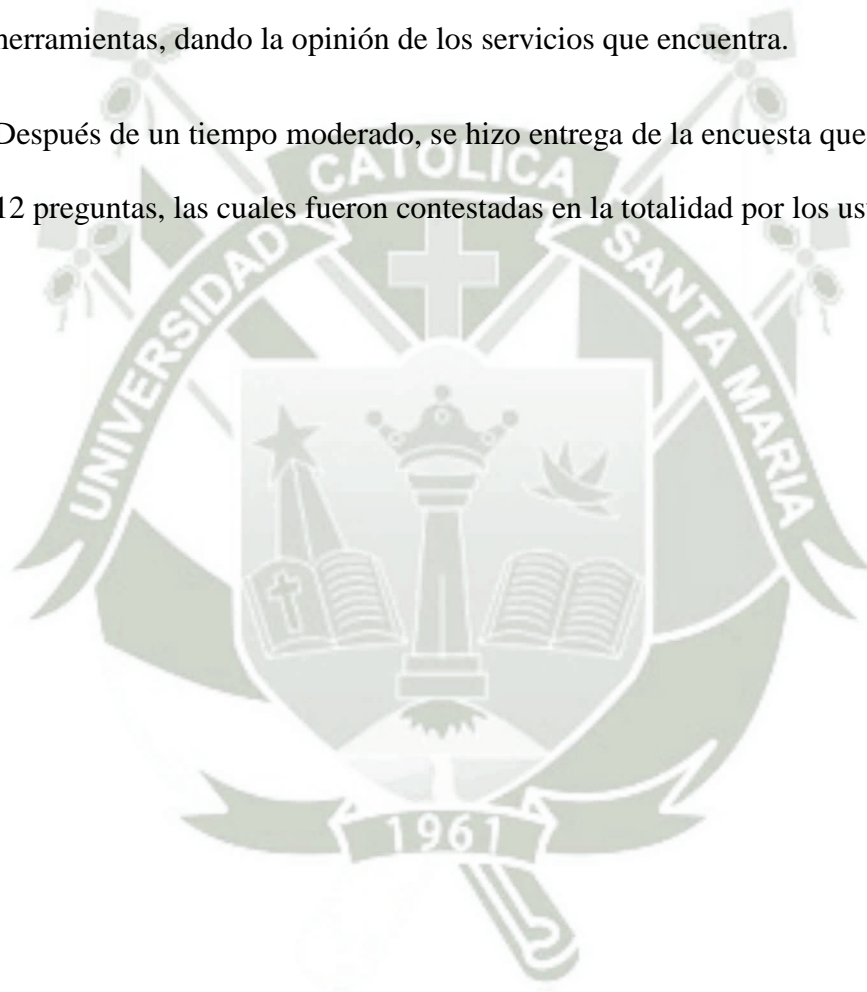
La encuesta será el instrumento que se utilizará para la recolección de información acerca de cómo funciona el aplicativo migrado en la empresa donde se realizó la migración del aplicativo Comisiones, lo que se quiere lograr principalmente es verificar el funcionamiento del aplicativo migrado que fue construido haciendo uso de la metodología propuesta, cuya validación estará determinada por la aprobación de los objetivos específicos propuestos.

### 6.2.3. Descripción de Procesos para la Evaluación del Aplicativo Migrado

El proceso de prueba de la migración consta en la explicación del aplicativo migrado y a que está orientado (satisfacer las necesidades del usuario).

Cada usuario fue encuestado en forma separada, haciendo uso de una PC con el aplicativo ejecutado, otros haciendo uso de *tablets* y *smartphones*. Los usuarios al ingresar al aplicativo, realizaron el uso de las diferentes herramientas, dando la opinión de los servicios que encuentra.

Después de un tiempo moderado, se hizo entrega de la encuesta que consta de 12 preguntas, las cuales fueron contestadas en la totalidad por los usuarios.



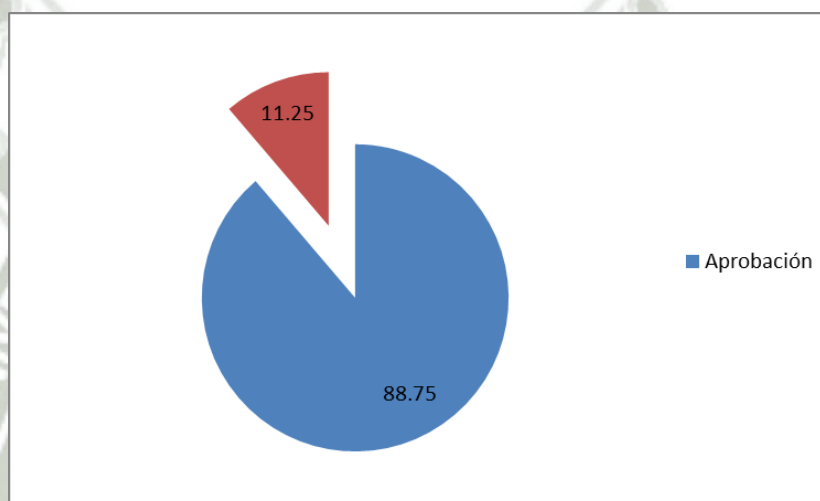
#### 6.2.4. Resultado de la Evaluación del Aplicativo Migrado

##### 1. ¿Cree usted que el aplicativo migrado es escalable a diferencia del legado?

Encuestados	Aprobación
10	88,75%

*Tabla 10: Resultado de la pregunta 1*

Fuente: Elaboración propia



*Figura 49: Gráfico de la pregunta 1*

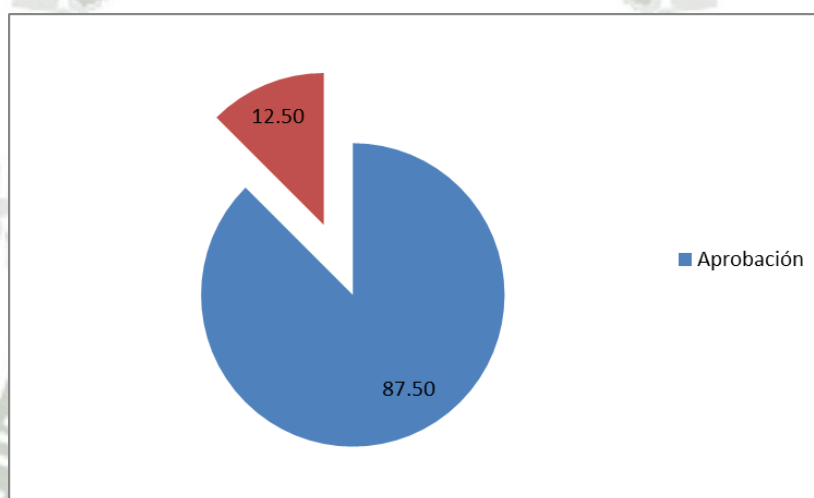
El 88,75% de los encuestados considera que el aplicativo migrado es escalable a diferencia del aplicativo legado. (**Anexo 26**)

2. ¿Cree usted que el aplicativo se adecua a sus necesidades y por lo tanto justifica su ejecución en la empresa?

Encuestados	Aprobación
10	87,50%

*Tabla 11: Resultado de la pregunta 2*

Fuente: Elaboración propia



*Figura 50: Gráfico de la pregunta 2*

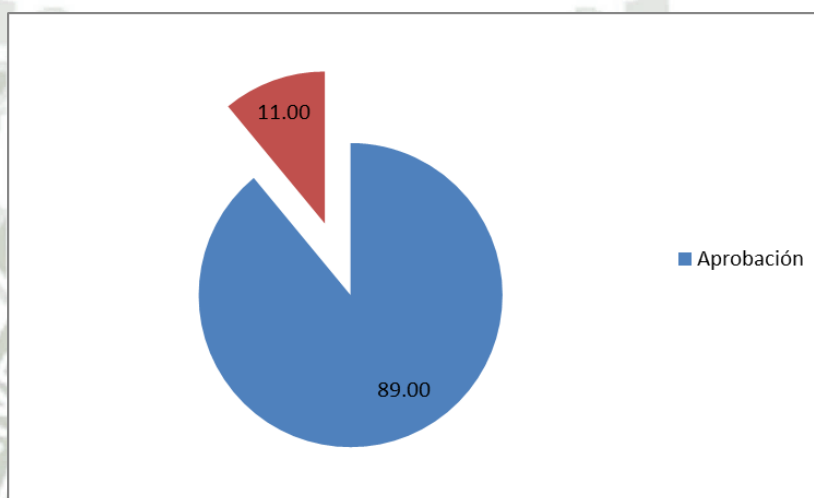
El 87,50% de los encuestados considera que el aplicativo migrado se adecua a sus necesidades. (**Anexo 27**)

3. ¿Considera usted que el aplicativo funcionó adecuadamente durante su ejecución?

Encuestados	Aprobación
10	89,00%

*Tabla 12: Resultados de la pregunta 3*

Fuente: Elaboración propia



*Figura 51: Gráfico de la pregunta 3*

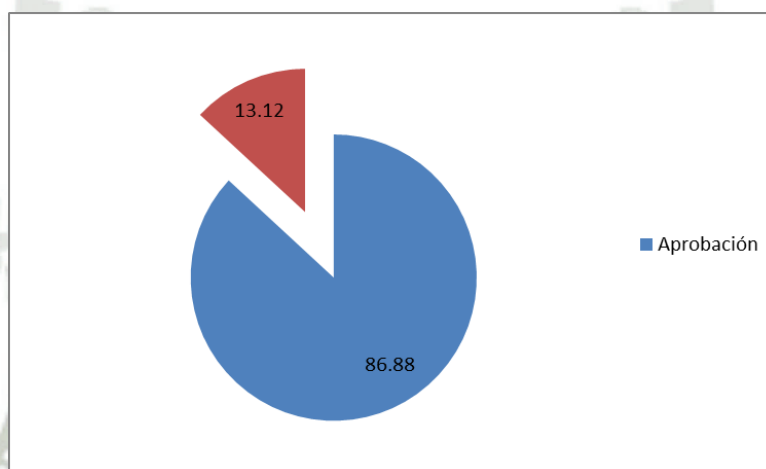
El 89,00% de los encuestados considera que el aplicativo funcionó adecuadamente al ser ejecutado. (**Anexo 28**).

4. ¿Considera usted que los menús y sub-menús se encuentran mejor organizados en el aplicativo migrado?

Encuestados	Aprobación
10	86,88%

*Tabla 13: Resultado de la pregunta 4*

Fuente: Elaboración propia



*Figura 52: Gráfico de la pregunta 4*

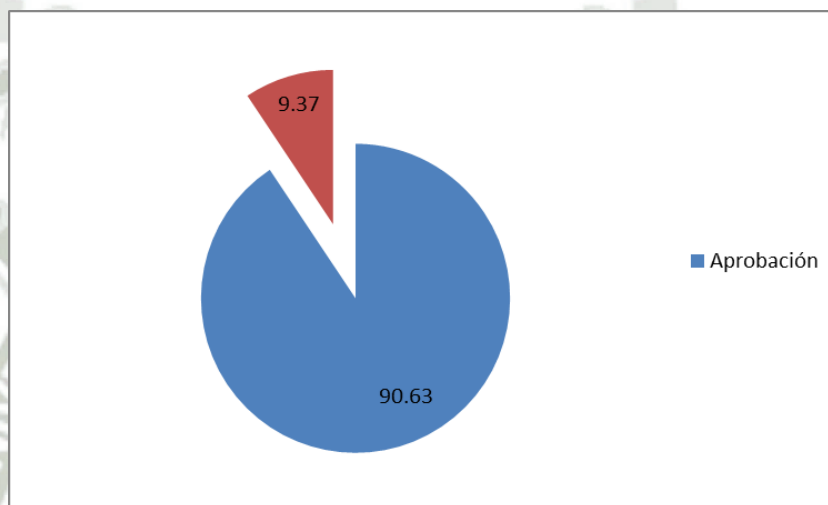
El 86,88% de los encuestados considera que el aplicativo cuenta con una mejor organización de menús y submenús. (Anexo 29)

5. ¿Siente que son confiables los datos que se encuentran en el aplicativo migrado?

Encuestados	Aprobación
10	90,63%

*Tabla 14: Resultado de la pregunta 5*

Fuente: Elaboración propia



*Figura 53: Gráfico de la pregunta 5*

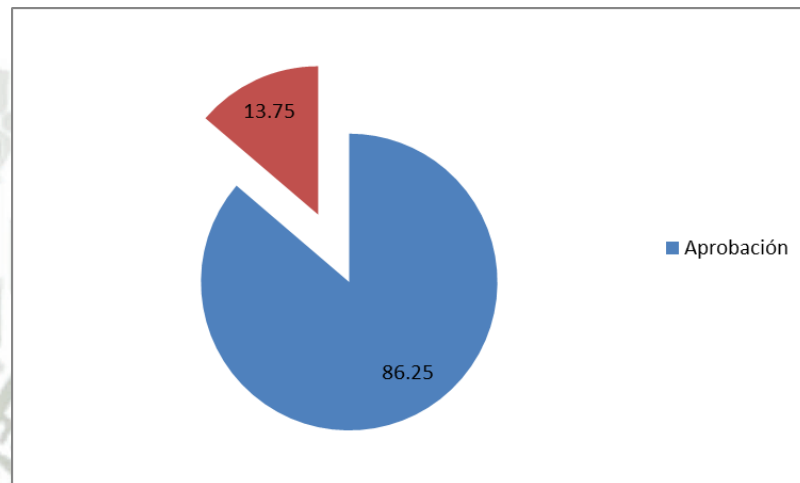
El 90,63% del total de los encuestados considera que los datos distribuidos en este aplicativo son muy confiables. (**Anexo 30**)

6. ¿Cómo considera usted el manejo del aplicativo?

Encuestados	Aprobación
10	86,25%

*Tabla 15: Resultados de la pregunta 6*

Fuente: Elaboración propia



*Figura 54: Gráfico de la pregunta 6*

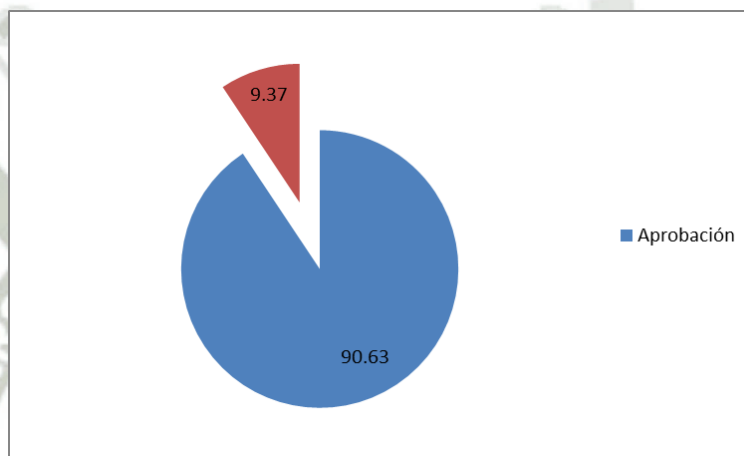
El 86,25% de los encuestados considera que el aplicativo no es complejo de utilizar. (**Anexo 31**)

7. ¿Se encuentra satisfecho con el trato personalizado que le brinda el aplicativo?

Encuestados	Porcentaje
10	90,63%

*Tabla 16: Resultado de la pregunta 7*

Fuente: Elaboración propia



*Figura 55: Gráfico de la pregunta 7*

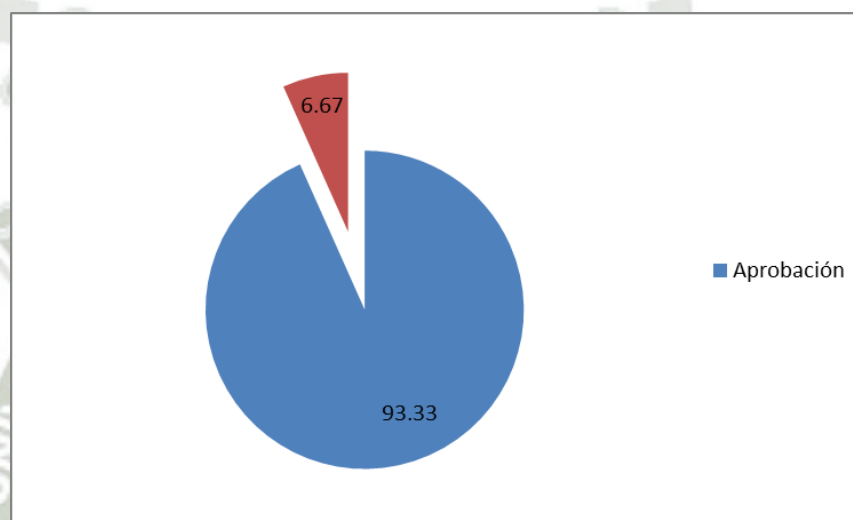
El 90,63% de los encuestados quedó completamente satisfecho con el trato personalizado que le brinda el aplicativo migrado. (**Anexo 32**).

8. ¿Considera usted que la portabilidad del aplicativo sea una ventaja?

Encuestados	Aprobación
10	93,33%

*Tabla 17: Resultado de la pregunta 8*

Fuente: Elaboración propia



*Figura 56: Gráfico de la pregunta 8*

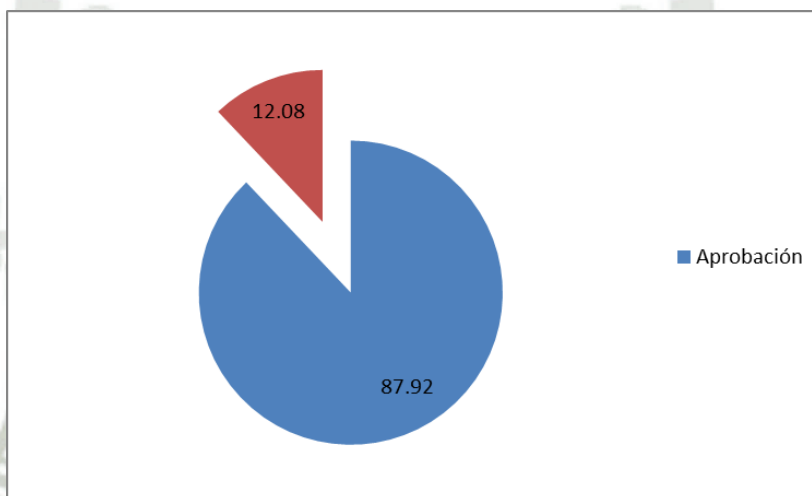
El 93,33% de los encuestados considera que la portabilidad del aplicativo es una ventaja (**Anexo 33**).

9. ¿Considera usted que la interface del aplicativo migrado es mejor que la interface del aplicativo legado?

Encuestados	Aprobación
10	87,92%

*Tabla 18: Resultado de la pregunta 9*

Fuente: Elaboración propia



*Figura 57: Gráfico de la pregunta 9*

El 87,92% del total de los encuestados considera que la interface del aplicativo migrado es mejor que la del legado. (Anexo 34)

### 6.2.5. Análisis de los Objetivos Específicos del Aplicativo Migrado

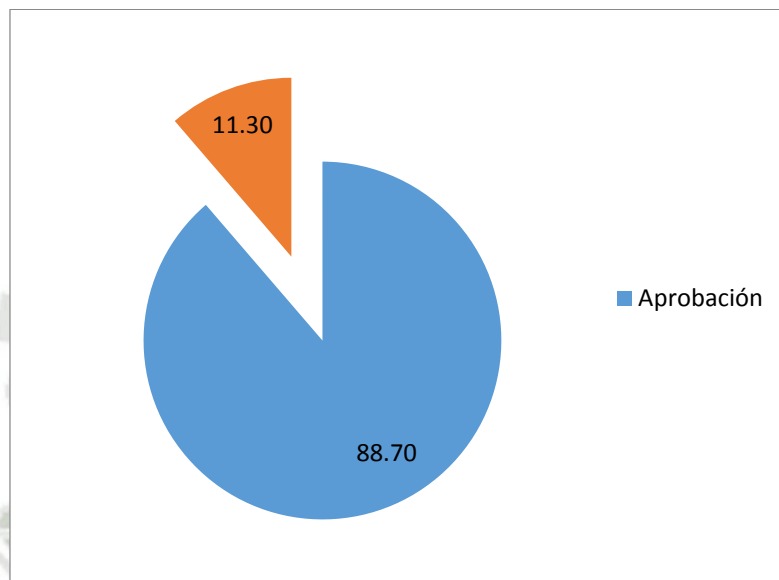
Todas las preguntas incluidas en la encuesta que se desarrollaron anteriormente, estaban dirigidas a probar la validez de los indicadores que se consideraron para este proyecto, y son:

Objetivos Específicos	Preguntas
1. Demostrar que un aplicativo migrado, puede ser mejor usado que un aplicativo legado.	1,2,7,9
2. Demostrar que el aplicativo migrado es portable.	8
3. Demostrar que el aplicativo migrado es igual de confiable que el aplicativo legado, respecto a la información que muestra.	5,3
4. Demostrar que el aplicativo migrado es más flexible en comparación al aplicativo legado.	4,6

*Tabla 19: Análisis de Objetivos Específicos - Usuario*

Los indicadores considerados en este trabajo de tesis, son evaluados mediante preguntas según la tabla anterior, así podemos analizar los siguientes resultados:

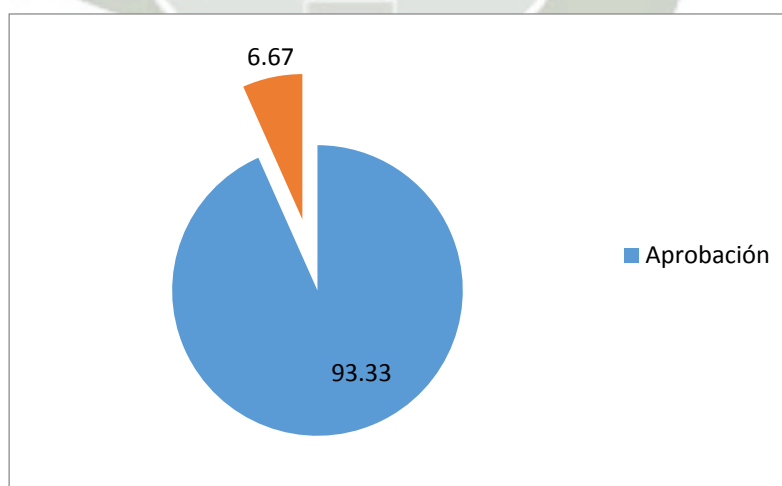
**Objetivo específico 1:** Demostrar que un aplicativo migrado, puede ser mejor usado que un aplicativo legado.



*Figura 58: Gráfico de resultado del objetivo específico 1*

El 88,70% de los encuestados aprueba que el aplicativo migrado puede ser mejor usado que el aplicativo legado.

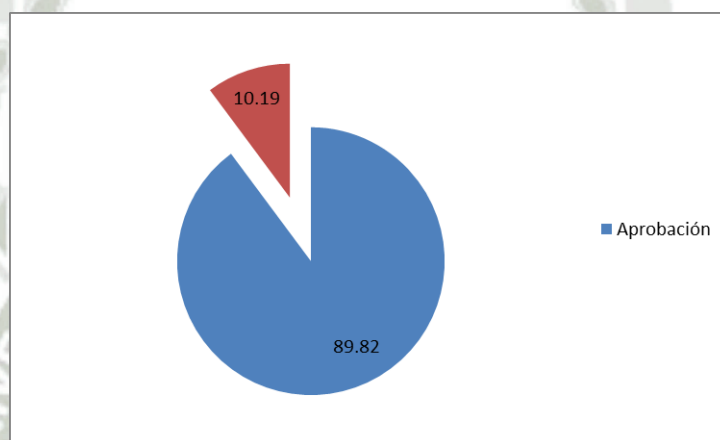
**Objetivo Especifico 2:** Demostrar que el aplicativo migrado es portable.



*Figura 59: Gráfico de resultados del objetivo específico 2*

La portabilidad del aplicativo migrado tiene un nivel muy aceptable, demostrando que es una gran ventaja, el 93,33% de los encuestados lo considera así.

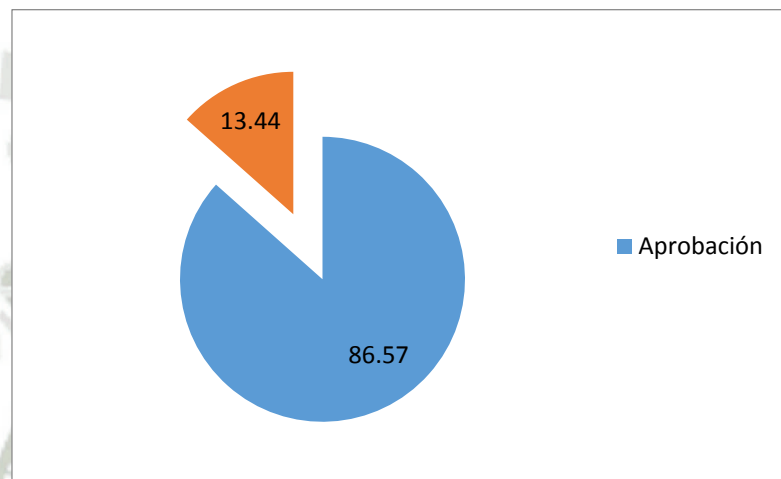
**Objetivo Específico 3:** Demostrar que el aplicativo migrado es igual de confiable que el aplicativo legado, respecto a la información que muestra.



*Figura 60: Gráfico de resultados del objetivo específico 3*

El nivel de confiabilidad del aplicativo migrado es aceptable, ya que el 89,82% de encuestados lo apreció así, permitiendo así que los encuestados confíen en el aplicativo.

**Objetivo Especifico 4:** Demostrar que el aplicativo migrado es más flexible en comparación al aplicativo legado.



*Figura 61: Gráfico de resultados del objetivo específico 4*

La flexibilidad del aplicativo migrado fue un indicador muy probado, demostrado por la satisfacción del usuario en un 86,57%.

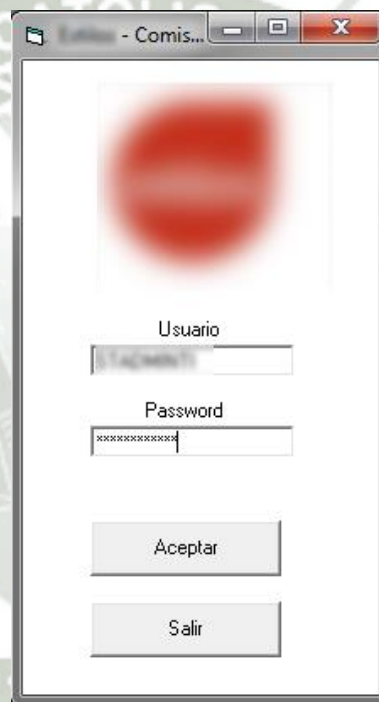
### 6.3. Resultados del Aplicativo Migrado VS Aplicativo Legado

Revisaremos algunas de las interfaces del aplicativo legado, comparando con el aplicativo migrado, rescatando las mejoras que se tiene en la migración

#### 6.3.1. Ingreso al Aplicativo Comisiones

##### a. Legado

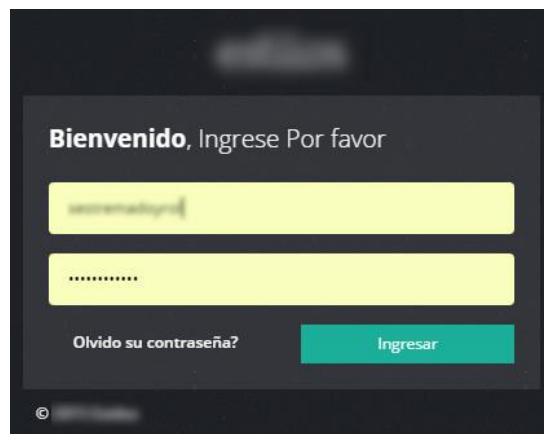
Para acceder al aplicativo, se ingresa el usuario y contraseña del aplicativo, guardados en una base de datos.



*Figura 62: Ingreso al aplicativo legado*

##### b. Migrado

Para acceder al aplicativo migrado, se ingresa el usuario y contraseña, que son validados por *Active Directory*, evitando que el usuario use más contraseñas indebidas, siendo esta una mejora para la migración.

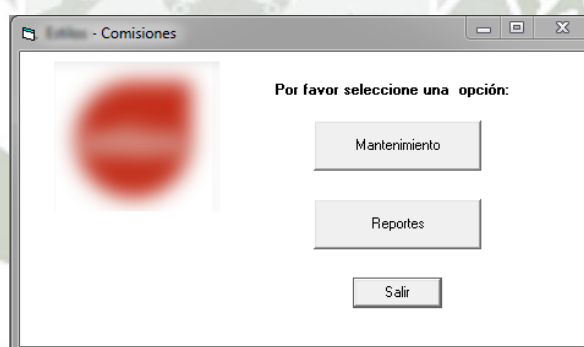


*Figura 63: Ingreso al aplicativo migrado*

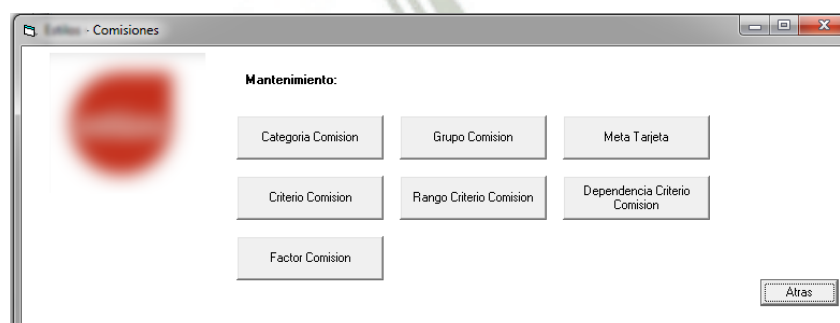
### 6.3.2. Menú Principal del Aplicativo Comisiones

#### a. Legado

Al acceder al aplicativo, tenemos un menú básico, y para ingresar al sub-menú nos muestra otra ventana.



*Figura 64: Menú principal - Legado*



*Figura 65: Sub-Menú Mantenimiento - Legado*

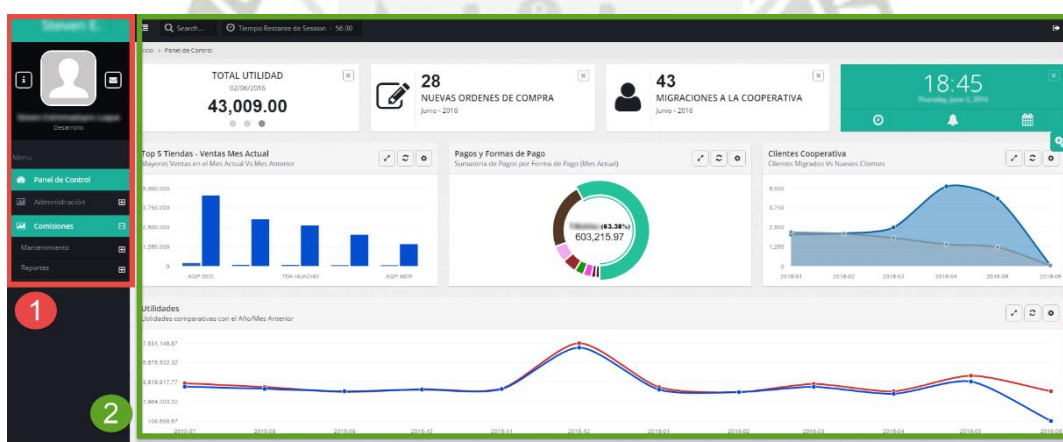


*Figura 66: Sub-Menú Reportes - Legado*

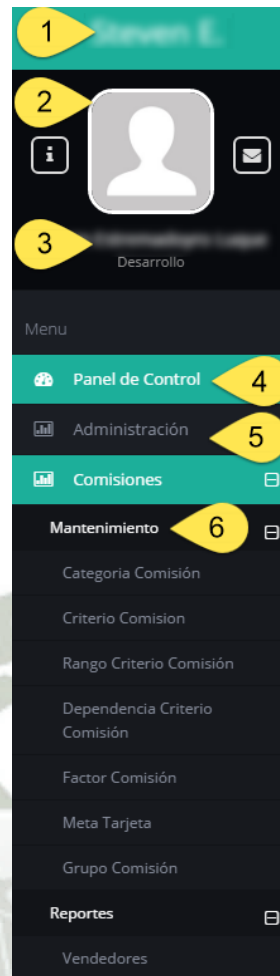
**b. Migrado**

Para el caso el aplicativo migrado, el menú principal se divide en dos:

1. Menú de opciones del aplicativo.
2. Widgets con el resumen estadístico entre tiendas, ya sea de ventas, utilidades, etc.



*Figura 67: Menú Principal - Migración*

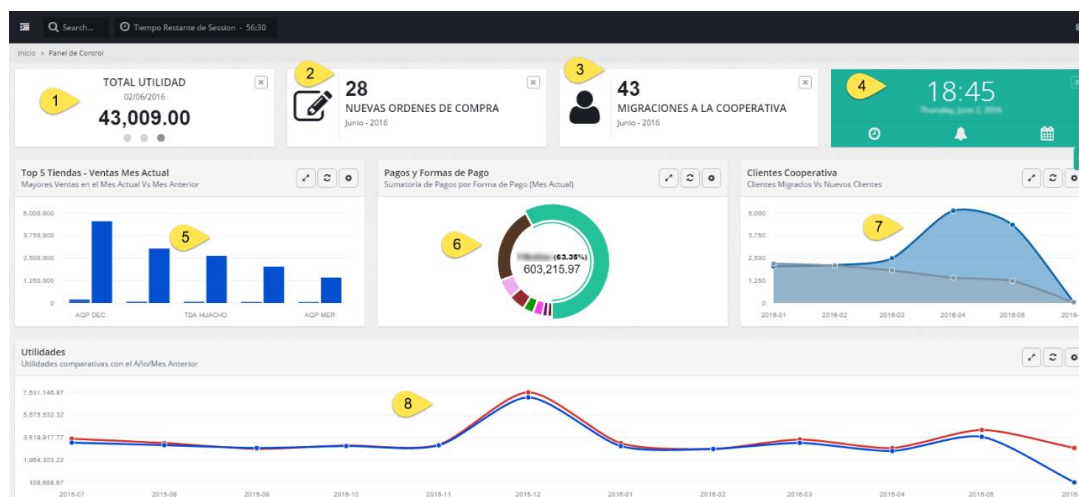


**Figura 68: Menú de opciones - Migración**

Como podemos ver en la figura anterior, el menú de opciones es sencillo de usar y dinámico, teniendo las siguientes mejoras:

1. Nombre del usuario que está usando el aplicativo
2. Se puede colocar la foto del usuario, para que sea personalizado
3. Rol que ocupa el usuario
4. Cuenta con un panel de control, que es básicamente el menú de widgets.

5. Cuenta con la opción de administración, que es donde están los usuarios que pueden acceder al aplicativo, según sean sus permisos de lectura y/o escritura, de acuerdo al rol que tengan.
6. Menú de opciones es desplegable.



**Figura 69: Menú de Widgets – Migración**

En la figura anterior, podemos ver que la interface es completamente dinámica, y muestra las siguientes mejoras:

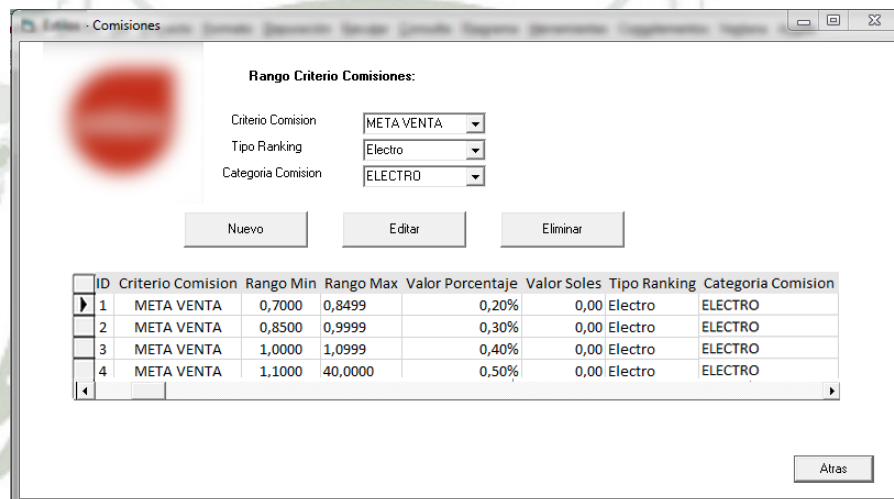
1. Total de utilidades, ventas y compras.
2. Nuevas órdenes de compra.
3. Clientes de la cooperativa.
4. Reloj y calendario
5. Top 5 de las tiendas que tienen más ventas.
6. Muestra el total de pagos, de acuerdo a las distintas formas de pago.
7. Cantidad de nuevos clientes a la cooperativa por mes.
8. Utilidades obtenidas por mes.

Estas mejoras se realizaron mediante procedimientos almacenados a la base de datos.

### 6.3.3. Mantenimiento de Rango Criterio Comisión

#### a. Legado

Para realizar el mantenimiento del rango de criterio, nos muestra la lista de los rangos actuales, y para poder ingresar o editar un registro, nos carga otra ventana donde nos muestra los datos para poder guardarlos.



**Rango Criterio Comisiones:**

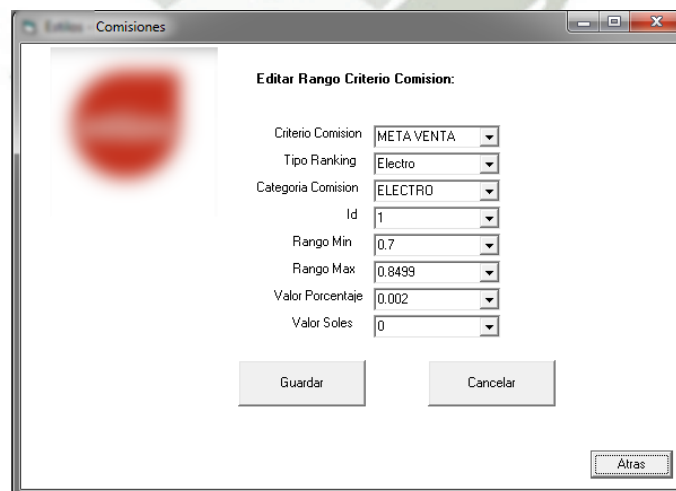
Criterio Comision: META VENTA  
 Tipo Ranking: Electro  
 Categoría Comision: ELECTRO

Nuevo    Editar    Eliminar

ID	Criterio Comision	Rango Min	Rango Max	Valor Porcentaje	Valor Soles	Tipo Ranking	Categoría Comision
1	META VENTA	0,7000	0,8499	0,20%	0,00	Electro	ELECTRO
2	META VENTA	0,8500	0,9999	0,30%	0,00	Electro	ELECTRO
3	META VENTA	1,0000	1,0999	0,40%	0,00	Electro	ELECTRO
4	META VENTA	1,1000	40,0000	0,50%	0,00	Electro	ELECTRO

Altras

*Figura 70: Mantenimiento de rango - Legado*



**Editar Rango Criterio Comision:**

Criterio Comision: META VENTA  
 Tipo Ranking: Electro  
 Categoría Comision: ELECTRO  
 Id: 1  
 Rango Min: 0,7  
 Rango Max: 0,8499  
 Valor Porcentaje: 0,002  
 Valor Soles: 0

Guardar    Cancelar

Altras

*Figura 71: Editar Rango - Legado*

## b. Migrado

The screenshot shows a web application interface for 'Rango Criterio Comisiones'. The interface includes a search bar (1), a session timer (2), a breadcrumb trail (3), a title bar (6), a 'Nuevo R' button (5), an 'Exportar' dropdown menu (6) with options for CSV, TXT, XLS, and PDF, a search filter section (4) with dropdowns for 'Criterio Comision' (META VENTA), 'Tipo Ranking' (Electro), and 'Categoria Comision' (ELECTRO), a table of records (7) with columns for Id, Criterio Comision, Rango Min, Rango Max, Valor Porcentaje, Valor Soles, Tipo Ranking, and Categoria Comision, and a search input field (7). The table shows 4 records. The interface also includes a 'Mostrar 10 Registros' label (4), a 'Buscar:' field (7), and a pagination bar (8) with 'Anterior', '1', and 'Siguiete' buttons.

Administración de Rango Criterio Comisiones

Filtros de Búsqueda

Criterio Comision: META VENTA  
Tipo Ranking: Electro  
Categoria Comision: ELECTRO

Mostrar 10 Registros

Id	Criterio Comision	Rango Min	Rango Max	Valor Porcentaje	Valor Soles	Tipo Ranking	Categoria Comision
1	META VENTA	0.7000	0.8499	0.20 %	0.00	Electro	ELECTRO
2	META VENTA	0.8500	0.9999	0.30 %	0.00	Electro	ELECTRO
3	META VENTA	1.0000	1.0999	0.40 %	0.00	Electro	ELECTRO
4	META VENTA	1.1000	40.0000	0.50 %	0.00	Electro	ELECTRO

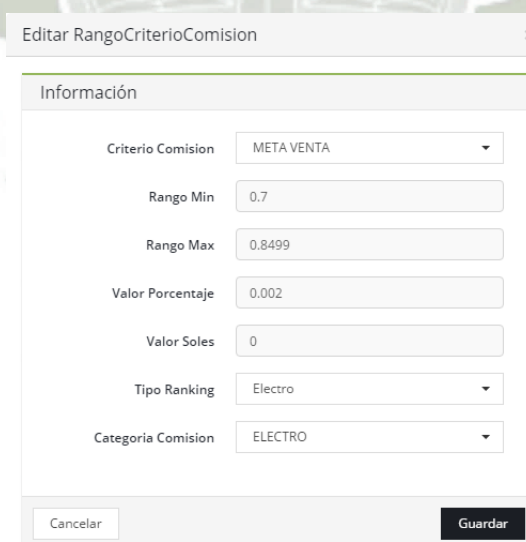
Mostrando 1 a 4 de 4 Registros

Figura 72: Mantenimiento de Rango - Migración

En la figura anterior podemos visualizar que la información se muestra de forma más ordenada, contando con las siguientes mejoras:

1. Búsqueda en todo el aplicativo.
2. Tiempo de sesión.
3. Ruta de acceso.
4. Cantidad de registros que desea mostrar por página, incluyendo la opción todos.
5. Se puede exportar los datos mostrados en distintos formatos.
6. Permite maximizar la información y actualizarlos.
7. Permite realizar la búsqueda dentro de los datos mostrados.
8. La forma que se muestra las opciones de editar y eliminar son dinámicas.

Para editar los registros se muestra como un pop-up, como se ve en la siguiente figura:



Información	
Criterio Comision	META VENTA
Rango Min	0.7
Rango Max	0.8499
Valor Porcentaje	0.002
Valor Soles	0
Tipo Ranking	Electro
Categoria Comision	ELECTRO

Cancelar Guardar

**Figura 73: Editar Rango - Migración**

### 6.3.4. Reporte de Vendedores del Aplicativo Comisión

#### a. Legado

En la siguiente figura, muestra el resultado del reporte de vendedores con los filtros indicados, mostrando 21 registros.

CODIGO	VENDEDOR	INGRESO	PERFORMANCE	SUPERVISOR	GRUPO	CUOTA
10549	LAURA FLORA MIRANDA QUISPE	12/06/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
15256	ANDREA ROJAS YANQUI	02/04/2012	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
18156	JESSEBELLA DOLLY RAMOS PERALTILLA	05/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
19891	MAURICIO RAFAEL YOSHIOKA VALDEZ	16/04/2015	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
20472	MICHAEL ALEXANDER TURPO GUILLEN	14/09/2015	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
20993	GABRIELA CALLATA PUMA	16/01/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	-136173.93
21132	BRANDON DENIS PALMA RODRIGUEZ	05/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21133	FANY VASQUEZ MACHACA	05/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21147	MAURICIO GERARDO SUEROS TORRES	06/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21220	JULINIO FELIX CACERES ESPINO	15/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21221	MARIA MERCEDES ROLDAN YANQUI	15/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21222	ASTRID JESABELI CAHUANA VALDIVIA	16/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
21296	RICHARD ALEXANDER HUARECALLO NINANTAY	25/02/2016	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
2639	MARIETA ULLOA CURAHUJA	01/05/2009	0	SUPERVISOR	COLCHON - JUGUETERIA INFANTIL,PERFUMERIA,MAQ	90782.59
18785	NORMA BEATRIZ CALCINA CHOQUEHUANCA	19/06/2014	0	SUPERVISOR	DECO - MUEBLES	1740000.03
20947	YONI AQUINO HUISA	07/01/2016	0	SUPERVISOR	DECO - MUEBLES	-1044000
20965	FIORELLA ALEJANDRA CORNEJO RAMOS	09/01/2016	0	SUPERVISOR	DECO - MUEBLES	1740000.03
30025	FREDY LUQUE HUANCA	11/01/2016	0	SUPERVISOR	DECO - MUEBLES	1740000.03
21175	JHON ALEX HUARZA TAIPE	11/02/2016	0	SUPERVISOR	DECO - MUEBLES	-1044000
21265	RAUL JOSE MALDONADO CACERES	18/02/2016	0	SUPERVISOR	DECO - MUEBLES	-1044000
21292	YULIANA SARMIENTO MAMANI	25/02/2016	0	SUPERVISOR	DECO - MUEBLES	-1044000

Figura 74: Reporte de vendedores - Legado

#### b. Migrado

El reporte muestra los mismos datos que el aplicativo legado, tomando como mejora las opciones ya descritas anteriormente, como la exportación, búsqueda de datos y la visualización.

Reporte Comisiones - Vendedores

Guardar Reporte | Exportar

Filtros de Búsqueda: Mes: Abril-2016, Tienda: TDA. DECO-HOGAR, Tipo Ranking: Tienda, Estado Reporte: ABIERTO

Guardado	Tienda	Codigo	Vendedor	Fec Ingreso	Performance Ant	Nombre Supervisor	Grupo	Costa	Logro	Avance 1	Comision Venta	Costa Credito
x	TDA. DECO-HOGAR	10549	LAURA FLORA MIRANDA QUISPE	12/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	26,345.49	0.29	0.00	0.85
x	TDA. DECO-HOGAR	15256	ANDREA ROJAS VANQUI	02/04/2012	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	85,323.19	0.94	361.54	0.85
x	TDA. DECO-HOGAR	18156	JESABELLA DOLLY RAMOS PERALTILLA	05/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	19891	MAURICIO RAFAEL YOSHOMA VALDEZ	16/04/2015	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	20472	MICHAEL ALEXANDER TURPO GUILLEN	14/03/2015	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	20,587.60	0.23	0.00	0.85
x	TDA. DECO-HOGAR	20993	GABRIELA CALLATA PUMA	16/01/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	-136,173.93	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21132	BRANDON DENIS PALMA RODRIGUEZ	05/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21133	FANY VASQUEZ MACHACA	05/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21147	MAURICIO GERARDO SUEROS TORRES	06/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21220	JULIANO TELIX CACERES ESPINO	15/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21221	MARIA MERCEDES ROLDAN VANQUI	15/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	21222	ASTRID JESSELI CAHUANA VALDIVIA	16/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	13,948.28	0.15	0.00	0.85
x	TDA. DECO-HOGAR	21296	RICHARD ALEXANDER HUARECALLO NINANTAY	25/02/2016	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	2639	MARILETA ULLCA CURAHUA	01/05/2009	0.00 %	SUPERVISOR	COLCHON - JUGUETERIA, INFANTIL, PERFUMERIA, MAQ	90,782.59	44,432.37	0.49	0.00	0.85
x	TDA. DECO-HOGAR	18785	NORMA BEATRIZ CALONA CHOQUEHUANCA	18/05/2014	0.00 %	SUPERVISOR	DECO - MUEBLES	1,740,000.00	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	20947	YONI AQUINO HUISA	07/01/2016	0.00 %	SUPERVISOR	DECO - MUEBLES	-1,044,000.00	90,992.61	-0.09	0.00	0.85
x	TDA. DECO-HOGAR	20965	PIRELLA ALEJANDRA CORNEJO RAMOS	08/01/2016	0.00 %	SUPERVISOR	DECO - MUEBLES	1,740,000.00	0.00	0.00	0.00	0.85
x	TDA. DECO-HOGAR	20988	FREDDY LUIS DE LA TORRE	11/01/2016	0.00 %	SUPERVISOR	DECO - MUEBLES	1,740,000.00	0.00	0.00	0.00	0.85

Mostrando 1 a 21 de 21 Registros

Figura 75: Reporte de vendedores – Migración

### 6.3.5. Análisis de Resultados con ISO 9126

Para concluir los resultados del aplicativo migrado versus el aplicativo legado, se realizó las métricas de calidad de software de la ISO 9126, donde ha establecido un estándar internacional para la evaluación de la calidad de productos de software. El estándar ISO-9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad; cada una de las cuales se detalla a través de un conjunto de sub-características que permiten profundizar en la evaluación de la calidad de productos de software.

[32]

**a. Funcionalidad**

<b>Sub-Característica</b>	<b>Aplicabilidad:</b> Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.
<b>Nombre:</b>	Completitud de implementación funcional
<b>Propósito:</b>	Qué tan completa está la implementación
<b>Método de aplicación:</b>	Contar las funciones faltantes detectadas en la evaluación y comparar con el número de funciones descritas en la especificación de requisitos.
<b>Medición, fórmula:</b>	$X = 1 - A/B$ A = número de funciones faltantes B = número de funciones descritas en la especificación de requisitos.
<b>Interpretación:</b>	$0 \leq X \leq 1$ Entre más cercano a 1, más completa.
<b>Fuente de medición:</b>	Especificación de requisitos Diseño Código fuente Informe de revisión
<b>Audiencia:</b>	Requeridores Desarrolladores
<b>Resultado Legado</b>	$X = 1 - 7/31$ $X = 0.77$
<b>Resultado Migración</b>	$X = 1 - 0/38$ $X = 1$

*Tabla 20: Resultado de la Funcionalidad - Aplicabilidad*

**b. Fiabilidad**

<b>Sub-Característica</b>	<b>Madurez:</b> Capacidad del producto software para evitar fallar como resultado de fallos en el software.
Nombre:	Suficiencia de las pruebas
Propósito:	Cuantos de los casos de prueba necesarios están cubiertos por el plan de pruebas.
Método de aplicación:	Contar con las pruebas planeadas y comparar con el número de pruebas requeridas para obtener una cobertura
Medición, fórmula:	$X = A/B$ A = número de casos de prueba en el plan. B = número de casos de prueba requeridos.
Interpretación:	$0 \leq X$ Entre X sea mayor, mejor la suficiencia.
Fuente de medición:	A proviene del plan de pruebas. B proviene de la especificación de requisitos.
Audiencia:	Validadores de Contenidos Desarrolladores
<b>Resultado Legado</b>	<b>X =</b> Para el caso del aplicativo legado, no se cuenta con la documentación donde nos muestre los casos de prueba en el plan y requeridos.
<b>Resultado Migración</b>	<b>X = 24/24</b> <b>X = 1</b>

*Tabla 21: Resultado de Fiabilidad - Madurez*

**c. Usabilidad**

<b>Sub-Característica</b>	<b>Entendibilidad:</b> Capacidad del producto software que permite al usuario entender si el software es adecuado y como puede ser usado para unas tareas o condiciones de uso particulares.
Nombre:	Funciones evidentes
Propósito:	Qué proporción de las funciones del aplicativo son evidentes al usuario.
Método de aplicación:	Contar las funciones evidentes al usuario y comparar con el número total de funciones.
Medición, fórmula:	$X = A/B$ A = número de funciones evidentes al usuario. B = total de funciones.
Interpretación:	$0 \leq X \leq 1$ Entre X sea más cercano a 1, mejor.
Fuente de medición:	Especificación de requisitos. Diseño Informe de revisión
Audiencia:	Requeridores Desarrolladores
<b>Resultado Legado</b>	$X = 22/31$ $X = 0.70$
<b>Resultado Migración</b>	$X = 34/38$ $X = 0.89$

**Tabla 22: Resultados de Usabilidad - Entendibilidad**

**d. Eficiencia**

<b>Sub-Característica</b>	<b>Comportamiento en el tiempo:</b> La capacidad del producto de software para proveer tiempos adecuados de respuesta y procesamiento, y ratios de rendimiento cuando realiza su función bajo las condiciones establecidas.
<b>Nombre:</b>	Tiempo de respuesta
<b>Propósito:</b>	Cuál es el tiempo estimado para completar una tarea.
<b>Método de aplicación:</b>	<p>Evaluar la eficiencia de las llamadas al SO y a la aplicación.</p> <p>Estimar el tiempo de respuesta basado en ello. Puede medirse:</p> <ul style="list-style-type: none"> <li>• Todo o partes de las especificaciones de diseño.</li> <li>• Probar la ruta completa de una transacción.</li> <li>• Probar módulos o partes completas del producto.</li> <li>• Producto completo durante la fase de pruebas.</li> </ul>
<b>Medición, fórmula:</b>	$X = \text{tiempo (calculado o simulado)}$
<b>Interpretación:</b>	Entre más corto, mejor.
<b>Fuente de medición:</b>	Sistema operativo conocido. Tiempo estimado en llamadas a sistema
<b>Audiencia:</b>	Requeridores Desarrolladores
<b>Resultado Legado</b>	<p>Para este caso se medirá de acuerdo a la ruta completa de una transacción</p> <p><math>X =</math> Consulta de reporte “Comisión Vendedores por el mes de junio 2016 de la tienda Deco-Hogar”</p>

	$X = 35.340 \text{ seg}$
<b>Resultado Migración</b>	Para este caso se medirá de acuerdo a la ruta completa de una transacción $X =$ Consulta de reporte “Comisión Vendedores por el mes de junio 2016 de la tienda Deco-Hogar” $X = 31.750 \text{ seg}$

*Tabla 23: Resultado de eficiencia – Comportamiento en el Tiempo*

**e. Facilidad de Mantenimiento**

<b>Sub-Característica</b>	<b>Analizabilidad:</b> Es la capacidad del producto software para serle diagnosticadas deficiencias a causa de los fallos en el software, o para identificar las partes que han de ser modificadas.
Nombre:	Densidad de líneas de código por función.
Propósito:	Determinar la densidad del código de las funciones.
Método de aplicación:	Contar el número de líneas de código de un componente software o todo el software entre el número de funciones de software.
Medición, fórmula:	$X = B * 100 / A$ A = Cantidad de líneas de código (no se incluyen los comentarios ni registros de cambios). B = Cantidad total de funciones del software.
Interpretación:	$0 \leq X < 100$ Entre más cercano a cero, mejor la densidad de líneas de código.
Fuente de medición:	Especificación de requisitos funcionales.

	Diseño
	Código fuente.
	Informe de revisión
Audiencia:	Experto de Contenidos
	Analista
	Experto Migración
<b>Resultado Legado</b>	Para este caso se utilizará el número de líneas de una función (Reporte Comisión Vendedores). $X = 1 * 100/28$ $X = 3.57$
<b>Resultado Migración</b>	Para este caso se utilizará el número de líneas de una función (Reporte Comisión Vendedores). $X = 1 * 100/42$ $X = 2.38$

**Tabla 24: Resultado Facilidad de Mantenimiento – Analizabilidad**

<b>Sub-Característica</b>	<b>Cambiabilidad:</b> Capacidad del producto software que permite que una determinada modificación sea implementada.
Nombre:	Registrabilidad de cambios.
Propósito:	¿Se registran adecuadamente los cambios a la especificación y a los módulos con comentarios en el código?
Método de aplicación:	Registrar la proporción de información sobre cambios a los módulos.
Medición, fórmula:	$X = A/B$ A = Numero de cambios a funciones o módulos que tienen comentarios confirmados. B = Total de funciones o módulos modificados.

Interpretación:	$0 \leq X \leq 1$ Entre más cercano a uno, más registrable. 0 indica un control de cambios deficiente o pocos cambios y alta estabilidad.
Fuente de medición:	Sistema de control de configuraciones Bitácora de versiones Especificaciones.
Audiencia:	Desarrolladores Validador de contenidos Requeridores
<b>Resultado Legado</b>	<p><b>X =</b> En caso del aplicativo legado no se cuenta con una bitácora de versiones, ni la documentación adecuada; pero si indican que sufrió varios cambios.</p> <p><b>X = 0</b> (indica control de cambios deficientes)</p>
<b>Resultado Migración</b>	<p><b>X =</b> Para el caso del aplicativo migrado, al contar con 1.5 meses en uso, no se cuenta con modificaciones</p> <p><b>X = 0</b> (alta estabilidad)</p>

**Tabla 25: Resultado Facilidad de Mantenimiento – Cambiabilidad**

<b>Sub-Característica</b>	<b>Estabilidad:</b> Capacidad del producto software para evitar efectos inesperados debido a modificaciones de software.
Nombre:	Grado de impacto.
Propósito:	Determinar si existen efectos colaterales originados por el cambio de adecuación.
Método de aplicación:	Probar todas las funciones del software.
Medición, fórmula:	$X = 1 - A/B$ A = Numero de funciones ejecutadas incorrectamente. B = Número total de funciones del software.

Interpretación:	$0 \leq X \leq 1$ Entre más cercano a uno, menor impacto.
Fuente de medición:	Especificación Documento de gestión de cambios. Software. Casos de prueba.
Audiencia:	Usuarios finales Experto validador de contenidos Desarrolladores
Resultado Legado	X = No se cuenta con la documentación necesaria. X = 0
Resultado Migración	X = 1 - 6/38 X = 0.84

*Tabla 26: Resultado Facilidad de Mantenimiento – Estabilidad*

#### f. Portabilidad

Sub-Característica	<b>Adaptabilidad:</b> Es la capacidad del producto de software para ser adaptado a diferentes entornos especificados sin aplicar acciones o medios diferentes de los previstos para el propósito del software considerado.
Nombre:	Nivel del tiempo medio de respuesta.
Propósito:	Conocer el tiempo de respuesta promedio de las funciones del software en el nuevo entorno.
Método de aplicación:	Ejecutar una función de software y contabilizar el tiempo promedio y comparar contra el tiempo promedio del entorno anterior
Medición, fórmula:	$X = 1 - A/B$ A = Tiempo medio del entorno nuevo.

	B = Tiempo medio del entorno anterior.
Interpretación:	$0 < X \leq 1$ Entre más cercano a uno, mejor. El impacto puede no ser significativo.
Fuente de medición:	Diseño. Casos de prueba. Informe de revisión.
Audiencia:	Desarrolladores Experto validador de contenidos
Resultado Legado	X = El aplicativo legado solo puede ser ejecutado en un entorno de escritorio con SO Windows. X = 0
Resultado Migración	Para este caso, será probado en una máquina de escritorio y una Tablet. X = 1- 27.380 seg./31.750 seg. X = 0.14

**Tabla 27: Resultado Portabilidad – Adaptabilidad.**

Sub-Característica	<b>Coexistencia:</b> Es la capacidad del producto de software para coexistir con otros productos de software independientes dentro de un mismo entorno, compartiendo recursos comunes.
Nombre:	Grado de coexistencia.
Propósito:	Determinar los componentes software incompatibles con los ya existentes en producción.
Método de aplicación:	Contabilizar aquellos errores producidos por incompatibilidad de componente software en un ambiente homologado destinado a pruebas.

Medición, fórmula:	$X = 1 - A/B$ A = Errores de incompatibilidad en una prueba. B = Numero de pruebas totales.
Interpretación:	Mientras X este más cercano a uno, mejor.
Fuente de medición:	Software. Casos de prueba. Inventario de homologación
Audiencia:	Desarrolladores Experto validador de contenidos Arquitectos de Software
Resultado Legado	X = El aplicativo legado solo puede ser ejecutado en un entorno de escritorio con SO Windows. X = 0
Resultado Migración	Para este caso, será probado en una máquina de escritorio y una Tablet. X = 1- 0/1 X = 1

*Tabla 28: Resultado Portabilidad – Coexistencia.*

Sub-Característica	<b>Reemplazabilidad:</b> La capacidad del producto de software para ser utilizado en lugar de otro producto de software, para el mismo propósito y en el mismo entorno.
Nombre:	Grado de reemplazabilidad.
Propósito:	Determinar las funciones software que no cumplen el propósito de la función anterior del software.
Método de aplicación:	Contabilizar aquellos funciones que no lo logran el mismo propósito de la función anterior.

Medición, fórmula:	$X = 1 - A/B$ A = Funciones ejecutadas que no logran el propósito. B = Número Total de funciones ejecutadas.
Interpretación:	Mientras X este más cercano a uno, mejor.
Fuente de medición:	Especificación de requerimientos. Software. Documento de gestión de cambios.
Audiencia:	Desarrolladores Experto validador de contenidos Analistas
Resultado Legado	$X = 1 - 31/31$ $X = 0$
Resultado Migración	$X = 1 - 0/38$ $X = 1$

**Tabla 29: Resultado Portabilidad – Reemplazabilidad.**

Sub-Característica	<b>Cumplimiento de la portabilidad:</b> Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.
Nombre:	Conformidad de portabilidad.
Propósito:	Que tan conforme es la portabilidad del producto con regulaciones estándares y convenciones aplicables.
Método de aplicación:	Contar los artículos encontrados que requirieren conformidad y comparar con el número de artículos en especificación que requerirán conformidad.
Medición, fórmula:	$X = A/B$ A = Número de artículos implementados de conformidad.

	B = Número Total de artículos que requieren conformidad.
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a uno, más completa.
Fuente de medición:	Especificación de conformidad y estándares, convenciones y regulaciones relacionados..
	Diseño.
	Código fuente.
	Informe de revisión
Audiencia:	Desarrolladores
	Requeridores
Resultado Legado	X = El aplicativo legado solo puede ser ejecutado en un entorno de escritorio con SO Windows. X = 0
Resultado Migración	X = 38/38 X = 1

**Tabla 30: Resultado Portabilidad – Cumplimiento de la portabilidad.**

Tipo de Calidad	Característica	Migración	Legado	Viabilidad de Migración
<b>Funcionabilidad</b>	Aplicabilidad	1	0.77	SI
<b>Fiabilidad</b>	Madurez	1	0	SI
<b>Usabilidad</b>	Entendibilidad	0.89	0.7	SI
<b>Eficiencia</b>	Comportamiento en el tiempo	31.750 seg.	35.34 seg.	SI
<b>Facilidad de Mantenimiento</b>	Analizabilidad	2.38	3.57	SI
	Cambiabilidad	0	0	SI
	Estabilidad	0.84	0	SI
<b>Portabilidad</b>	Adaptabilidad	0.14	0	SI
	Coexistencia	1	0	SI
	Reemplazabilidad	1	0	SI
	Cumplimiento de la portabilidad	1	0	SI

*Tabla 31: Resultados de la ISO 9126*

## CONCLUSIONES

1. Es posible desarrollar una metodología para realizar la migración de aplicativos legados a aplicativos que cuenten con tecnología moderna sin alterar los procesos legados; y de esta manera lograr que grandes empresas tengan un soporte técnico de calidad.
2. El aplicativo migrado nos permite ahorrar en costos, respecto al mantenimiento y soporte técnico.
3. Ahorro de tiempo al desarrollar la migración, ya que se cuenta con un repositorio de buenas prácticas, debido a los problemas de compatibilidad de versiones entre el lenguaje de programación legado y migrado.
4. El uso de los procesos de calidad impuestos en la metodología, logran que la documentación en la gestión de la migración sea satisfactoria para el centro de desarrollo.
5. Se concluye que el aplicativo migrado con la metodología propuesta obtuvo niveles de aceptación altos en cuanto a la usabilidad, flexibilidad, reusabilidad y portabilidad.

## RECOMENDACIONES

- Se recomienda incentivar la utilización de esta metodología en otras fábricas de software que cuenten con clientes donde tengan un aplicativo legado.
- Se recomienda a futuro implementar un estándar de seguridad de la información a la presente metodología para garantizar la integridad de la información llevada en cada proceso.
- Esta metodología puede ser un buen punto de partida para la migración de bases de datos legadas.
- Esta metodología puede ser un buen punto de partida para la migración de aplicativos legados a aplicativos con tecnología moderna, usando software libre.
- Se puede proponer la creación del área llamada "Área de Migración de Aplicaciones" dentro de las empresas tecnológicas para el control y elaboración de nuestra metodología propuesta, ya que existen muchas empresas que cuentan con aplicativos legados.
- Es recomendable el uso de incentivos con los colaboradores, para la ejecución de dicha metodología y de esta manera lograr un mejor rendimiento del equipo de trabajo.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Bianchiotti, F. y Casas, S. (2014). *Guía para la Reingeniería de Sistemas Legados: Una Experiencia Práctica y Real*. Rio Gallegos, Argentina.
- [2] Trasobares, A. (2003). *Los sistemas de información: evolución y desarrollo. Proyecto social: Revista de relaciones laborales*. Zaragoza, España: UNIZAR.
- [3] Salvatierra, G. , Mateos, C. , Crasso, M. , Zunino, A., and Campo, M. (2013). *Legacy System Migration Approaches. Latin America Transactions*. Buenos Aires, Argentina: UNCPBA-CONICET
- [4] Rojas, N. (2007). *Conviviendo con sistemas legados*. Bogotá, Colombia: UNIANDES
- [5] Cáceres P. , Fernández D. y Uslenghi C. (2003). *Integración de aplicaciones legadas a través de la Web. Tercer congreso iberoamericano de telemática* .Recuperado de <http://cita2003.fing.edu.uy/articulosvf/97.pdf>
- [6] O'Callaghan, A.J. (1999). *Focus Issue on Legacy Information Systems and Business Process Change: Migrating Large-Scale Legacy Systems to Component-Based and Object Technology: The Evolution of a Pattern Language,* " *Communications of the Association for Information Systems*. Retrieved from <http://aisel.aisnet.org/cais/vol2/iss1/3>
- [7] Lloyd A., Dewar R. and Pooley R. (1999). *Legacy information systems and business process change: a patterns perspective,* "in *Communications of the association for information systems. Association for Information Systems*. Atlanta, USA: Journal Communications of the AIS.

- [8] Greenfield J., Short K. , Cook S. , Kent S. and Crupi J. (2004). *Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools*. Wiley, UK: Microsoft Corporation.
- [9] Mehta A. and Heineman G. (2001). *Evolving legacy systems features using regression test cases and components,*” in *Proceedings of the 4<sup>th</sup> International Workshop on Principles of Software Evolution*. New York, USA: IWPSE.
- [10] Sommerville I. (2002). *Ingeniería de Software*. Wesley, UK: UCV
- [11] Pinheiro P., Golgher P. and Laender A. (2001). *A simulation model for the performance evaluation when migrating legacy systems*. *5th IEEE European Conference on Software Maintenance and Reengineering*. Manchester, UK: IEEE
- [12] Álvarez J., Mateos M., y Moreno M. (2004). "*Metodología de reingeniería del software para la remodelación de aplicaciones científicas heredadas. Informe técnico DPTOIA-IT-2004-003*". Salamanca, España: USAL.
- [13] Pedraza G. (2008). *Evolución e Integración de Aplicaciones Legadas: Comenzar de Nuevo o Actualizar?* . Medellín, Colombia: UNIPILOTO
- [14] Sicilia M. (2009). *Técnicas de Mantenimiento de Software*. *OpenStax: Creative Commons*. Houston, EEUU: CNX
- [15] Escalera G., Masa C. y García E. (2008). *Implantación de la reingeniería por procesos, actividades, técnicas y herramientas*. Salamanca, España: Asociación Europea de Dirección y Economía de Empresa
- [16] Project Management Institute. (2008). *PMBOK® Guide - La pasión por la Ejecución*. Pennsylvania, EEUU: PMI.

- [17] Gramajo, E., García-Martínez, R., Rossi, B., Claverie, E., Britos, P. y Totongi, A. (1999). *Combinación de Alternativas para la Estimación de Proyectos Software*. Buenos Aires, Argentina: ITBA.
- [18] Escobar-Pérez, J. y Cuervo-Martínez, Á. *Validez de Contenido y Juicio de Expertos: una Aproximación a su Utilización*. Bogotá, Colombia: UNAL.
- [19] Mendoza S. (2005). *Criterio de expertos. Su procesamiento a través del método delphy*. La Habana, Cuba: SLD
- [20] Cusirramos R. (2012). *Diseño de un lenguaje de cuarta generación para la creación de aplicaciones web basado en el patrón MVC*. Peru: Concytec
- [21] Leon S. (2013). *Rompiendo el hielo con EF Code First*. Recuperado de <http://panicoenlaxbox.blogspot.pe/2013/04/romper-el-hielo-con-ef-code-first.html>
- [22] Bennett K. (1995). *Legacy Systems: Coping with success*. California, USA: IEEE.
- [23] Palacio J. (2006). *El Modelo Scrum*. Recuperado de <http://ww.navegapolis.net>
- [24] Deemer P. , Benefield G. , Larman C. y Vodde B. (2012). *Una introducción básica a la teoría y práctica de Scrum. Enterprise Software Development Series*. Zurich, Switzerland: Scrum- Institute.
- [25] Gonzales D. (2010). *Metodología en Sistemas – Reingeniería*. Buenos Aires, Argentina: UTN.
- [26] Monterroso E. (2002). *Reingeniería: un enfoque de todo o nada*. Buenos Aires, Argentina: UNLU
- [27] Salazar G. (2009). *Estimación de proyectos de software: Un caso práctico*.

.Medellín, Colombia: EAFIT.

- [28] Mendieta A. (2016). *El 40% de la población tiene acceso a internet. 2016, abril 04*. Recuperado de <http://peru21.pe/economia/40-poblacion-tiene-acceso-internet-2244173>
- [29] Tecnológico B. (2010). *¿Qué son los breadcrumbs?. 2010, junio 26, de Blog Tecnológico*. Recuperado de <http://www.blogtecnologico.net/que-son-los-breadcrumbs/>
- [30] Alegsa A. (2010). *Definición de Ingeniería inversa*. Recuperado de <http://www.alegsa.com.ar/Dic/ingenieria%20inversa.php>
- [31] Sepulveda B. (2014). *Reestructuración de Código. Secretaria de Educación Superior Tecnológico Nacional de México*. Recuperado de <http://dsc.itmorelia.edu.mx/pDSC/index.php/otros/binf/matd/24-rc>
- [32] Figueroa, M. (2005) *Calidad en la Industria del Software. La Norma ISO-9126*. Recuperado de <http://148.204.210.204/revistaupiicsa/34/34-2.pdf>

## ANEXOS

### ANEXO 1

#### Funciones del Aplicativo Legado por Modulo

<b>Funciones del Sistema Legado por Modulo.</b>	
<b>Empresa:</b>	<i>Nombre de la empresa del cliente</i>
<b>Código de Proyecto:</b>	<i>Nº de código del proyecto de migración</i>
<b>Fecha:</b>	<i>Fecha en la que se está registrando el formato</i>
<b>Responsable Legado:</b>	<i>Persona responsable por parte del equipo legado, el más recomendable es el experto Legado</i>
<b>Responsable Migración:</b>	<i>Persona responsable por parte del equipo legado, el más recomendable es el Experto Migración</i>
<b>Nombre de sistema Legado:</b>	<i>Nombre actual que posee el sistema legado</i>
<b>Modulo:</b>	<i>Nombre del módulo que se está revisando</i>
<b>Nivel de Criticidad:</b>	<i>La criticidad se manejará del 1 al 5; siendo 1 el menos crítico, y 5 el más crítico.</i>
<b>Funcionabilidad:</b>	<i>Explicar toda la funcionabilidad del modulo</i>
<b>Ejemplo de caso real:</b>	<i>Dar un ejemplo real de cómo se usa el modulo</i>
<b>Dependencias:</b>	<i>Listar los módulos que dependen de este.</i>
<b>Reportes:</b>	<i>Listar los reportes que genera el reporte</i>
<b>Detalle:</b>	<i>Detallar los campos que contiene el reporte</i>
<b>Errores actuales:</b>	<i>Listar los posibles errores que se tengan actualmente en el sistema legado</i>
<b>Mejoras:</b>	<i>Listar las posibles mejoras que se podrían dar a este módulo.</i>

## ANEXO 2

### Seguimiento de Proyecto de Migración - SPM

	Nivel	Descripción Tarea
<b>PROYECTO:</b>		
<b>ANÁLISIS</b>	1	
<b>INICIO</b>	1	
	1	<b>Revisión Alcance del REQ por el Analista</b>
	2	<b>INICIO</b>
	2	Revisión del Código de proyecto
	2	Revisión/Modificación de Matriz inicial
	2	Reunión con el Estimador
	2	<b>FIN</b>
	1	<b>Tarea Gestión del Proyecto</b>
	2	<b>INICIO</b>
	2	Solicitar Criticidad al Proyecto
	2	<b>FIN</b>
	1	<b>Marco Referencial Solución</b>
	2	<b>INICIO</b>
	2	Contexto General Actual(Técnica Funcional)
	2	Alternativas propuestas(Si existe, sino no incluir esta actividad)
	2	<b>FIN</b>
	1	<b>Marco Funcional-Técnico</b>
	2	<b>INICIO</b>
	2	Marco General del Proceso ( Esquema Actual y Esquema Nuevo)

	2	<b>Marco Funcional</b>
	3	<b>INICIO</b>
	3	<b>Procesos del Negocio(definir los procesos del negocio)</b>
	4	<b>INICIO</b>
	4	<b>Análisis</b>
	5	<b>INICIO</b>
	5	<b>Funcionalidad 1</b>
	6	<b>INICIO</b>
	6	Detalle 1
	6	Detalle 2
	6	<b>FIN</b>
	5	<b>Funcionalidad 2</b>
	6	<b>INICIO</b>
	6	Detalle 1
	6	Detalle 2
	6	<b>FIN</b>
	5	<b>Funcionalidad n</b>
	6	<b>INICIO</b>
	6	Detalle 1
	6	Detalle 2
	6	<b>FIN</b>
	5	<b>FIN</b>
	4	<b>FIN</b>
	3	Descripción y diagrama del proceso técnico
	3	<b>Descripción de las unidades de programación</b>
	4	<b>INICIO</b>
	4	UPI(Interfaz, Negocio, Componente)

	4	UP2(Interfaz, Negocio, Componente)
	4	UP.....N (Interfaz ,Negocio, Componente)
	4	<b>FIN</b>
	3	<b>FIN</b>
	2	<b>Marco Técnico</b>
	3	<b>INICIO</b>
	3	Diseño de Interfaces(si es necesario)
	3	<b>Definición de procesos tecnológicos</b>
	4	<b>INICIO</b>
	4	Descripción y Diagrama de Contexto
	4	<b>Consideraciones Técnicas</b>
	5	<b>INICIO</b>
	5	Consideraciones para evaluación de rendimiento
	5	Lineamientos y Consideraciones Generales
	5	<b>FIN</b>
	4	<b>FIN</b>
	3	Definición de seguridad y auditoría
	3	<b>Guía de Casos de Pruebas</b>
	4	<b>INICIO</b>
	4	Definición del Formato de Caso de Prueba - PUE
	4	Anexos(si es necesario)
	4	<b>FIN</b>
	3	<b>FIN</b>
	2	<b>Elaboración de Formatos de la Fase (Agregar/Eliminar los documentos que correspondan a la fase)</b>
	3	<b>INICIO</b>
	3	PDI

	3	PUE
	4	<b>INICIO</b>
	4	Prep. Ambiente Desarrollo: Solicitar-Preparar- Seguimiento
	4	Proceso de Revisión Interna(revisión de Pares - Ejecutar IQA)
	4	Proceso de Revisión del Experto( incluir revisión de CCI en esta fecha)
	4	Revisión del experto cliente
	4	Ejecutar CCI
	4	Transferencia DAM a Desarrollador: Reuniones y Coordinaciones con los involucrados.
	4	<b>FIN</b>
	3	<b>FIN</b>
	2	Levantar Observaciones
	2	Verificación de Observaciones
	2	DAM Aprobado
	2	<b>FIN</b>
<b>FIN</b>	1	
<b>CONSTRUCCIÓN</b>	1	
<b>INICIO</b>	1	
	2	<b>INICIO</b>
	2	<b>Codificar las Unidades de Programación</b>
	3	<b>INICIO</b>
	3	<b>Funcionalidad 1</b>
	4	<b>INICIO</b>
	4	<b>Módulo A</b>
	5	<b>INICIO</b>

	5	UP1
	5	UP2
	5	<b>FIN</b>
	4	<b>FIN</b>
	3	<b>Funcionalidad n</b>
	4	<b>INICIO</b>
	4	UP1
	4	<b>FIN</b>
	3	Modificar DAM
	3	Pruebas Unitarias: Preparar Ejecutar - PUE
	3	<b>FIN</b>
	2	<b>Pruebas de Integración: Apoyar - Ejecutar casos</b>
	3	<b>INICIO</b>
	3	Funcionalidad 1
	4	<b>INICIO</b>
	4	Caso de Prueba 1
	4	Caso de Prueba 2
	4	<b>FIN</b>
	3	Funcionalidad 2
	3	Funcionalidad n
	3	<b>FIN</b>
	2	<b>Elaborar/Actualizar Documentación y Formatos (Detallar todos los documentos de la fase)</b>
	3	<b>INICIO</b>
	3	MIM
	3	PDI
	3	<b>FIN</b>

	2	Ejecutar CCI
	2	Ejecutar CCE
	2	<b>Prep. Ambiente Certificación./Congelamiento: Solicitar 1er Ciclo</b>
	3	<b>INICIO</b>
	3	Pasar CheckList de validaciones iniciales
	3	Solicitud de congelamiento.
	3	Apoyo al proceso de congelamiento
	3	<b>FIN</b>
	2	<b>FIN</b>
<b>FIN</b>	1	
<b>CERTIFICACIÓN</b>	1	
<b>INICIO</b>	1	
	1	Actualizar MIM
	1	Actualizar CDD
	1	Soporte a Certificación
<b>FIN</b>	1	
<b>IMPLEMENTACIÓN</b>	1	
<b>INICIO</b>	1	
	1	Preparar Plan de Instalación preliminar - PDI
	1	Revisión y aprobación del Plan de Instalación - PDI
	1	Ejecución de actividades (preparación de servidores, ambientes)
	1	Anunciar Proyecto
	1	Pase a producción
	1	Ratificación en producción
	1	Seguimiento post producción
<b>FIN</b>	1	

### ANEXO 3

#### ACR-Acta de Reuniones

ACTA DE REUNIONES	
<b>Grupo:</b>	<i>Acta Nro</i>
<b>Citada por:</b>	<b>Fecha:</b>
<b>Coordinador:</b>	<b>Hora inicio:      Fin:</b>
<b>Secretario:</b>	<b>Lugar:</b>

PARTICIPANTES			
No.	Nombre	Cargo	Teléfono
1			

PUNTOS DE DISCUSION	
1	

DESARROLLO DE LA REUNIÓN
<b>Observaciones:</b>

CONCLUSIONES				
No	Tarea	Responsable	Período de cumplimiento	Observaciones

## ANEXO 4

### CheckList de Planificación

CheckList de planificación	
<b>Datos Generales</b>	
1.	Nombre de la Aplicación:
2.	Gerencia a la que pertenece:
3.	Área del a la que pertenece el aplicativo:
4.	Unidad a la que pertenece el aplicativo:
5.	Líder Usuario del aplicativo:
6.	Especialista del aplicativo:
7.	Anexo del especialista:
<b>Datos de Encargados</b>	
1.	Nombre del Analista de Seguridad:
2.	Nombre(s) de los encargado(s) de actualizar la Matriz de Roles:
3.	Indicar quiénes tendrán acceso a Servidores de Producción:
a)	Especialista
b)	Proveedor
c)	Especialista y Administrador de BD
d)	Otros (Especificar)
<b>Datos Técnicos</b>	
1.	¿Se cuenta con la última versión de las fuentes del aplicativo? ¿Cada cuánto tiempo pueden cambiar las fuentes?(Sin password) Si/No. _____.
2.	¿Cuáles son los Inputs (archivos planos, Excel, base de datos)? Considerar si la Base de Datos esta con datos o vacía. Si/No. _____.
3.	¿Cuáles son los Outputs (archivos planos, Excel, base de datos)? Considerar si la Base de Datos esta con datos o vacía. Si/No. _____.
4.	¿Se tiene el conocimiento completo de la Arquitectura actual (nombre de los servidores o PC involucrados)? Si/No. _____.
5.	¿Existe alguna estructura de carpetas para la aplicación? Indicar la estructura.

---

6. Esta aplicación maneja perfiles. ¿Cuántos? Enumerar los perfiles.

---

7. ¿Está claro el proceso del negocio sobre la cual actúa la aplicación?

---

8. ¿El aplicativo cuenta con un sistema de backup (BD)?

---

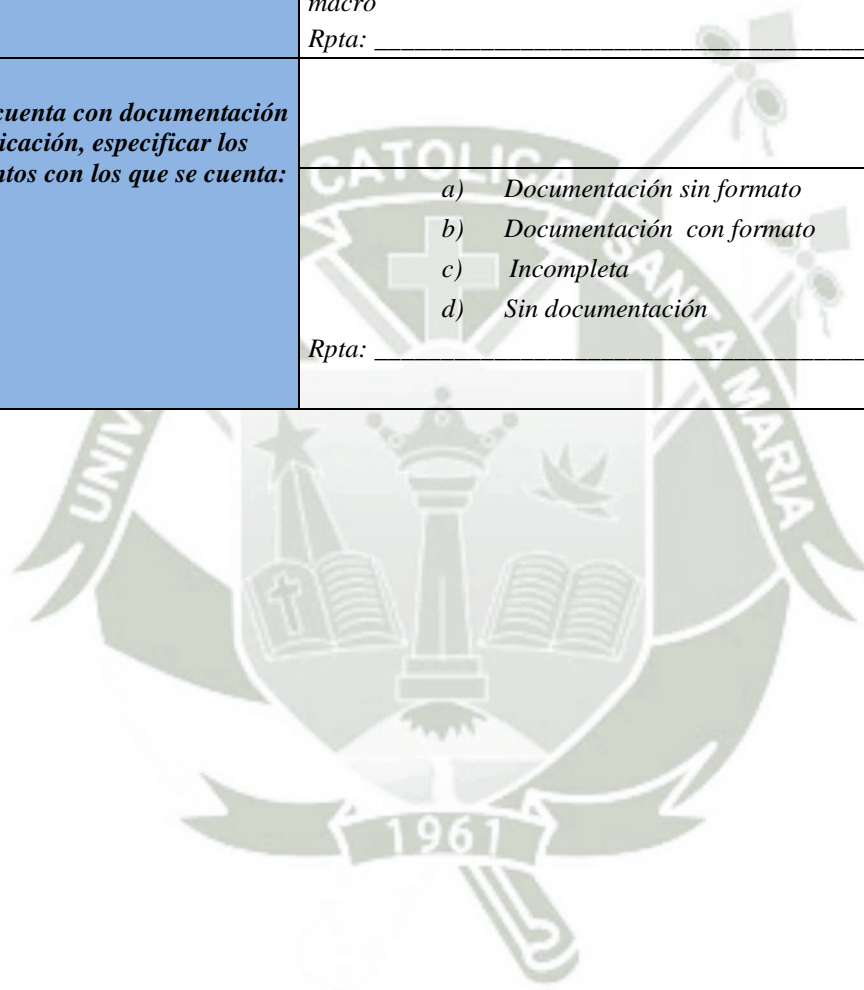


## ANEXO 5

### CheckList de Migración


<i>CheckList de migración</i>	
<b>1. Nombre de Aplicación:</b>	
<b>2. Gerencia:</b>	
<b>3. Área:</b>	
<b>4. Unidad de Negocio:</b>	
<b>5. Líder Usuario:</b>	
<b>6. Usuario Especialista:</b>	
<b>7. Anexo del Especialista:</b>	
<b>8. Tecnología de aplicación</b>	
	a) VBA - Macros (Especificar Versión Excel) b) Otro (Especificar) Rpta: _____
<b>9. Estado de Aplicación</b>	
	pendientes a) Macro en funcionamiento b) Macro en funcionamiento con mejoras c) Macro con errores por corregir d) Macro en desuso e) Otro (Especificar) Rpta: _____
<b>10. La aplicación necesita como información entrante, elegir uno o varios:</b>	
	a) Del mismo Excel b) Archivos Planos de la misma PC c) Archivos Planos de ruta compartida(File Server) d) Base de Datos(SQL, Access) e) Ninguno f) Otro(Especificar) Rpta: _____
<b>11. La aplicación genera información de salida, elegir uno o varios:</b>	
	a) Al mismo Excel b) Archivos Planos a misma PC c) Archivos Planos en ruta compartida(File Server) d) Base de Datos(SQL, Access) e) Ninguno f) Otro(Especificar)

	Rpta: _____
<b>12. El desarrollo de la aplicación la realizó:</b>	
	<ul style="list-style-type: none"> <li>a) Propio(BCP)</li> <li>b) Proveedor</li> <li>c) Ambos</li> </ul>
	Rpta: _____
<b>13. Conocimiento de la aplicación</b>	
	<ul style="list-style-type: none"> <li>a) Se cuenta con personal que desarrolló la macro</li> <li>b) Se cuenta con personal que conoce la macro</li> <li>c) No se cuenta con conocimientos técnicos de la macro</li> </ul>
	Rpta: _____
<b>14. Se cuenta con documentación de la aplicación, especificar los documentos con los que se cuenta:</b>	
	<ul style="list-style-type: none"> <li>a) Documentación sin formato</li> <li>b) Documentación con formato</li> <li>c) Incompleta</li> <li>d) Sin documentación</li> </ul>
	Rpta: _____



## ANEXO 6

### ChekList de Seguimiento de Proyecto de Migración

ChekList de Seguimiento de Proyecto de Migración - SPM				
<b>Código de Proyecto:</b>				
<b>Empresa de Cliente:</b>				
<b>Líder de Proyecto:</b>				
<b>Fecha de revisión</b>				
<b>Analista:</b>				
Ítem de Verificación	Criterio	Conforme?	Comentario (ML/PL)	Descargo
1.- Nomenclatura	¿El nombre de documento es el correcto?, ¿Cumple con el estándar? << SPM - N° de proyecto>>			
	¿La actividad raíz tiene la nomenclatura estándar <Nombre del proyecto><Código de proyecto>			
2.- Contexto General del proyecto	Se tienen TODAS las columnas estándar, tales como: - Requisito funcional - Nombre de la tarea - % Completado - Duración - Comienzo			

	- Fin			
	- Comienzo real			
	- Fin real			
	- Duración real			
	- Predecesores			
- Nombres de recursos				
- Comentarios				
ANALISTA/DEVELOPER				
- Comentarios REVIEWER				
- Comentarios Líder de proyecto, experto migración				
¿Todas las tareas están escritas consistentemente y con el nivel apropiado de detalle sin conflicto entre ellas?				
¿Cada tarea tiene especificada:  - Duración: Esto implica que en la duración no se tengan periodos con "?"  - Fechas de inicio y fin válidas (mes y año correctos)				
¿El cálculo de fechas se ajusta con las fechas de inicio y fin de cada fase del proyecto?				
¿Se tiene especificado el nombre del recurso que atenderá cada tarea? (Las tareas de CCI,CCE pueden no tener un recurso				

		asignado, esto dependerá de la planificación del Líder de proyecto)			
3.- Fase del requerimiento	3.1.- Estimación	Se incluyeron tareas de Control de Calidad.			
		Se incluyen tareas de Gestión y seguimiento			
		Se incluyen tareas de Congelamiento y Pase a producción			
		Se incluyen tareas de apoyo a Certificación			
	3.2.- Análisis	¿Las tareas referidas a las especificaciones del requerimiento (no tareas generales) se encuentran relacionadas al requisito funcional u otra sección del DAM.			
		¿Se tienen indicadas tareas de?: - Revisión de alcance - Marco funcional-técnico - Casos de prueba - Documentación			
		¿Están especificadas las tareas de revisiones de control de calidad?			

		¿Las tareas referidas a las especificaciones del requerimiento (no tareas generales) se encuentran relacionadas al requisito funcional u otra sección del DAM.			
	3.3.- Construcción	¿Se tienen indicadas tareas de?: - Código del proyecto. - Dentro de cada código deben indicar las funcionalidades esperadas del requerimiento. - Pruebas unitarias - Pruebas de integración. Dentro de estas también deben indicarse las funcionalidades esperadas del requerimiento - Documentación - Preparación de ambiente de certificación / congelamiento			
		¿Se tienen indicadas tareas de certificación?			
		¿Se tienen indicadas tareas de pase a producción?			
		¿Están especificadas las tareas de revisiones de control de calidad?			

## ANEXO 7

### Bloque de Preguntas

ChekList de Seguimiento de Proyecto de Migración - SPM						
Código de Proyecto:						
Empresa de Cliente:						
Líder de Proyecto:						
Fecha de revisión						
Analista:						
Consulta	Planteada por:	Fecha de Consulta	Respuesta o Solución	Contestada por:	Fecha de respuesta	Comentarios



## ANEXO 8

### Documento de Análisis de Migración – DAM

### HOJA DE CONTROL

<b>Empresa</b>			
<b>Proyecto</b>	<Nombre Proyecto>		
<b>Entregable</b>	<Código del proyecto de migración>		
<b>Autor</b>	DAM - Documento de Análisis de Migración		
<b>Aprobado por</b>		<b>Fecha Aprobación</b>	DD/MM/A AAA
		<b>N° Total de Páginas</b>	

#### REGISTRO DE CAMBIOS

<b>Versión</b>	<b>Causa del Cambio</b>	<b>Responsable del Cambio</b>	<b>Fecha del Cambio</b>
0.1	Versión inicial	<Nombre Apellido1 Apellido2>	DD/MM/AA AA

#### CONTROL DE DISTRIBUCIÓN

<b>Nombre y Apellidos</b>
<Nombre Apellido1 Apellido2>

### Acta de Compromiso

Documento de Análisis de Migración (DAM)

[Nombre y número del requerimiento]

El presente Documento de Análisis y Diseño (DAM) del requerimiento [Nombre y número del Proyecto] representa el análisis y diseño del requerimiento elaborado por el Líder de Proyecto.

El Analista es responsable de informar al Experto de Contenidos los cambios en el aplicativo plasmados en el DAM, con el fin que el experto de contenidos identifique y envíe los lineamientos de las pruebas unitarias y los casos de prueba a proponer en la certificación.

El Analista es responsable que el documento sea leído y revisado por todos los implicados en el desarrollo y construcción del requerimiento.

El/los elaboradores(es) del documento se comprometen a seguir el procedimiento de control de cambios en el caso que se realice un cambio al documento.

**Analista(s)** (elaborador(es) del DAM)

**Aprobación del DAM:(\*) Escenarios:**

- Cuando el Usuario Solicitante del pedido provenga del Área de Sistemas y sea el Requerimiento de tipo Técnico.
- Cuando el Requerimiento solicitado por un Usuario impacta el desarrollo previo de un Requerimiento Técnico, en este caso el aprobador es el Jefe de Equipo que solicitó el desarrollo al otro equipo de trabajo.

**Rol Aprobador:**

Rol del DAM	Aprobador	Nombres	Capítulo / Sección del DAM	Fecha de Aprobación	Incrustar documento(s) de Conformidad (E-mail o Acta):  1. De aprobación 2. De asignación a nuevo rol
Líder Usuario y/o Rol asignado					

**Historial de versiones**

La siguiente tabla describe la historia de modificación del entregable del proyecto, del más reciente al más antiguo, para propósitos de seguimiento

Versión (V.X)	Fecha dd/mm/aaaa	Modificaciones	Modificado por

## Glosario de Términos

Término	Definición

**NOTA IMPORTANTE:** Todas las secciones del DAM deben ser llenadas. En ningún caso debe usarse la frase "No Aplica". Pueden usarse frases tales como "No hay cambios", "No hay impacto en esta sección", "La solución que se está implementando no tiene impacto en esta sección", etc.

### 1. Trazabilidad

Si es un requerimiento de tipo funcional y técnico, las cuatro primeras columnas del cuadro de trazabilidad son registradas desde el análisis funcional, las columnas restantes (funciones principales y tipo) se llenarán durante el Análisis.

Si es un requerimiento de tipo técnico, indicar en la columna Requisito Funcional que "No hay Impacto en esa sección", asimismo las tres primeras columnas del cuadro (Proceso de Negocio, Funcionalidades, Proceso Ad-Hoc) podrán ser mapeadas, las columnas restantes (Funciones Principales y Tipo de Funciones) se registrarán durante el análisis Técnico.

- **Tipo de Funciones:**
  - Nuevas: Seleccionar si las nuevas funciones se incluirán en el componente, programa, módulo, registros, etc.
  - Consumidas/Modificadas/Eliminadas: Seleccionar si las funciones actuales del componente, programas, módulo, registros tienen cambios.

Ejemplo:

Proceso de Negocio	Funcionalidades	Proceso Ad-Hoc	Requisito Funcional	Funciones Principales	Tipo Funciones
[Proceso 1]	Sección Técnica - Aplicativo- Funcionalidad 1-Detalle	Proceso Ad-Hoc 1	[Requisito Funcional 1.1] [Requisito Funcional 1.2]	Describir las funciones, programas, registros, módulos, métodos que impacta por Req. Funcional y/o Proceso de Negocio /Ad-Hoc.	Nueva, Modificada Eliminada, Consumida.
[Proceso 2]	Sección Técnica - Aplicativo- Funcionalidad 2-Detalle	Proceso Ad-Hoc 2	[Requisito Funcional 2.1]		
[Proceso 3]	Sección Técnica - Aplicativo- Funcionalidad 3-Detalle	Proceso Ad-Hoc 3	[Requisito Funcional 3.1]		
[Proceso 4]	Sección Técnica - Aplicativo- Funcionalidad 4-Detalle	Proceso Ad-Hoc 4	[Requisito Funcional 4.1]		
[Proceso N]	Sección Técnica - Aplicativo- Funcionalidad N-Detalle	Proceso Ad-Hoc N	[Requisito Funcional N]		

## 2. Contexto General de la Solución

### 2.1 Evaluación Técnica Preliminar

Describir en forma general las posibles alternativas de solución preliminares, seleccionar en base a los siguientes criterios:

Costo, performance, complejidad, crecimiento de la funcionalidad, limitaciones tecnológicas, sensibilidad a los cambios, riesgo, discontinuación del aplicativo, capacidades y limitaciones de los Usuarios finales y operadores.

- **Alternativa 1:**
- **Alternativa 2:**
- **Alternativa n:**

#### Conclusiones:

Describir las principales conclusiones derivadas de las evaluaciones técnicas preliminares.

En caso se indique una sola alternativa, justificar el motivo de porqué se propuso sólo una.

### 2.2 Funciones Principales

Describir las funciones, programas, registros, módulos, métodos que impacta por requisito funcional y/o Proceso de Negocio /Ad-Hoc.

#### Tipo de Funciones:

- **Nuevas:** Seleccionar si las nuevas funciones se incluirán en el componente, programa, módulo, registros, etc.
- **Consumidas/Modificadas/Eliminadas:** Seleccionar si las funciones actuales del componente, programas, módulo, registros tienen cambios o re-uso.

Función Principal	Tipo de Funciones	Descripción Funciones Principales
	Nuevas/Modificadas/ Eliminadas/Consumidas	

### 3. Descripción del Proceso

Describir el contexto general de las funcionalidades que permita entender el flujo que se seguirá en el aplicativo y mostrar gráficamente como se dan las relaciones entre ellos.

#### 3.1 Proceso de Negocio /Proceso Ad-Hoc

En esta parte se describirá el proceso ad-hoc mediante un flujograma que sirva de contexto a las funcionalidades del aplicativo, para lo cual se considerará las actividades de las unidades responsables y su interacción con los aplicativos. Tener en cuenta que no necesariamente todos los procesos son automáticos. Bosquejar las entradas y salidas por cada proceso. De no existir un proceso representar un diagrama de bloques. **Indicar el impacto de los nuevos requisitos funcionales en el proceso.**

##### 3.1.1 Desarrollo de los Requisitos Funcionales

###### 3.1.1.1 Requisito Funcional

###### Descripción

Describir la funcionalidad que se va a desarrollar, haciéndola explícita mediante el uso de flujogramas o casos de uso.

###### Interfaces

Describir las características de las interfaces para el requisito funcional. El nivel de detalle de deberá permitir entender el objetivo general de cada interface, así como los datos involucrados en el uso de dicha interface.

###### Requisitos de seguridad

Esta sección se completará sólo si existen características de seguridad específicas para la funcionalidad en mención. De lo contrario deberá mantenerse la sección e indicar “No presenta características específicas de seguridad”.

###### Pantallas/ Reportes

a) [Pantalla1/Reporte1]

###### **Funcionalidades: Escenarios y flujo de eventos**

Colocar los escenarios y secuencia de navegación entre pantalla (reporte) y pantalla (reporte) o entre pantalla (reporte) y algún otro componente de la interfaz gráfica de usuario.

### Diseño

Colocar el diseño de la pantalla (reporte). El formato a utilizar dependerá de la complejidad y esfuerzo asociado al diseño de la pantalla (reporte).

#### Campos y controles nuevos en la pantalla / reporte

Grupo / Secciones de la pantalla	Nombre del Campo/Control (Botón, tecla, link)	Tipo de Dato	Longitud (Máx y Min)	Descripción del campo (Teclas de función y/o controles)	Reglas y/o Validaciones
Datos Básicos del Usuario	Fecha de Activación	Date	8	Fecha de activación de cuenta del usuario.	No puede ingresar una fecha mayor a la actual

### Mensajes de error

Describir los mensajes de error que el sistema emitirá en la ejecución de cada requisito funcional descritos en los escenarios anteriores. Puede ser una lista de ellos o los gráficos de cómo se mostrarán; en ambos casos se debe incluir números de mensaje.

Dichos mensajes de error deberán colocarse de manera exacta de cómo el sistema los mostrará.

Ejemplo: En caso algún campo obligatorio no estuviera lleno se mostrará el mensaje de error/advertencia. Antes de grabar solicitará al usuario que confirme si está seguro de proceder.

Tipo	Mensaje de Error	Descripción del Error
Advertencia / Informativo / Error	Título del Mensaje	Detalle del significado del Mensaje de Error

b) [Pantalla n/Reporte n]

### 3.1.2 Descripción y diagrama del proceso técnico

El objetivo de este punto es identificar a través de flujograma, diagramas de bloque u otro diagrama de diseño, los cambios asociados a cada proceso ad-hoc.

Para este fin se debe describir cada uno de los procesos afectados, sus entradas y salidas (ó resultados) así como la secuencia de ejecución de cada uno de ellos.

Para el caso de requerimientos en los que se cree/modifique/elimine elementos como parte de un proceso de depuración automática de la aplicación (archivos, BD, Logs, etc.) se deberá describir el flujo planteado, detallando las necesidades propias en el que se basa el flujo para la depuración (Por fecha, por estado, etc).

### 3.1.3 Descripción de las unidades de programación

#### Descripción general de cada unidad

Describir el objetivo de cada unidad de programación agregada o modificada, su lógica de proceso en términos generales, sus funciones principales y las entradas / salidas de cada una de ellas.

Nombre del Elemento o Programa	
Descripción del cambio	
Entradas	
Lógica de programación (pseudocódigo)	
Salidas	
Interfaces usadas	
Otras consideraciones	Sección específica para describir aspectos relevantes de la implementación de ésta unidad de programación.  Asimismo se podrá incluir el responsable del elemento (puede ser una empresa proveedora).
Requisito (s) Funcional(es) que se implementa.	3.1.1.1, 3.1.1.n...

Otros procesos Ad-Hoc impactados	3.2, etc.
Funciones generales que se implementan en la Sección 2	Registrar las funciones nuevas, modificadas, consumidas o eliminadas que impactan a cada unidad de Programación.
Alertas y Mensajes de Retorno	En esta sección se especificarán los diferentes mensajes de errores técnicos (propios de la aplicación, del canal, Back End, etc) y definición de tipos de alertas.

Adicionalmente si una o más unidades de programación implementan varios requisitos funcionales, el analista deberá referenciarlos en la tabla, campo “Requisito(s) Funcional(es) que se implementa”.

Los elementos nuevos de aplicaciones no estandarizadas, deben estandarizarse. En este caso completar 2 filas en el cuadro adjunto, una para el elemento “X” actual con estado “eliminar” y otra para el elemento “X” estandarizado con estado “nuevo”.

Elementos	Sistema	Subsistema	Tipo	Estado (nuevo/modificado/ a eliminar de Producción)

#### 4. Diseño de Interfaces

Describir el objetivo de cada interfaz ya sea creada, modificada o consumida. Por su forma de procedimiento puede ser en línea o que corra un proceso a una determinada hora.

Tipo de interfaz	Link del servicio
On Line	
Batch	

#### 5. Definición de procesos tecnológicos

##### 5.1 Descripción y Diagrama de Contexto

Describir los Componentes, Interfaces y elementos que forman parte del contexto general de la solución y mostrar gráficamente como se dan las relaciones entre ellos.

## 5.2 Consideraciones Técnicas

### 5.2.1 Consideraciones para evaluación de rendimiento

#### 5.2.1.1 Tipo de prueba:

¿Qué tipo de Prueba se requiere para este aplicativo?
<p>Prueba de Carga <input checked="" type="checkbox"/> [ x ]</p> <p>Se realiza para comprender el comportamiento de una aplicación ante una carga determinada. Esta carga puede ser el número de usuarios esperado o un número de transacciones durante un tiempo determinado. El resultado de esta prueba nos dará el tiempo de respuesta de todas las transacciones críticas.</p>
<p>Prueba de Estabilidad <input type="checkbox"/> [ ]</p> <p>Se realiza con el fin de determinar si la aplicación puede mantener la carga esperada de manera continua y durante un largo tiempo. El objetivo principal de este tipo de pruebas es verificar que no existen fugas de memoria o procesos que pierdan rendimiento en un cierto periodo de tiempo.</p>
<p>Prueba de Stress <input type="checkbox"/> [ ]</p> <p>Se realiza para someter a la aplicación al límite de su funcionamiento mediante la ejecución de un número de usuarios muy superior al esperado. Tiene como finalidad el determinar la robustez de una aplicación cuando la carga es extrema y determinar las alarmas del sistema entre otras cosas.</p>

#### 5.2.1.2 Definiciones de la Aplicación

General: (Breve descripción de la aplicación que será validada a través de pruebas técnicas.)	
¿Esta aplicación se encuentra en producción?	<p>SI <input type="checkbox"/> [ ]</p> <p>NO <input checked="" type="checkbox"/> [ x ]</p>
¿En qué fase del desarrollo de la aplicación se encuentra?	<p>En desarrollo <input type="checkbox"/> [ ]</p> <p>En certificación <input checked="" type="checkbox"/> [ x ]</p> <p>En producción sin usar <input type="checkbox"/> [ ]</p> <p>Piloto en producción <input type="checkbox"/> [ ]</p> <p>En producción <input type="checkbox"/> [ ]</p> <p>Consultoría <input type="checkbox"/> [ ]</p>
¿Monitorización de operaciones es requerida?	<p>SI <input type="checkbox"/> [ ]</p> <p>NO <input checked="" type="checkbox"/> [ x ]</p>

Descripción de eventualidades, situación actual y los problemas presentados				
¿Cuál es la expectativa de usuarios concurrentes en toda la aplicación?	Número de usuarios simultáneos durante la prueba de rendimiento		Número de usuarios máximo que soporta	
	100		600	
¿Cuál es la expectativa transaccional?	Transacciones en un seg:	Transacciones por hora pico: (indicar el tiempo para llegar a dicha cantidad de operaciones)	Transacciones en un seg:	
	10	2000 en 50 minutos	10	
Listar los procesos de negocio más usados o más lentos y sus características en tiempo de respuesta.	Procesos o Funcionalidades de Negocio		Tiempo de Respuesta actual en Producción	Tiempo de Respuesta Esperado en Certificación
	SISML- REGUSUBD		1.5	2.5

5.2.1.3 Aplicación con el Cliente

General:(Breve descripción del método que se usara para comunicarse con la aplicación)	
¿El cliente usa un browser u otro método para comunicarse con la aplicación?	Web(HTTP/HTML) [ ]
	Web Service [ x ]
	Terminal Emulation [ ]
	MQ [ ]
	Cliente Servidor [ ]
	Otro [ ]

5.2.1.4 Datos de prueba (Llenar la sección que corresponda):

5.2.1.4.1 Número de Usuarios y perfil necesario

Procesos o Funcionalidad de Negocio	Números de Usuarios	Perfil del usuario
SISML- REGUSUBD	100	Usuario

**5.2.2 Lineamientos y Consideraciones Generales**

a) Describir los lineamientos generales bajo los cuales se deberán diseñar y construir los procesos, (Ej. Agrupar funciones automáticas a desarrollar que se realizan en forma repetitiva que puedan ser preparadas para convertirse en procesos genéricos y reusables), consideraciones propias de la plataforma en la que trabajará la solución, restricciones, consideraciones para la performance de cada proceso.

b) Homologación de servidores: indicar nombres de servidores de Desarrollo, Certificación y Producción.

S. Desarrollo	S. certificación	S. Producción

c) Indicar información para uso de rutinas generales: códigos y nombres de servicios/interfaces que usará la aplicación y códigos y nombres del canal o aplicativo que los invocará

d) Describir las Instrucciones/criterios/ consideraciones especiales para la creación de los ambientes de Prueba

e) Describir los lineamientos técnicos que se deberán tener en cuenta para el pase a producción. Algunas consideraciones:

- Indicar la secuencia de la Ejecución de los formatos.
- Indicar los módulos o elementos programables que deberán pasar en días consecutivos, el mismo día del pase y al día siguiente o en diferentes horarios.
- El pase no puede ser en quincena ni fin de mes.
- Para Cliente Servidor, en caso se requiera una distribución de la versión se debe indicar el plan de dicha distribución, el horario de activación y si es necesario completar la distribución para dar como atendido el pase.
- En caso de reversión se debe indicar explícitamente los elementos que deben revertirse obligatoriamente.
- Se debe indicar todas las actividades necesarias para el pase, incluyendo aquellas acordadas por excepción con Control de Calidad.

**6.- Índice de Complejidad**

El objetivo de esta sección es levantar la información de las variables de complejidad impactadas con el presente requerimiento, en este sentido, en la tabla adjunta debe registrarse la información referida a las “Interfaces”.

Consideraciones:

En caso de que el requerimiento no tenga ningún impacto en las variables de complejidad definidas, no será necesario efectuar algún registro en la tabla adjunta.

Esta tabla debe ser llenada sólo si existe eliminación de una interfaz o eliminación del consumo de una interfaz



## INTERFACES

Código de proyecto	Cliente	Proveedor	Interfaz ( Online o Batch)	Código de la interfaz consumida	Tipo de Interfaz	Nombre de la Interfaz	Descripción de la interfaz	Nombre Físico (Archivo)	Nombre Corto/Transacción	Tipo de Comunicación		Tipo de Actualización

## 7. Definición de seguridad y auditoría

### 7.1 Seguridad e integridad de los procesos

- ✓ Descripción de controles en la secuencia del Procesos Programados.
- ✓ Descripción de controles para la transmisión y recepción de archivos.
- ✓ Descripción de métodos de sincronización de transacciones en línea.
- ✓ Establecer puntos de control en el flujo de proceso de la aplicación. Identificar fuente de tiempo: fecha y hora del proceso principal (time stamping).
- ✓ Descripción de procesos de recuperación de archivos y BD.
- ✓ Definición de mantenimiento de datos y copias de respaldo.

### 7.2 Administración de perfiles de acceso.

Definir y describir los módulos vs. Perfiles de acceso a la aplicación.

- ✓ Descripción del módulo de seguridad (Ejm: Authorization Manager u Otro).
- ✓ Descripción de perfiles.
- ✓ Diseño perfiles de usuario
- ✓ Completar la Plantilla de Seguridad de la Aplicación.
- ✓ En caso de no utilizar los mecanismos estándar de autorización, incluir lo siguiente:

Descripción del Módulo de Seguridad.  
 Longitud mínima de password.  
 Número de intentos de acceso fallidos antes de bloquear el acceso.  
 Número de veces que se puede repetir un password.  
 Otras políticas de password implementadas.  
 Confirmación de password al crear cuentas y resetear.  
 Existencia de opción deshabilitar cuenta y/o eliminar.  
 Métodos de encriptación de passwords.  
 Número de estaciones por usuario que pueden acceder a la vez.

### 7.3 Controles de acceso

Definir y describir los Usuarios genéricos (no aplicativo), como usuarios del dominio, BD, mail's, directorios compartidos, etc.

- Identificación de permisos. Ej. Sistema operativo, tablas, stored procedures, programas y transacciones Host a proteger, etc.
- Identificación de usuarios especiales y su necesidad: Ej. System Administrador, etc.
- Identificación de programas que manejan elementos confidenciales. Ej. Rutinas de encriptación, etc.
- Identificación de conexiones a redes externas: Si se requiere apertura o retiro de puertos en time

### 7.5 Auditoría

- Auditoría en disco: Nombre y ubicación del archivo.
- Tamaño de Logs, alertas y procedimientos de limpieza (Ej. backup's)
- Tipos de eventos a registrar:
  - Registro de logons.
  - Registro de bloqueos de acceso.
  - Registro de intentos de acceso inválidos.
  - Registro de modificaciones a la información (validar la necesidad de registrar operaciones de consultas).

- Registro de acciones realizadas por los administradores.
- Registro del origen de la transacción (IP). Considerar también el código matrícula del usuario/ operador que originó el cambio, fecha y hora.

Consideración:

En caso se implemente un servidor nuevo, para la configuración de Software base de dicho servidor, en la siguiente ruta se especifican las mejores prácticas de Seguridad informática por producto a tener en cuenta, las cuales consideran los siguientes tipos de configuración:

- Actualización de cuentas de usuario creadas durante la instalación.
- Definición de roles y usuarios administradores.
- Parches de seguridad.
- Protección de las comunicaciones.
- Auditorías.
- Des-habilitación de usuarios y servicios no Utilizados.
- Autenticación y autorización sobre recursos críticos.
- Otras configuraciones especiales, de acuerdo al sistema operativo y servicios instalados.

**8. Guía de Casos de Prueba**

**8.1 Registro y Exportación de Casos de Pruebas**

- Registrar los casos de pruebas funcionales y técnicos en la plantilla Casos de Pruebas, finalizado el registro adjuntar la plantilla en ésta sección del documento.

ID	CASO DE PRUEBA	DESCRIPCIÓN	FECHA	AREA FUNCIONAL / TÉCNICA	FUNCIONALIDAD

**ANEXOS**

Nro.	Descripción	Detalle
1	PUE - Pruebas Unitarias Especificas	Son creadas por el analista y ejecutadas por el desarrollador.

Nota: En caso de existir algún anexo necesario agregar a este documento.

## ANEXO 9

### Pruebas Unitarias Especificas – PUE

PUE - Pruebas Unitarias Especificas				
<b>Nombre de Proyecto</b>				
<b>Código de Proyecto</b>				
<b>Nombre de Analista</b>				
<b>ID de Prueba</b>		<b>Version No.</b>		<b>Referencia</b>
<b>Elemento de prueba</b>				
<b>Ambiente</b>				
<b>Preparado por:</b>		<b>Fecha de Preparación:</b>		
<b>Verificación de Control de Calidad - Análisis</b>		<b>Fecha de Revisión:</b>		
<b>Programa desarrollado por:</b>				
<b>Programa probado por:</b>		<b>Fecha de revisión:</b>		
<b>Verificación de Control de Calidad - Construcción</b>		<b>Fecha de verificación:</b>		
Datos iniciales para realizar las pruebas				
Para:				
Archivos a usar:				
Pre-requisitos:				
Caso de Prueba Unitaria		Resultado		
N°	Descripción	Resultado Esperado	Resultado Observado	Verificado
				Si / No
Matriz de Trazabilidad				
N°	N° de Prueba Unitaria	Numero de Prueba en DAM		

## ANEXO 10

### Control de Calidad Interno – CCI

### INFORME DE REVISIÓN

#### Detalles de Revisión

Cliente:	
Proyecto:	
Revisión aprobada:	<i>Si / No</i>
Fecha de revisión:	
Tipo de revisión:	<i>CCI</i>
Descripción:	

<i>Equipo de Revisión</i>		<i>Lista de elementos revisados</i>			
Nombre	Rol	Nro.	Item ID	Versión Nro.	Tamaño del elemento (Nro. de páginas, etc.)

## Informe de revisión - CCI

### Resultados de la revisión

**Principales preocupaciones:**

**Estado de la revisión anterior:**

**Discrepancias/Recomendaciones:**

Para cada discrepancia/recomendación:

<Texto que describe discrepancia / recomendación con referencia al documento relacionado con la calidad correspondiente, en su caso>

Gravedad, tipo, fuente y la causa del defecto son opcionales para los elementos de trabajo de proceso, tales como planes de proyecto

Fuente y Causa No se requiere de sugerencias y Severidad 4 defectos.

### Verificación de Iteración 1

(Por favor, repita estas columnas de múltiples iteraciones de la verificación y cierre)

Discrepancia/ Recomendación	Severidad	Tipo	Fuente	Causa	Acción tomada	Realizada por:	Observaciones:	Acción observada por:	Fecha
	<i>Alta/ Media / Baja</i>	<i>requisitos ambiguos / incompletas</i>							

### Estadísticas de revisión

Horas-persona efectivas utilizadas para su revisión

Numero de discrepancias

Horas-persona efectivas utilizadas para la resolución de defectos

Fecha

Nombre del Evaluador

## ANEXO 11

### Control de Calidad Externo – CCE

### INFORME DE REVISIÓN

#### Detalles de Revisión

Cliente:	
Proyecto:	
Revisión aprobada:	<i>Si / No</i>
Fecha de revisión:	
Tipo de revisión:	<i>CCE</i>
Descripción:	

<i>Equipo de Revisión</i>		<i>Lista de elementos revisados</i>			
Nombre	Rol	Nro.	Item ID	Versión Nro.	Tamaño del elemento (Nro. de páginas, etc.)

## Informe de revisión - CCE

### Resultados de la revisión

**Principales preocupaciones:**

**Estado de la revisión anterior:**

**Discrepancias/Recomendaciones:**

Para cada discrepancia/recomendación:

<Texto que describe discrepancia / recomendación con referencia al documento relacionado con la calidad correspondiente, en su caso>  
Gravedad, tipo, fuente y la causa del defecto son opcionales para los elementos de trabajo de proceso, tales como planes de proyecto  
Fuente y Causa No se requiere de sugerencias y Severidad 4 defectos.

Verificación de Iteración 1

(Por favor, repita estas columnas de múltiples iteraciones de la verificación y cierre)

Discrepancia/ Recomendación	Severidad	Tipo	Fuente	Causa	Acción tomada	Realizada por:	Observaciones:	Acción observada por:	Fecha
	<i>Alta/ Media / Baja</i>	<i>requisitos ambiguos / incompletas</i>							

### Estadísticas de revisión

Horas-persona efectivas utilizadas para su revisión

Numero de discrepancias

Horas-persona efectivas utilizadas para la resolución de defectos

Fecha

Nombre del Evaluador

## ANEXO 12

## Manual de Instalación de Migración – MIM

<b>Organismo</b>	<Nombre Consejería u Organismo Autónomo>		
<b>Proyecto</b>	<Nombre Proyecto>		
<b>Entregable</b>	Manual de Instalación		
<b>Autor</b>	<Nombre de la Empresa>		
<b>Aprobado por</b>		<b>Fecha Aprobación</b>	DD/MM/A AAA
		<b>Nº Total de Páginas</b>	

## REGISTRO DE CAMBIOS

<b>Versión</b>	<b>Causa del Cambio</b>	<b>Responsable del Cambio</b>	<b>Fecha del Cambio</b>
0.1	Versión inicial	<Nombre Apellido1 Apellido2>	DD/MM/AA AA

## CONTROL DE DISTRIBUCIÓN

<b>Nombre y Apellidos</b>
<Nombre Apellido1 Apellido2>

## 1. INTRODUCCIÓN

## 1.1 Objeto

El presente anexo tiene como objetivo servir de guía en la instalación del sistema de migración. Para ello, en primer lugar, deberá de especificar los requerimientos hardware y sistema necesarios para el correcto funcionamiento del sistema, para luego describir cada uno de los pasos necesarios para la configuración, ejecución e instalación del sistema. Además, se deberán incluir las pruebas que se deberán realizar para asegurar que la instalación se ha realizado correctamente, así como el procedimiento de marcha atrás a aplicar en caso de que no haya resultado exitosa la instalación del sistema.

## 1.2 Alcance

Unidades organizativas y responsabilidades a las que va dirigida el documento y que participan en su generación, validación, certificación y registro según la metodología propuesta.

## 2 DESCRIPCIÓN DEL SISTEMA

### 2.1 Antecedentes y descripción funcional del sistema

En esta sección se describirá la situación previa de la organización antes de la construcción del sistema, así como los motivos que llevaron a la construcción del sistema de información. Además, se describirá a alto nivel la funcionalidad que el nuevo sistema aporta o sustituye dentro de la organización.

Se recomienda la utilización de diagramas en estilo libre para ilustrar la funcionalidad del nuevo sistema.

### 2.2 Componentes fundamentales

En esta sección se recogerán los principales módulos del sistema, incluyendo una pequeña descripción de cada uno de ellos.

Módulo	Descripción

### 2.3 Relación con otros sistemas

En esta sección se deberán describir los otros sistemas con los que se relaciona el sistema y describir dichas relaciones. Estos sistemas pueden ser utilizados para obtener información (registros, sistemas de información geográfica...), para realizar acciones necesarias para nuestro sistema (programas de cálculo, sistemas de cifrado...), para realizar tareas de soporte (gestión de agendas, generación de informes...), etc.

Sistema	Relación

## 3 RECURSOS HARDWARE

### 3.2 Ambientes

Es recomendable usar un servidor para cada ambiente (certificación y producción).

### 3.2 Servidores

Para cada uno de los servidores necesarios para la implantación del sistema (servidor de aplicaciones, servidor de base de datos, servidor de archivos, etc.), se detallarán sus requisitos hardware, diferenciando entre requisitos mínimos y recomendados.

Incluir una tabla para cada Servidor.

En caso de no tratarse de servidores virtuales, deberá justificarse en este apartado. Previamente de deberá haber solicitado la creación de la máquina virtual mediante la plantilla correspondiente.

<b>Servidor 1</b>		
<b>Dato</b>	<b>Valor mínimo</b>	<b>Valor recomendado</b>
Procesador		
Memoria RAM		
Tamaño Almacenamiento		
Otros		

### 3.2 Estaciones cliente

Se detallarán los requisitos hardware de las estaciones cliente necesarios para la ejecución del sistema, diferenciando entre requisitos mínimos y recomendados.

<b>Dato</b>	<b>Valor mínimo</b>	<b>Valor recomendado</b>
Procesador		
Memoria RAM		
Tamaño Almacenamiento		
Otros		

### 3.3 Conectividad

Se detallarán todos los requisitos mínimos y recomendados de recursos hardware de conectividad entre servidores y estaciones cliente necesarios para la implantación. Para facilitar la comprensión, se recomienda incluir el diagrama de despliegue correspondiente.

<b>Dato</b>	<b>Valor mínimo</b>	<b>Valor recomendado</b>
Tarjeta de Red		
Tipo de Red		
Otros		

### 3.4 Restricciones

En esta sección deben mostrarse todos aquellos aspectos, circunstancias, etc. que especifiquen las limitaciones del elemento en cuanto a características, funcionamiento, acceso a otros productos, etc.

<b>Restricción</b>	<b>Detalle</b>

## 4 INSTALACIÓN Y CONFIGURACIÓN DEL SISTEMA

Sabemos por sistema base, que es aquel del que depende directamente el sistema para su correcta ejecución, instalación y funcionamiento. Dentro de lo que se identifica típicamente como sistema base tenemos el

entorno de ejecución, el servidor de aplicaciones, el servidor de base de datos. Adicionalmente pueden existir otros en función de las características de nuestro sistema.

En esta sección se describirá el procedimiento de instalación del sistema base. Para cada uno de los componentes sistema utilizado se indicará una pequeña descripción del sistema, la localización en la que se puede obtener, la descripción detallada de todos los pasos que componen su instalación, el procedimiento de configuración y los parámetros a configurar. Toda esta información se cumplimentará en la tabla adjunta (se incluirá una tabla para cada componente sistema).

En caso de que sea necesario realizar alguna configuración adicional en otro sistema (Sistema Operativo, sistema de almacenamiento...), se realizará igualmente en este apartado.

SISTEMA		
Descripción		
Localización		
Procedimiento de instalación		
Paso 1		
...		
Procedimiento de configuración		
Paso 1		
...		
Parámetros a configurar		
Parámetro 1	[Ubicación (BBDD, fichero xml, properties...)]	[Valor]
...	...	...

## 5 CONFIGURACIÓN DEL SISTEMA

### 5.1 Configuración del sistema

En esta sección se detallará la configuración que se llevará a cabo durante la implantación del sistema. Diferenciaremos entre la configuración que es necesario realizar inicialmente, durante la ejecución, durante la instalación y despliegue y al final del proceso. Las configuraciones correspondientes a las fases de ejecución e instalación y despliegue serán referenciadas desde los apartados correspondientes.

Para cada configuración a realizar se indicará el elemento a configurar, el efecto de la configuración, la ubicación en la que realiza la configuración y una descripción detallada de las acciones a llevar a cabo y los valores a establecer. Esta información se estructurará en forma de tabla.

Como ejemplo, se muestra la configuración de los parámetros de conexión a base de datos:

Configuración: Datasource	
<b>Efecto</b>	Se establecen los parámetros de conexión a base de datos.
<b>Fase</b>	Configuración final después del despliegue.
<b>Ubicación</b>	Fichero [directorio de despliegue]/ C:\inetpub\wwwroot
Paso	Descripción
1°	Asegurarse de que la aplicación está correctamente desplegada y de que el servidor de aplicaciones está parado.
2°	Editar el archivo de configuración indicado y establecer los siguientes valores: Server=myServerAddress;Port=1234;Database=myDataBase;Uid=myUsername;Pwd=myPassword;
3°	Guardar el archivo.

Configuración: <Elemento a configurar>	
<b>Efecto</b>	
<b>Fase</b>	
<b>Ubicación</b>	
Paso	Descripción
1°	

## 6 EJECUCIÓN DEL SISTEMA

En esta sección se describirán todos los requisitos existentes y las tareas a realizar para la correcta ejecución del sistema. Como requisitos generales, deben cumplirse todas las indicaciones referentes a construcción y ejecución del sistema que estén recogidas en la normativa técnica.

La información mínima necesaria para la ejecución incluye la ubicación de las fuentes, la configuración del proyecto (profiles, plugins...), los repositorios a partir de los cuales se obtendrán los recursos necesarios, las dependencias del proyecto y el producto final generado.

Finalmente se describirán detalladamente todas las acciones a realizar en el proceso de ejecución, desde la obtención de los ficheros fuente hasta la obtención del producto final, pasando por la ejecución de comandos de configuración y ejecución.

En caso de que sea necesario realizar alguna configuración, se hará referencia a la descripción realizada en el apartado 6 de este documento.

Requisitos de ejecución	
Requisito	Descripción
Ubicación Fuentes	
Configuración 1	
...	
Repositorio 1	
...	
Dependencia 1	
...	
Producto final	

Procedimiento de ejecución	
Paso	Descripción
1	

## 7 INSTALACIÓN DEL SISTEMA

En esta sección se incluirá una descripción detallada, completa y en orden secuencial de todas las tareas que hay que realizar. Para cada acción, se deberá indicar claramente tipo de acción (configuración copia de ficheros, despliegue...), los componentes implicados (ficheros, scripts..) y los permisos necesarios para llevar a cabo la acción (permisos de administración en el servidor de aplicaciones, propietario del esquema de base de datos...).

En caso de que se realice una instalación desde 0 habrá que incluir una referencia al apartado 5 de este documento. Si en cambio la instalación se realiza a partir del sistema base, un requisito previo será la instalación de dicho sistema base.

Por otro lado, en caso de que sea necesario realizar alguna configuración, se hará referencia a la descripción realizada en el apartado 6 de este documento.

Procedimiento de instalación	
Paso 1	
Tipo	
Componente	
Permisos	
Descripción	

Paso 2	
<b>Tipo</b>	
<b>Componente</b>	
<b>Permisos</b>	
<b>Descripción</b>	

## 8 VERIFICACIÓN DEL PROCESO DE INSTALACIÓN

En esta sección se recogerán las comprobaciones mínimas que deberán realizarse una vez realizado el despliegue para asegurar la correcta configuración e instalación del sistema (acceso al sistema, funcionalidad básica, comunicación con los diferentes sistemas externos, etc.).

## 9. REVERSIÓN DEL SISTEMA

En esta sección se describirá la secuencia de tareas que hay que realizar en el caso de querer desinstalar y dar marcha atrás a la instalación y configuración del sistema. Todas las tareas deben describirse detalladamente indicando para cada una de ellas, el tipo de tarea (modificación de configuración, ejecución de scripts...), el componente implicado (aplicación, script...) y los permisos necesarios para llevar a cabo la acción (permisos de administración en el servidor de aplicaciones, propietario del esquema de base de datos...).

Procedimiento de marcha atrás	
Paso 1	
<b>Tipo</b>	
<b>Componente</b>	
<b>Permisos</b>	
<b>Descripción</b>	
Paso 2	
<b>Tipo</b>	
<b>Componente</b>	
<b>Permisos</b>	
<b>Descripción</b>	

## 10. ERRORES U OBSERVACIONES

En caso existan errores al ejecutar el MIM, se procederá a registrar los errores para luego informarle al desarrollador.

Error u Observaciones	Descripción

**11 GLOSARIO**

Término	Descripción



## ANEXO 13

### Plan de implementación – PDI

Responsable	Actividad	% de Avance	Objetivo	Fecha	Hora Inicio	Hora Fin	Observaciones	Predecesor

**Total de cumplimiento:** %

**SISTEMA DE MIGRACIÓN**

%

Clausula Norma	REQUERIMIENTOS GENERALES

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

Clausula Norma	REQUISITOS DE LA DOCUMENTACION

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

**RESPONSABILIDAD DE LA DIRECCION**

%

Clausula Norma	RESPONSABILIDAD DE LA DIRECCION

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

**GESTION DE LOS RECURSOS**

%

Clausula Norma	GESTION DE LOS RECURSOS

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

**REALIZACION DEL SISTEMA**

%

Clausula Norma	REALIZACION DEL PRODUCTO

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

**MEDICIÓN, ANÁLISIS Y MEJORA**

%

Clausula Norma	MEDICION ANALISIS Y MEJORA

%

FECHAS PROPUESTAS		ESTADO
INICIO	FIN	

## ANEXO 14

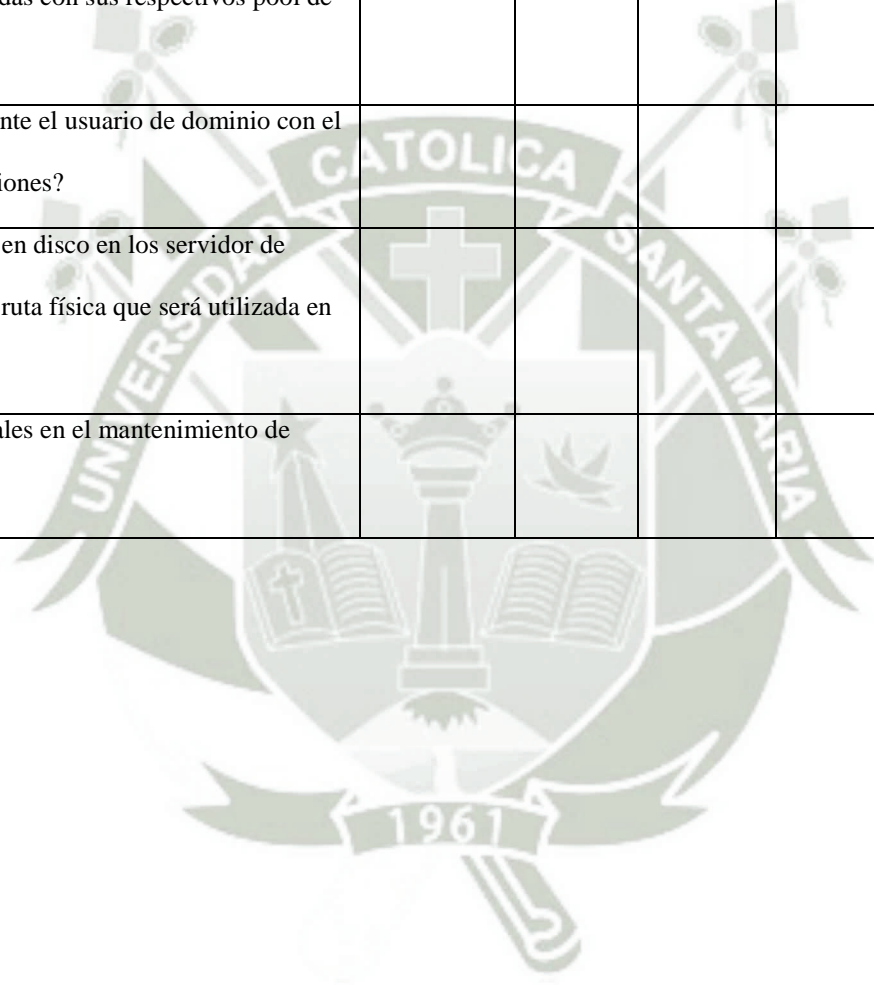
### CheckList de Manual de Instalación de Migración

Checklist de MIM					
<b>Código de proyecto:</b>					
<b>Desarrollador:</b>	<<Nombre>>				
<b>Revisor:</b>	<<Nombre>>				
<b>Tipo de revisión</b>	CCI / CCE				
<b>Fecha de revisión</b>	<<fecha>>				
<b>Items a ser revisados</b>	<b>Madurez(1 a 5)</b>	<b>Auto revisión</b>	<b>CCI</b>	<b>CCE</b>	<b>Comentarios (Opcional)</b>
	<b>1-Baja 5-Alta</b>	<b>Propia</b>	<b>Evaluado</b>	<b>Evaluado</b>	
<b>Validaciones previas al congelamiento</b>					
Se verificó que el Analista envió el correo de fin de análisis al líder de proyecto y experto legado?					
Todos los elementos que se utilizarán en la instalación, se encuentran debidamente congelados					

<b>Validaciones Iniciales MIM</b>					
Se realizó las pruebas de instalación en ambientes de desarrollo, siguiendo cada uno de los pasos del MIM?					
Se realizó las pruebas de reversión en ambientes de desarrollo, siguiendo cada uno de los pasos del MIM?					
¿Si existen entregables que se encuentran en una ruta diferente, se indica claramente en el MIM?					
En el MIM, si se indica la utilización de rutas de carpetas en certificación y producción, se validó la existencia de las mismas?					
En el MIM, si se crean carpetas se está validando que los permisos a otorgar para los usuarios correspondientes sean solo Read/Write.					
En el MIM, si se comparten carpetas se está validando que los permisos a otorgar para los usuarios correspondientes sean solo Read/Change.					
En el MIM si se están editando archivos .config u otros archivos de configuración, se ha validado que las rutas de origen de los mismos sea correcta y que el contenido a impactar realmente exista?					
En el MIM, verificó que no se realice ninguna instalación en la					

unidad de disco C:\, debido a que es reservada para el sistema operativo?					
En el MIM Verificó que no se haga referencia a validaciones, verificaciones, revisiones de hardware y software en ninguna parte del MIM					
El proceso de reversión se explica claramente en el MIM, en caso de sufrir un problema durante el proceso de instalación?					
En el MIM, se evita solicitar inscripción, permisos sobre aplicaciones, carpetas compartidas a los usuarios finales?					
Los permisos en NTFS y Shares se detallan de manera clara (rutas físicas y compartidas)?					
Para shares nuevos, se indica la ruta física, usuarios de grupos de dominio, nivel de permisos y descripción del share?					
Para shares existentes, se indica la ruta física, usuarios de grupos de dominio y el nivel de permisos?, considera también los permisos NTFS?					
Las cuentas locales creadas en el servidor, cumplen con el estándar de nombres?					

Respecto a las políticas locales de seguridad, indica sólo las que utilizará en la instalación?					
Las aplicaciones web están configuradas con sus respectivos pool de aplicaciones y web site?					
En aplicaciones se indica explícitamente el usuario de dominio con el que se ejecutará en el pool de aplicaciones?					
Se verificaron las unidades y espacio en disco en los servidor de producción y certificación, para cada ruta física que será utilizada en la instalación?					
Se validó que no existen usuarios finales en el mantenimiento de permisos y accesos del MIM?					



## ANEXO 15

### CheckList de Programación

Checklist de programación			
<b>Desarrollador:</b>	<< Nombre >>		
<b>Evaluador:</b>	<< Nombre >>		
<b>Tipo opinión</b>	CCI / CCE		
<b>Fecha de revisión</b>	<< fecha >>		
Elementos a comprobar	Coefficiente de ponderación (1-5)	¿CUMPLE?	Comentarios (Opcional)
	1-Bajo 5-Alto	Crítico	
<b>Los estándares de nomenclatura (*)</b>			
El nombre del archivo se deriva del nombre de la clase.			
¿Todas las variables de nomenclatura están de acuerdo con las Normas .NET?			
¿Todos los nombres de las variables indican su uso / función?			
<b>Estándares de Codificación (*)</b>			
Están adecuados los comentarios escritos para las funciones / Sub Rutinas?			
Están todas las constantes declaradas en la parte superior de la clase / módulo?			
¿Está el código indentado correctamente?			
¿Hay algunas líneas de código comentadas en los archivos no deseados?			
¿Hay algún código inservible en los archivos?			
¿El código en los archivos contienen las variables no utilizadas?			
¿Cada control oculto tiene un comentario explicando su propósito?			
Utiliza el estilo de codificación definido en la norma			

<b>Normas funcionales</b>			
¿Están todos los estilos en css?			
¿Están los mensajes de la aplicación en un archivo de recursos?			
Existen llamadas a base de datos directas, funciones de utilidad o SQL dinámico?			
Existe una definición entidad o función de utilidad que esté presente en la clase de acceso a datos.			
No hay llamadas de base de datos o funciones de utilidad que estén presentes en los archivos .aspx.vb.			
Hay funciones comunes de funcionalidad / subrutinas utilizadas, existentes?			
Se inicializan todas las variables antes de ser utilizadas?			
Antes de acceder a las variables de sesión, ¿se está comprobando si existen nulos?			
Las rutas relativas son utilizadas en lugar de las rutas absolutas, siempre que sea aplicable?			
¿Están utilizando propiedades en lugar de variables públicas? Aumenta la encapsulación y da la oportunidad de validar los cambios de valor.			
<b>Actuación</b>			
Se importan sólo los espacios de nombres necesarios?			
Usa espacios de nombres en su caso para evitar conflictos de nombres			
Todas las variables han sido declaradas en el ámbito más pequeño posible?			
¿El uso de nombres propios minimiza el código? Las constantes con nombre se deben utilizar para este propósito.			
La ejecución de objetos se realiza justo antes de su punto de uso, para mejorar el rendimiento?			
Se reduce al mínimo el uso de variables globales?			

Para cada función, son valores devueltos explícitamente a través de instrucción Return, para todos los casos posibles en las que la función podría abandonarse?			
AutoPostBack se establece en falso para los controles como el cuadro de texto, donde no se requiere?			
¿El uso de los controles HTML se usa más en controles web de contenido estático?			
Los controles en los formularios web no realizan operaciones del lado del servidor.			
¿Es la pantalla de cambio de dirección se realiza desde el lado del cliente siempre que sea posible? (Cancelar eventos no deben tener un controlador del lado del servidor)			
Asegúrese de que cualquier Response.Write no debería estar en archivos ASPX.			
Response.Redirect se coloca siempre fuera del bloque Try ... Catch? De lo contrario se generará un hilo de error al ser abortado y registrado.			
La longitud de la matriz siempre se comprueba antes de acceder a un elemento de este conjunto?			
Antes de la conversión de una variable de un tipo de datos Fecha, se usa la función IsDate utiliza para comprobar la validez de la cadena de entrada?			
StringBuilders se utilizan para mejorar el rendimiento?			
StringBuilder establece en un valor óptimo para reducir el coste de las nuevas asignaciones?			
Utilice + cuando el número a añadir al final es conocido			
Están <i>utilizando</i> instrucción en C # y try / finally bloques en Visual Basic.NET para desechar?			
Se utiliza repetidor cada vez que no es necesario clasificar?			
¿Están utilizando correctos tipos de variables? Elección de tipo de datos correcto para asegurar el tamaño de una variable que no es excesivamente grande.			

Las enumeraciones se utilizan cada vez que una variable puede tener un valor de una lista limitada de valores?			
Las variables y rutinas tienen una sola finalidad. Además, de sus funciones no relacionadas.			
Están creando una instancia del objeto y luego asignar la variable con algún otro objeto?			
Están utilizando la menor cantidad posible de llamadas a la base de datos?			
Se minimiza el número de iteraciones en código?			
<b>Administración del ESTADO</b>			
SessionState se deshabilita para una página si no requiere de Sesiones?			
Es mínimo el uso de variables de sesión?			
¿Está el 'ViewState' desactivado para los controles web que no requeridos?			
Si el valor que se utilizará es la de la página, la vista de estado está siendo utilizada en lugar de la sesión?			
<b>Manejo de excepciones</b>			
Se escribe código adecuado en el registro de error en la aplicación?			
La página de error personalizada (que se muestra en caso de error de configuración que no puede ser manejada por código) está incluido?			
Cualquier error en la aplicación muestra error de ejecución?			
¿Hay cualquier error de control en la lectura y escritura de archivos o registro del sistema?			
<b>Seguridad</b>			
¿Se utiliza seguridad integrada? Un usuario para acceder a los recursos.			
Se utiliza la seguridad de Active Directory?			

La gestión de usuarios se lleva a cabo por el área de la seguridad?			
Como mínimo, el acceso está comprobado por página / formulario?			
<b>ADO.NET</b>			
Antes de acceder a las columnas en una fila, se comprueba si el valor es nulo?			
Se utiliza DataReader para el enlace de datos rápido y eficiente?			
Cuando se utiliza DataReader, el CommandBehavior.CloseConnection pasa como parámetro al método ExecuteReader? Cuando se ejecuta el comando, el objeto de conexión asociado se cierra cuando el objeto DataReader asociado está cerrado.			
¿Los descriptores de acceso escrito como GetString, GetInt32 utilizan al acceder a datos de la columna? Esto le ahorra el procesamiento necesario para convertir el objeto de retorno de GetValue como un tipo particular de DataReader.			
DataReader se utiliza en lugar del conjunto de datos para devolver una sola fila			
ExecuteNonQuery se utiliza para los comandos que no devuelven datos?			
ExecuteScalar se utiliza para devolver valores individuales?			
DataTable.Rows.Find se utiliza en lugar de DataTable.Select. ¿Si la consulta está en contra de las columnas que componen el PrimaryKey de la DataTable? Asigna clave principal a la columna en la tabla de datos antes de realizar la implementación anterior.			
Mejorar el rendimiento de varias consultas de los datos mediante un DataView, si la consulta implica columnas de clave no primarios como criterios de búsqueda.			
Están especificados los parámetros de procedimiento almacenado creados de forma explícita y tipos de parámetros?			

¿Mientras pasa los datos XML a un procedimiento almacenado, se está haciendo un check para reemplazar caracteres especiales?			
<b>Configuración</b>			
El atributo de depuración se establece en false en el archivo Web.config?			
Son los atributos executionTimeout y maxRequestLength ajustes de valor mínimo y suficiente?			
<b>java script</b>			
Usa java script para reducir la recarga de la página siempre que sea posible?			
Todo el código JS común se trasladó a un solo archivo?			
Todos los identificadores de control son accesibles de forma dinámica mediante el control de clientid de asp.net en lugar de codificar el identificador siempre que sea posible?			
Se tiene cuidado para hacer el trabajo de código JS en varios navegadores?			
TOTAL			

## ANEXO 16

### Control de Defectos - CDD

<b>Código de Proyecto</b>			
<b>Proyecto</b>	<Nombre Proyecto>		
<b>Entregable</b>	Control de defectos		
<b>Autor</b>	<Nombre de la Empresa>		
<b>Versión/Edición</b>		<b>Fecha Versión</b>	DD/MM/AAAA
<b>Aprobado por</b>		<b>Fecha Aprobación</b>	DD/MM/AAAA
		<b>Nº Total de Páginas</b>	

#### REGISTRO DE CAMBIOS

Versión doc	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0.1	Versión inicial	<Nombre Apellido1 Apellido2>	DD/MM/AAAA

#### CONTROL DE DISTRIBUCIÓN

<b>Nombre y Apellidos</b>
<Nombre Apellido1 Apellido2>

#### 1.- Reporte de defectos

<b>Nombre de quien ejecuta la prueba:</b>	
<b>Fecha del reporte (dd/mm/aaaa):</b>	
<b>Actividad:</b>	617
<b>Error:</b>	

CAPTURA DE PANTALLA DEL DEFECTO



## ANEXO 17

### Documento de Pruebas Legado – DPL

DPL - Documento de Pruebas Legado - Casos de Prueba				
Nombre de Proyecto				
Código de Proyecto				
Experto Legado				
ID de Prueba		Versión No.		Referencia
Elemento de prueba				
Ambiente	<<Certificación>> o <<Producción>>			
Preparado por:			Fecha de Preparación:	
Verificado por:			Fecha de verificación:	
Datos iniciales para realizar las pruebas				
Para:				
Archivos a usar:				
Pre-requisitos:				
Caso de Prueba		Resultado		
N°	Descripción	Resultado Esperado	Resultado Observado	Verificado
				<i>Si / No</i>

## ANEXO 18

### Aceptación de Plan Piloto - APP

APP - Aceptación de Plan Piloto					
Nombre de Proyecto:					
Código de Proyecto:					
Cliente:					
Ratificación:	<<Parcial>> - <<Plan Piloto>>				
Entrega de:					
Fecha de entrega:					
Lista de módulos entregados:					
<p>El Cliente certifica que la totalidad de los módulos reseñados en la presente acta de recepción han sido entregados/terminados y que, habiendo sido sometidos a las pruebas de validación y aceptación indicadas, están de acuerdo con las especificaciones formales y demás requisitos contractualmente convenidos y establecidos entre las partes para dar inicio al plan piloto, con las siguientes OBSERVACIONES :</p>					
<table border="1"> <thead> <tr> <th>Por el Contratista</th> <th>Por el Cliente:</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>		Por el Contratista	Por el Cliente:		
Por el Contratista	Por el Cliente:				
Líder de Proyecto:	Gerente General:				
Fecha:	Fecha:				
<p align="center"><b>Formato de aceptación parcial de los trabajos</b> Con la firma del acta de aceptación, el cliente certifica que ha revisado los trabajos presentados y que está conforme con ellos.</p>					

APP - Aceptación de Plan Piloto	
Nombre de Proyecto:	
Código de Proyecto:	
Cliente:	
Ratificación:	<<Parcial>> - <<Plan Piloto>>

Entrega de:		
Fecha de entrega:		
Lista de módulos entregados:		
<p>El Cliente certifica que la totalidad de los módulos reseñados en la presente acta de recepción han sido entregados/terminados y que, habiendo sido sometidos a las pruebas de validación y aceptación indicadas, están de acuerdo con las especificaciones formales y demás requisitos contractualmente convenidos y establecidos entre las partes para dar inicio al plan piloto, con las siguientes OBSERVACIONES :</p>		
Por el Contratista		Por el Cliente:
Líder de Proyecto:	Gerente General:	
Fecha:	Fecha:	
<p><b>Formato de aceptación parcial de los trabajos</b> Con la firma del acta de aceptación, el cliente certifica que ha revisado los trabajos presentados y que está conforme con ellos.</p>		

## ANEXO 19

### Fin de Proyecto – FDP

FDP - Fin de Proyecto			
Nombre de Proyecto:			
Código de Proyecto:			
Cliente:			
Ratificación:	<<Total>>		
Fecha Inicio:		Fecha Fin:	
Trabajo realizado - Alteraciones al alcance previsto			
Dificultades Encontradas			
Acciones Futuras			
Resumen de Estado:			
criterio	Si	No	Comentario
Modificaciones al alcance			
Retrasos			
Incremento del riesgo			
Sobrecoste			
Insatisfacción del Cliente			
Ampliaciones al contrato			
Carencia de recursos			
Conflictos interpersonales			
Falta de formación y/o experiencia			
Por el Contratista		Por el Cliente:	
Lider de Proyecto:		Gerente General:	
Fecha:		Fecha:	
<b>Formato de aceptación final del proyecto</b>			
Con la firma del acta de aceptación, el cliente certifica que ha culminado el proyecto en su totalidad .			

## ANEXO 20

### Buenas Prácticas – BUP

APP - Aceptación de Plan Piloto	
N° de BP	
Nombre de Proyecto:	
Código de Proyecto:	
Cliente:	
Fecha :	
Módulo:	
Descripción del problema:	
Riesgo:	<< del 1 al 5, siento 1 menos riesgoso y 5 mayor riesgo>>
Días de retraso:	
Descripción de la solución:	

## ANEXO 21

Califique del 0 al 4, considerando que <b>0=Muy Malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy Bueno</b>																			
<b>PERSONAS ENCUESTADAS</b>																			
<b>1. ¿Aprueba usted la metodología?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	
Que tan bueno fue el proceso de Planificación	3	3	4	3	3	4	4	3	4	4	4	2	4	4	4	4	4	3	
Que tan bueno fue el proceso de Análisis	4	4	3	4	4	3	4	4	4	4	3	4	3	4	3	4	3	4	
Que tan bueno fue el proceso de Construcción	2	4	4	4	3	4	2	4	2	3	4	3	4	4	4	3	4	4	
Que tan bueno fue el proceso de Certificación	4	4	4	4	4	3	4	4	4	4	4	4	4	3	3	4	4	3	
Que tan bueno fue el proceso de Implementación	4	4	4	3	4	3	4	4	4	4	3	4	4	4	4	4	3	4	
<b>Total</b>	17	19	19	18	18	17	18	19	18	19	18	17	19	19	18	19	18	18	
<b>Total Porcentaje</b>	85	95	95	90	90	85	90	95	90	95	90	85	95	95	90	95	90	90	
<b>Porcentaje total de aprobación</b>	<b>91,11%</b>																		

**Pregunta 1 de la Evaluación Metodológica**

## ANEXO 22

Califique del 0 al 4, considerando que <b>0=Total desacuerdo, 1=No de acuerdo, 2=Irrelevante, 3=De Acuerdo, 4=Totalmente de acuerdo</b>																		
<b>PERSONAS ENCUESTADAS</b>																		
<b>2.- ¿Considera usted que ésta metodología es compleja?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18
Fue complejo realizar el proceso de Planificación	1	0	2	0	1	1	0	1	0	1	0	2	1	0	0	2	0	2
Fue complejo realizar el proceso de Análisis	0	1	0	0	0	0	2	0	0	0	1	0	1	1	2	0	0	0
Fue complejo realizar el proceso de Construcción	0	2	1	1	0	0	0	2	1	0	0	0	1	0	0	1	0	0
Fue complejo realizar el proceso de Certificación	1	0	2	1	0	1	0	1	2	2	0	1	1	1	0	0	1	1
Fue complejo realizar el proceso de Implementación	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	2	0	0
<b>Total</b>	2	3	5	2	1	2	2	4	3	3	1	3	5	2	2	5	1	3
<b>Total Porcentaje</b>	10	15	25	10	5	10	10	20	15	15	5	15	25	10	10	25	5	15
<b>Porcentaje total de aprobación</b>	<b>13,61%</b>																	

### Pregunta 2 de la Evaluación Metodológica

## ANEXO 23

Califique del 0 al 4, considerando que <b>0=Muy Malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy Bueno</b>																			
<b>PERSONAS ENCUESTADAS</b>																			
<b>3. ¿Considera usted que el control de calidad propuesto (CCI, CCE) fue un buen influyente en la realización de esta metodología?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	
Cómo considera el producto terminado en la etapa de Planificación después de realizar el CCI y CCE	4	3	4	4	4	4	4	3	4	4	4	3	4	4	4	4	3	4	
Cómo considera el producto terminado en la etapa de Análisis después de realizar el CCI y CCE	4	3	4	4	4	4	3	3	3	4	4	4	3	3	4	3	4	4	
Cómo considera el producto terminado en la etapa de Construcción después de realizar el CCI y CCE	3	4	4	4	3	4	4	4	4	4	4	3	4	3	4	4	3	4	
Como considera la influencia de la documentación en la entrega del producto final	4	4	4	4	4	4	4	3	4	4	4	4	4	4	3	3	4	3	
Cómo considera la influencia del proceso de calidad en la entrega del producto final	4	3	4	3	4	3	3	4	4	4	4	4	4	4	4	4	4	4	
<b>Total</b>	19	17	20	19	19	19	18	17	19	20	20	18	19	18	19	18	18	19	
Total Porcentaje	95	85	100	95	95	95	90	85	95	100	100	90	95	90	95	90	90	95	
<b>Porcentaje total de aprobación</b>	<b>93,33%</b>																		

### Pregunta 3 de la Evaluación Metodológica

## ANEXO 24

Califique del 0 al 4, considerando que <b>0=Nada Útil, 1=Poco Útil, 2=Irrelevante, 3=Útil, 4=Muy útil</b>	
<b>PERSONAS ENCUESTADAS</b>	
<b>4. ¿Considera usted que el uso de las buenas practicas fueron de gran ayuda para el modelamiento de la metodología?</b>	P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 P13 P14 P15 P16 P17 P18
Las buenas prácticas sirvieron de ayuda en la etapa de Planificación	3 3 4 4 4 4 4 4 4 3 3 4 4 4 4 4 4 4
Las buenas prácticas sirvieron de ayuda en la etapa de Análisis	3 4 4 4 4 3 4 4 3 4 4 4 4 4 4 3 4 3
Las buenas prácticas sirvieron de ayuda en la etapa de Desarrollo	3 3 4 4 4 4 3 4 4 4 4 3 4 3 4 4 4 4
Considera útil contar con un repositorio de buenas prácticas y hacer uso de ellas.	4 4 4 4 4 4 4 3 3 4 3 3 4 4 3 4 4 4
<b>Total</b>	<b>13 14 16 16 16 15 15 15 14 15 14 14 16 15 15 15 16 15</b>
<b>Total Porcentaje</b>	<b>81 88 100 100 100 94 94 94 88 94 88 88 100 94 94 94 100 94</b>
<b>Porcentaje total de aprobación</b>	<b>93,40%</b>

### Pregunta 4 de la Evaluación Metodológica

## ANEXO 25

Califique del 0 al 4, considerando que <b>0=Total desacuerdo, 1=No de acuerdo, 2=Irrelevante, 3=De Acuerdo, 4=Totalmente de acuerdo</b>																			
PERSONAS ENCUESTADAS																			
<b>5. ¿Considera usted las buenas prácticas como un valor agregado para la metodología?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	
Por su relevancia en las etapas de la metodología	3	4	3	4	4	3	3	4	4	4	4	3	4	3	4	4	4	4	
Por su ahorro de tiempo en el desarrollo de la metodología	3	3	4	3	3	4	4	4	3	4	4	4	3	4	2	4	4	4	
Por su transferencia de conocimiento al equipo migrado	3	4	3	3	4	3	4	3	4	3	4	3	3	3	4	4	3	3	
<b>Total</b>	<b>9</b>	<b>11</b>	<b>10</b>	<b>10</b>	<b>11</b>	<b>10</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>12</b>	<b>10</b>	<b>10</b>	<b>10</b>	<b>10</b>	<b>12</b>	<b>11</b>	<b>11</b>	
Total Porcentaje	75	92	83	83	92	83	92	92	92	92	100	83	83	83	83	100	92	92	
<b>Porcentaje total de aprobación</b>	<b>88,43%</b>																		

### Pregunta 5 de la Evaluación Metodológica

## ANEXO 26

Califique del 0 al 4, considerando que <b>0=En total desacuerdo, 1=En desacuerdo, 2=Irrelevante, 3=De acuerdo, 4=Totalmente de acuerdo</b>										
PERSONAS ENCUESTADAS										
1. ¿Cree usted que el aplicativo migrado es escalable a diferencia del legado?	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Se presta a futuras implementaciones de módulos web	4	3	3	4	4	4	4	4	4	4
Se presta a futuras implementaciones de plugins	3	3	4	3	3	3	4	4	3	3
Tiene la capacidad de poder interactuar con otros sistemas	3	3	4	4	3	3	3	3	4	4
Se adapta a las actuales tendencias web	4	3	4	4	4	4	4	3	3	4
Total	14	12	15	15	14	14	15	14	14	15
Total Porcentaje	88	75	94	94	88	88	94	88	88	94
<b>Porcentaje total de aprobación</b>	<b>88,75%</b>									

### Pregunta 1 de la Evaluación del Aplicativo Migrado

## ANEXO 27

Califique del 0 al 4, considerando que <b>0=En total desacuerdo, 1=En desacuerdo, 2=Irrelevante, 3=De acuerdo, 4=Totalmente de acuerdo</b>										
PERSONAS ENCUESTADAS										
2. ¿Cree usted que el aplicativo se adecua a sus necesidades y por lo tanto justifica su ejecución en la empresa?	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Considera que el mantenimiento es el correcto	3	2	4	2	4	3	4	3	4	3
Considera usted que los datos que muestran los reportes son los mismos que mostraba el legado	4	3	3	4	4	4	3	4	4	4
Considera que los roles para el ingreso fueron bien distribuidos	4	3	4	4	3	3	4	4	3	4
Considera que el aplicativo Migrado tiene funciones más evidentes al usuario que el Legado	3	4	3	3	4	4	2	3	3	4
Considera que la distribución de elementos del aplicativo Migrado es más ordenado que el Legado	4	3	4	3	4	4	4	4	4	3
Total	18	15	18	16	19	18	17	18	18	18
Total Porcentaje	90	75	90	80	95	90	85	90	90	90
<b>Porcentaje total de aprobación</b>	<b>87,50%</b>									

**Pregunta 2 de la Evaluación del Aplicativo Migrado**

## ANEXO 28

Califique del 0 al 4, considerando que <b>0=En total desacuerdo, 1=Desacuerdo, 2=Irrelevante , 3=De acuerdo, 4=Totalmente de acuerdo</b>										
<b>PERSONAS ENCUESTADAS</b>										
<b>3. ¿Considera usted que el aplicativo funcionó adecuadamente durante su ejecución?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
No se encontraron enlaces rotos	3	3	4	4	3	4	3	4	4	2
El aplicativo migrado conservó la relación de aspecto en los diferentes navegadores	3	4	3	3	4	4	3	4	4	4
El aplicativo migrado conservó la relación de aspecto en los diferentes dispositivos	3	2	3	4	4	3	3	4	4	2
No existió ningún tipo de error al ejecutar el aplicativo migrado	3	4	4	4	4	4	4	3	3	4
La funcionalidad del aplicativo migrado es igual al aplicativo legado	4	4	3	4	4	4	4	4	4	4
<b>Total</b>	<b>16</b>	<b>17</b>	<b>17</b>	<b>19</b>	<b>19</b>	<b>19</b>	<b>17</b>	<b>19</b>	<b>19</b>	<b>16</b>
<b>Total Porcentaje</b>	<b>80</b>	<b>85</b>	<b>85</b>	<b>95</b>	<b>95</b>	<b>95</b>	<b>85</b>	<b>95</b>	<b>95</b>	<b>80</b>
<b>Porcentaje total de aprobación</b>	<b>89,00%</b>									

### Pregunta 3 de la Evaluación del Aplicativo Migrado

## ANEXO 29

Califique del 0 al 4, considerando que <b>0=Muy malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy bueno</b>										
<b>PERSONAS ENCUESTADAS</b>										
<b>4. ¿Considera usted que los menús y sub-menús se encuentran mejor organizados en el aplicativo migrado?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Que tan favorable es la distribución de menús	3	2	4	3	4	3	4	3	4	3
Cómo considera la distribución del menú respecto al Legado	4	3	3	3	4	4	2	4	3	4
Cómo considera la clasificación del menú por roles de usuario	4	4	2	4	3	3	4	3	3	4
Cómo considera el menú es evidente a la vista del usuario	2	4	4	4	4	4	4	4	4	4
<b>Total</b>	<b>13</b>	<b>13</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>14</b>	<b>14</b>	<b>14</b>	<b>14</b>	<b>15</b>
<b>Total Porcentaje</b>	<b>81</b>	<b>81</b>	<b>81</b>	<b>88</b>	<b>94</b>	<b>88</b>	<b>88</b>	<b>88</b>	<b>88</b>	<b>94</b>
<b>Porcentaje total de aprobación</b>	<b>86,88%</b>									

**Pregunta 4 de la Evaluación del Aplicativo Migrado**

## ANEXO 30

Califique del 0 al 4, considerando que 0=En total desacuerdo, 1=Desacuerdo, 2=Irrelevante, 3=De acuerdo, 4=Totalmente de acuerdo										
PERSONAS ENCUESTADAS										
5. ¿Siente que son confiables los datos que se encuentran en el aplicativo migrado?	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
La información guardada es la correcta	4	3	4	4	3	3	4	3	4	3
La información mostrada en los reportes es la correcta	4	3	3	4	4	4	4	4	4	4
La información actualizada en los reportes es la correcta	4	4	3	4	4	4	3	3	3	4
La información mostrada en widgets es la correcta	3	4	4	4	2	4	4	4	4	3
<b>Total</b>	<b>15</b>	<b>14</b>	<b>14</b>	<b>16</b>	<b>13</b>	<b>15</b>	<b>15</b>	<b>14</b>	<b>15</b>	<b>14</b>
Total Porcentaje	94	88	88	100	81	94	94	88	94	88
<b>Porcentaje total de aprobación</b>	<b>90,63%</b>									

### Pregunta 5 de la Evaluación del Aplicativo Migrado

## ANEXO 31

Califique del 0 al 4, considerando que <b>0=Muy difícil, 1=Difícil, 2=Irrelevante, 3=Fácil, 4=Muy fácil</b>										
<b>PERSONAS ENCUESTADAS</b>										
<b>6. ¿Cómo considera usted el manejo del aplicativo?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Cómo considera el Ingreso al aplicativo	4	3	4	2	4	3	3	4	3	4
Cómo considera el ingreso a los menús	2	2	4	3	3	3	4	3	4	4
Cómo considera el ingreso al mantenimiento	4	3	3	4	3	3	4	4	4	3
Cómo considera el ingreso de la actualización	3	4	4	3	4	4	3	3	3	4
Cómo considera el uso de widgets	4	3	3	4	3	4	3	3	4	4
Cómo considera la ejecución de los reportes	4	4	3	3	4	4	4	4	4	2
Total	21	19	21	19	21	21	21	21	22	21
Total Porcentaje	88	79	88	79	88	88	88	88	92	88
<b>Porcentaje total de aprobación</b>	<b>86,25%</b>									

**Pregunta 6 de la Evaluación del Aplicativo Migrado**

## ANEXO 32

Califique del 0 al 4, considerando que <b>0=Muy Malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy Bueno</b>										
<b>PERSONAS ENCUESTADAS</b>										
<b>7. ¿Se encuentra satisfecho con el trato personalizado que le brinda el aplicativo?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Cómo considera la distribución de roles	3	4	3	3	3	3	3	3	3	3
Cómo considera el ingreso al Aplicativo por roles	4	4	3	4	4	4	4	4	3	4
Cómo considera los permisos asignados en cada rol	4	4	3	4	3	3	4	4	4	4
Cómo considera los Widgets mostrados por roles	4	4	4	4	4	4	4	4	4	3
<b>Total</b>	<b>15</b>	<b>16</b>	<b>13</b>	<b>15</b>	<b>14</b>	<b>14</b>	<b>15</b>	<b>14</b>	<b>15</b>	<b>14</b>
Total Porcentaje	94	100	81	94	88	88	94	88	94	88
<b>Porcentaje total de aprobación</b>	<b>90,63%</b>									

**Pregunta 7 de la Evaluación del Aplicativo Migrado**

### ANEXO 33

Califique del 0 al 4, considerando que <b>0=Muy Malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy Bueno</b>										
<b>PERSONAS ENCUESTADAS</b>										
<b>8. ¿Considera usted que la portabilidad del aplicativo sea una ventaja?</b>	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Cómo considera la ejecución del aplicativo en un Smartphone	4	3	4	3	3	4	4	3	4	4
Cómo considera la ejecución del aplicativo en una Tablet	3	4	4	4	4	3	4	4	3	4
Cómo considera la ejecución del aplicativo en una máquina de escritorio	4	4	4	4	4	4	3	4	4	4
<b>Total</b>	<b>11</b>	<b>11</b>	<b>12</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>11</b>	<b>12</b>
<b>Total Porcentaje</b>	<b>92</b>	<b>92</b>	<b>100</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>100</b>
<b>Porcentaje total de aprobación</b>	<b>93,33%</b>									

#### Pregunta 8 de la Evaluación del Aplicativo Migrado

## ANEXO 34

Califique del 0 al 4 que tan bueno fue el Aplicativo Migrado respecto al Legado, considerando que <b>0=Muy Malo, 1=Malo, 2=Regular, 3=Bueno, 4=Muy Bueno</b>	
<b>PERSONAS ENCUESTADAS</b>	
<b>9. ¿Considera usted que la interface del aplicativo migrado, es mejor que la interface del aplicativo legado?</b>	P1 P2 P3 P4 P5 P6 P7 P8 P9 P10
Cómo considera la forma de ingreso	3 4 4 4 4 4 3 4 4 3
Cómo considera los widgets mostrados	4 2 3 3 4 4 4 4 3 4
Cómo considera la distribución de menús y submenús	2 4 2 4 3 3 4 2 4 3
Cómo considera los datos del usuario que ingresa	4 3 4 3 4 4 3 4 4 3
Cómo considera los reportes mostrados	4 3 4 4 2 3 4 4 4 4
Cómo considera la exportación de reportes a distintos formatos	3 4 4 4 4 3 4 4 3 3
<b>Total</b>	<b>20 20 21 22 21 21 22 22 22 20</b>
<b>Total Porcentaje</b>	<b>83 83 88 92 88 88 92 92 92 83</b>
<b>Porcentaje total de aprobación</b>	<b>87,92%</b>

**Pregunta 9 de la Evaluación del Aplicativo Migrado**