UNIVERSIDAD CATÓLICA SANTA MARÍA

FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES

PROGRAMA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Sistema Web Para El Intercambio Seguro De Documentos Electrónicos, Utilizando Firmas Y Certificados Digitales X509, Sobre Un Canal Ssl

PARA OPTAR EL TÍTULO PROFESIONAL DE

INGENIERO DE SISTEMAS

PRESENTADO POR LOS BACHILLERES:

LUIS ALFREDO TACO ARIAS

SERGIO ENRIQUE GAMARRA RAMIREZ

AREQUIPA – PERU

2014

DEDICATORIA

A mis padres quienes me dieron el ejemplo y la oportunidad de culminar la carrera y darme la posibilidad de titularme, a mis hermanos y abuelos por sus consejos y preocupación sobre mi futuro.

Luis Alfredo Taco Arias

A mis padres por el apoyo que me brindaron desde el inicio de este proyecto, motivándome diariamente a alcanzar una meta de vida, y a mis compañeros que con sus consejos impulsaron el avance del mismo.

Sergio Enrique Gamarra Ramirez

AGRADECIMIENTOS

A la universidad Católica Santa María por la formación profesional y humana que nos inculcó.

A nuestro asesor el ingeniero José Sulla Torres por su tiempo y guía en la elaboración de esta tesis.

A la ingeniera Karina Rosas por su desinteresado apoyo y sus consejos y observaciones en la revisión del borrador de tesis.

PRESENTACIÓN

Sra. Director del Programa Profesional de Ingeniería de Sistemas.
Srs. Miembros del Jurado.

De conformidad con las disposiciones del Reglamento de Grados y Títulos del Programa Profesional de Ingeniería de Sistemas, ponemos a vuestra consideración el presente trabajo titulado:
"Sistema web para el intercambio seguro de documentos electrónicos, utilizando firmas y certificados digitales X509, sobre un canal SSL", el mismo que de ser aprobado nos permitirá optar por el Título Profesional de Ingeniería de Sistemas.

Luis Alfredo Taco Arias
Sergio Enrique Gamarra Ramirez

RESUMEN

En la actualidad debido al avance de la tecnología muchos estamos familiarizados con el uso de documentos digitales y manejamos dicha información a diario, pero pocas veces le prestamos la debida atención a la seguridad que se tiene con el uso de dichos documentos, y es debido a que en un medio público y ampliamente usado como es internet se carece del soporte necesario para saber si un documento ha sido realmente creado por la persona que dice ser al autor o que en el transcurso de su envío no fue alterado por alguna otra persona que no sea al autor y en ocasiones puede surgir la necesidad en la cual deseamos que sólo el destinario pueda leer los documentos que enviamos. Es por ello que en el presente trabajo nosotros planteamos un software accesible a través de internet que permita cubrir aspectos de seguridad como son la integridad, autoría, no repudio y confidencialidad de los documentos electrónicos, para cumplir con este objetivo usamos técnicas de criptografía que han sido ampliamente estudiadas entre los cuales hacemos mención a los criptosistemas asimétricos los cuales dieron paso al desarrollo técnicas más avanzadas como son las firmas digitales y protocolos SSL de transmisión segura de datos.

El sistema que proponemos aplicará el uso de firmas digitales y protocolos SSL en una plataforma web de tal forma que los usuarios ingresen al sistema para subir documentos, firmarlos y compartirlos con sus contactos, también se permite a los usuarios que tienen acceso al documento puedan reportar sus observaciones en cuyo caso se informará a todos los involucrados con el documento.

Por último documentamos el desarrollo del sistema y lo construimos sobre una arquitectura en capas basado en componentes de tal manera que se facilite el mantenimiento, cambios y reutilización del sistema.

ABSTRACT

At present due to the advancement of technology many of us are familiar with the use of digital documents and manage this information daily, but rarely adequate attention is paid to safety involved with the use of these documents, and it is because in a public environment widely used as internet, it lacks the support needed to know if a document was actually created by the person claiming to be the author or during transit if this document was not altered by someone other than the author and sometimes the need may arise in which we want documents sent by us could be read only by consignee. That is why in this present work we propose a software accessible via the Internet that will cover security aspects such as integrity, authorship, confidentiality and non-repudiation of electronic documents, to achieve this objective we use cryptographic techniques which have been widely studied among with them we mention the asymmetric cryptosystems which allowed the development to more advanced techniques such as digital signatures and SSL protocols for secure data transmission.

The system we propose will apply the usage of digital signatures and SSL protocols in a web platform so users can access the system to upload documents, also users who have access to document are also allowed to report their observations in which case it will be notified to all users involved with document.

Finally we document the development of the system and build it upon a layered architecture based on components so it facilitates maintenance, change and reuse of the system.

INTRODUCCIÓN

Actualmente usamos a diario documentos digitales para diversas tareas y uno de los medios principalmente usados para trasladar dicha información es internet, pero pocas veces prestamos la debida atención a la seguridad que se debe tener al momento de crear dichos documentos y enviarlos a través de internet.

Debido al uso masivo de internet como medio de transporte surgen nuevas formas de vulnerar la seguridad de los documentos digitales, y es cuando no se puede tener la certeza de que los documentos recibidos no han sido modificados por personas ajenas al emisor y receptor, o que no han sido creados por la persona que dice ser el autor y que finalmente no se pueda asegurar que solamente el destinatario reciba dicha información.

Frente al problema de la seguridad de la información surgen técnicas como la criptografía cuya finalidad es ocultar la información y que el intercambio de mensajes sólo puedan ser leídos por personas a las que van dirigidos; la criptografía es una ciencia ampliamente estudiada y través de esta surgen técnicas más avanzadas como son las firmas digitales, protocolos de transmisión segura SSL, sistemas de encriptación simétrica y asimétrica.

En el presente trabajo se presenta una propuesta para incrementar la seguridad de los documentos digitales por medio de un sistema web en el que usamos firmas digitales para permitir confirmar la identidad de las personas que firman los documentos, proteger la integridad de los documentos y garantizar el no repudio de los firmantes; finalmente aplicamos protocolos de transmisión segura (SSL) para proteger las operaciones que se realizan en un medio público como es internet.

En el primer capítulo plantearemos el problema a resolver, definiremos los objetivos generales y específicos, definiremos las hipótesis y las variables dependientes e independientes, finalmente listaremos los trabajos de tesis relacionados con el tema de estudio.

En el segundo capítulo recabaremos un marco teórico en el cual presentaremos el tema de encriptación asimétrica, algoritmos hash, también hablaremos de firmas digitales, certificados digitales, protocolos SSL, los estándares que se han planteado con respecto a la firma y certificado digital y mencionaremos la normatividad peruana que regula las firmas y certificados digitales.

En el tercer capítulo abarcaremos las fases de análisis y diseño del sistema, cabe destacar que en cuanto a la metodología para desarrollar el software usamos SCRUM debido a que es una metodología ágil de desarrollo y se centra más en dar resultados que una extensa documentación.

En el cuarto capítulo mostramos los resultados de las pruebas realizadas al sistema basados en los indicadores anteriormente planteados en el primer capítulo.

Finalmente mostramos las conclusiones y recomendaciones a las que se llegó con el desarrollo del presente trabajo.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1: PLANTEAMIENTO TEÓRICO

1.1 EL PROBLEMA

1.1.1 Definición del problema

Hoy en día el manejar documentos en formato digital es parte del quehacer diario de las personas, editamos, creamos y enviamos dichos documentos a nuestros contactos por diversos motivos, pero no prestamos la debida atención a su seguridad y correspondiente integridad, lo cual conlleva un riesgo, principalmente cuando se trata de documentos laborales importantes, ya que estos pueden ser interceptados en el envío y después ser modificados sin el consentimiento del remitente.

El mismo escenario se da en empresas que muchas veces no cuentan con sistemas para comunicarse con sus proveedores y se limitan a editar documentos en la computadora o escanear imágenes de sus recibos, solicitudes, órdenes de compra, etc. y finalmente las envían por correo electrónico, realizar este tipo de transacciones conlleva un riesgo, ya que vulnerar su integridad es cada vez más sencillo, solo con una herramienta editora de imágenes se puede alterar el contenido del archivo de imagen y pocos notarían la diferencia.

Lo anterior nos lleva a las siguientes conjeturas, ¿es posible en la actualidad confiar plenamente en el contenido de un documento digital?, ¿es el documento digitalizado la representación real del documento físico?, ¿la persona que me envía el documento es realmente quien dice ser?, son por estas preguntas que se propone un sistema web capaz de incrementar la seguridad en el intercambio de documentos digitales.

1.1.2 Área científica a la que corresponde el problema

- **Área**

  Comunicación de datos.

- **Línea**

  Seguridad.

1.1.3 Tipo y nivel de investigación

- **Tipo:** Aplicada

  El objetivo es usar el conocimiento adquirido de las diferentes técnicas criptográficas para crear un sistema web que incremente la seguridad en el intercambio de documentos digitales.

- **Nivel:** Descriptiva

  Se trata de analizar el incremento de seguridad documental al utilizar un sistema web que posee técnicas criptográficas y canales seguros de transmisión.

1.2 OBJETIVOS GENERAL Y ESPECÍFICOS

1.2.1 Objetivo general

- Desarrollar el Sistema Web de firmas digitales, Certificados X509 y SSL que permita incrementar la seguridad en el intercambio de documentos electrónicos.

1.2.2 Objetivos específicos

- Investigar y Elegir la metodología de desarrollo de software que más se adecue al desarrollo del sistema.
  - Búsqueda de metodologías.
  - Evaluación de metodologías y herramientas a utilizar.
- Investigar y documentar sobre las diferentes técnicas y algoritmos de encriptación.
  - Búsqueda de Bibliografía.
  - Revisión y generación de resumen.
- Analizar y diseñar el sistema web de intercambio documental seguro.
- Desarrollar sistema web propuesto para garantizar la seguridad documental.
  - Desarrollar el módulo de registro de usuarios.
  - Desarrollar el módulo de contactos.
  - Desarrollar el módulo para compartir e intercambiar documentos digitales.
  - Desarrollar los módulos para la firma de documentos.
  - Implementar SSL en el sistema web.
- Elaborar pruebas y testeo general del sistema web.

- Redactar el informe final del sistema web desarrollado.
  - Elaboración de Manual de Usuario.
- Probar el sistema desarrollado en un caso real, al utilizarlo en una empresa local.

1.3 FORMULACIÓN DE HIPÓTESIS

1.3.1 Hipótesis

Dado al creciente riesgo que presenta la manipulación de documentos digitales al ser estos en su mayoría, información vital de las personas y empresas, es probable que el uso de un sistema web para Intercambio de documentos electrónicos usando firmas digitales, certificados X509 y SSL incremente la seguridad y confidencialidad del contenido de dichos documentos.

1.3.2 Variables

- **Variables independientes**
  - Variable: Firmas digitales.
  - Variable: Certificados X.509.
  - Variable: SSL.
- **Variables dependientes**
  - Variable: Sistema web para intercambio de documentos usando firmas digitales, certificados X509 y SSL.
  - Indicadores:
    - Funcionalidad
    - Fiabilidad
    - Usabilidad
    - Eficiencia
    - Facilidad de Mantenimiento
    - Portabilidad.

1.4 SOLUCIÓN PROPUESTA

1.4.1 Justificación

Debido al incremento en el uso de las tecnologías de información y al uso cada vez más frecuente de internet como medio de transporte es que surgen más

vulnerabilidades y formas de alterar la información sin el consentimiento del remitente o el destinatario, es por ello que la seguridad debe ser una parte importante en el intercambio de documentos, debido a esto en el presente trabajo se propone un sistema web para aumentar la seguridad en el intercambio digital de documentos como una parte importante en la gestión de información electrónica.

1.4.2 Descripción de la solución

La solución propuesta para la seguridad documental hará uso de técnicas criptográficas como son la firma digital y protocolos SSL en donde se garantizará cuatro aspectos en la seguridad:

- Autenticación: Que la persona que firma el documento es quien dice ser.
- Integridad: Permite saber que el documento no ha sido alterado después de haber sido firmado.
- No repudio. Que la persona encargada de firmar un documento no pueda negar el haber realizado dicha acción.
- Confidencialidad: Que la información enviada solo pueda ser leía por las personas a las que va dirigida.

En cuanto a las funcionalidades que se desarrollarán en el sistema web se tienen las siguientes:

- El sistema web permitirá al usuario registrarse y mantener una lista de contactos con otros usuario registrados en el sistema.
- El sistema web permitirá al usuario subir sus documentos y firmarlos digitalmente.
- El sistema web permitirá que el usuario comparta los documentos que firme.
- El sistema web permitirá, a los usuarios autenticados, descargar los documentos firmados digitalmente.
- Para toda la comunicación entre el cliente y el servidor se hará uso de SSL para una comunicación segura.

1.4.3 Alcances y limitaciones

- El presente trabajo está limitado al uso de la criptografía y uso de protocolos

seguros SSL, mas no se orienta a la seguridad de las redes y hardware usados.

- El sistema no generará las claves y certificados digitales, estos deberán ser ingresados al momento de firmar algún documento.
- **Viabilidad:** Es viable dado que no se contará con el costo de licencias para lo cual se desarrollará con la versión libre del JDK que ofrece JAVA y la potencia de los framework que proporciona, la base de datos a utilizar será PostgreSQL el cual es un sistema gestor de base de datos relacional.

1.5 ESTADO DEL ARTE

Sistema prototipo para firmar digitalmente documentos haciendo uso de criptografía asimétrica y el estándar de certificados digitales x.509 para la universidad Católica de Santa María (Autor: Fuentes Revilla José Andrés, 2011, Universidad Católica Santa María): Se propone un prototipo para firmar digitalmente documentos oficiales emitidos por la oficina de registro y archivo académico, proveyendo al alumno un medio electrónico que avale sus estudios realizados en la universidad, se utilizan los certificados X509 y además se realiza TimeStamp al momento de la firma digital de tal manera que quede constancia de la fecha de firma.

Implementación de un sistema de voto electrónico utilizando criptografía cuántica (Autores: Rojas Villena Johana María, Talavera Portocarrero Jesús Martin, 2008, Universidad Católica Santa María): Se propone un sistema de voto electrónico utilizando computación cuántica la cual es expuesta como el método que será usado a futuro en los sistemas de seguridad y autenticación, el protocolo que se usa en este trabajo es BB84 el cual proporciona un método en el cual dos partes intercambien una clave de encriptación con una seguridad absoluta basado en las leyes que la computación cuántica ofrece.

Implementación de un web site de comercio electrónico utilizando una infraestructura de red segura: autoridad de certificación, usando un esquema PKI para generación de firmas y certificados digitales (Autores: Víctor Manuel Ponce Diaz, Wilson Fernando Peñafiel Anchundia, Christian Xavier Cobeña Pino, 2005, Escuela Superior Politécnica del Litoral): En este trabajo se expone la infraestructura PKI y se implementa los servicios que debe cumplir una entidad de certificación de tal manera que esta provea el par de claves necesarias para la firma digital; también se consideran otros aspectos de la seguridad como antivirus, firewall, detección de intrusos y administración de usuarios.

Diseño integral de una VPN utilizando encriptación en terminales wireless en multiplataformas (Autor: Yoél Arturo Ramos Moscoso, 2005, Universidad Panamericana - México): En este trabajo se profundiza en temas de autenticación y criptografía, se resalta la importancia de la autenticación de usuarios en internet, relacionado a este tema se hace mención a las firmas digitales y algoritmos de llaves públicas.

Creación de una empresa proveedora de certificados de firma digital para operadores de comercio exterior (OCE's) en el Ecuador (Autores: Dalila Marisela Irobo Borja, Christian Javier Alvarez Villalva, Oscar Mendoza, 2005, Escuela Superior Politécnica del Litoral - Ecuador): Se presenta una base teórica sobre infraestructura, funcionamiento y la logística que requiere un entidad certificadora para la emisión, verificación y revocación de certificados digitales.

La firma electrónica y las entidades certificadoras (Autor: Alfredo Alejandro Reyes Kraft, 2002, Universidad Panamericana - México): En este trabajo se describen conceptos de firma, firma electrónica, firma digital, certificados digitales y entidades certificadoras, tomando como referencias a empresas extranjeras privadas y públicas.

1.5.1 Discusión

Las tesis revisadas en nuestro estado del arte nos sirvieron para tener una base teórica y referencia sobre la firma digital y el modo en que las autoridades de certificación operan.

Asimismo uno de los trabajos con fecha más reciente y que tiene mayor símil con nuestro tema es “Sistema prototipo para firmar digitalmente documentos haciendo uso de criptografía asimétrica y el estándar de certificados digitales x.509 para la universidad Católica de Santa María”, dicho trabajo fue realizado como una solución a medida para la universidad integrándose a sistemas ya existentes utilizando el lenguaje de programación C# y ejecutándose como una aplicación de consola en la máquina cliente para firmar documentos. Nuestro trabajo a diferencia de este propone un sistema web bajo una arquitectura cliente-servidor, con lo cual buscamos que el sistema sea accesible a cualquier persona que disponga de un navegador web y una conexión a internet. Cada cliente dispondrá de una cuenta de usuario en el sistema desde el cual podrá subir documentos, firmarlos y compartirlos con sus contactos registrados. Todas las operaciones que se realizan sobre los documentos (firmas digitales, asignación de firmantes, observaciones efectuadas) son notificadas a través de los correos de los usuarios registrados en el sistema, además se mantiene un historial por cada documento subido; con esto buscamos un sistema más abierto a las necesidades de las personas que requieran compartir documentos de forma segura, comunicarse con las personas asociadas al documento y tener el respaldo de la firma digital.

CAPÍTULO 2: MARCO TEÓRICO

2.1 CRIPTOGRAFÍA

Rama inicial de las Matemáticas y en la actualidad también de la Informática y de la Telemática, que hace uso de métodos y técnicas con el objeto principal de cifrar, y por tanto proteger, un mensaje o archivo por medio de un algoritmo, usando una o más claves. **[RAMI06]**

Así pues ya desde la antigüedad el hombre tuvo la necesidad de transmitir información sin que otros hombres enemigos tuvieran la posibilidad de obtenerla. De esta necesidad de mantener secretos surgió la criptografía, el arte de ocultar la información.

Una de las formas de clasificación de la criptografía es según el tipo de claves en el cual tenemos la criptografía simétrica y la criptografía asimétrica, en el cual todos los algoritmos criptográficos clásicos son de carácter simétrico, ya que hasta mediados de los años setenta no nació la criptografía asimétrica.

2.1.1 Criptografía simétrica

Se basa en el hecho de usar una misma clave para cifrar y descifrar el mensaje, debido a este motivo es que el emisor necesita enviar la clave para que el receptor pueda descifrar el mensaje.

Uno de los sistemas simétricos más conocidos es DES, otro también muy usado es IDEA, pero es AES por su extrema seguridad el que más se ha extendido hasta ahora.

2.1.2 Criptografía asimétrica

La gran ventaja de los sistemas simétricos es la velocidad que emplean para el cifrado, sin embargo esto exige que el remitente y el destinatario hayan intercambiado previamente la clave por algún otro medio, lo que constituye un verdadero problema en entornos tan grandes y públicos como internet; este problema queda resuelto con el cifrado asimétrico y los algoritmos de clave

pública. Cada usuario dispone de dos claves, una privada y una pública, de manera que lo que se cifra con una, se descifra con la otra. El remitente cifra los mensajes con la clave pública del destinatario y éste los descifra con su clave privada. **[HUID05]**

De esta manera se elimina la necesidad del envío previo de la clave. El precio que hay que pagar por este aumento de la seguridad es un mayor coste computacional, como ejemplo se sabe que DES (algoritmo simétrico) trabaja a unos 20 Mbits/s mientras que el RSA (algoritmo asimétrico) funciona a una velocidad 100 veces menor. **[HUID05]**

Una de las aplicaciones del cifrado asimétrico es la firma digital en el cual para agilizar el tiempo de cifrado se implementan funciones unidireccionales de resumen (funciones hash), de esta manera en vez de firmar un documento se cifra el resumen del mismo. **[HUID05]**

Otro uso de la criptografía asimétrica es en el transporte de claves simétricas a través de un canal inseguro y a su vez aprovechar la ventaja en velocidad de estas para el cifrado, por ejemplo un uso habitual es usar RSA para el cifrado de las claves simétricas DES y usar DES para el cifrado de un texto plano. **[HUID05]**

Hasta la fecha han aparecido multitud de algoritmos asimétricos, la mayoría de los cuales son inseguros; otros son poco prácticos, bien sea porque el criptograma es considerablemente mayor que el mensaje original, bien sea porque la longitud de la clave es enorme. Se basan en general en plantear al atacante problemas matemáticos difíciles de resolver. En la práctica muy pocos algoritmos son realmente útiles. El más popular por su sencillez es RSA, que ha sobrevivido a multitud de ataques, si bien necesita una longitud de clave considerable. **[LUCE09]**

Se han propuesto muchos otros sistemas criptográficos con clave pública, tal como lo revelan los informes de las conferencias anuales sobre criptografía. Un

problema matemático, llamado el problema de la mochila (knapsack), ha sido la base de varios sistemas que no han llegado a prosperar porque muchas versiones han sido rotas. [HUID05]

2.1.2.1 Algoritmo Diffie-Hellman [HUID05]

Primer algoritmo de clave pública, enunciado por W. Diffie y M. Hellman en 1976, este algoritmo fue el punto de partida para los sistemas asimétricos; matemáticamente este algoritmo se basa en las potencias de los números y en la función mod (módulo discreto). Si bien el cálculo de potencias discretas es fácil, la obtención de su función inversa, el logaritmo discreto, no tiene una solución analítica para números suficientemente grandes.

La gran importancia de este algoritmo fue el haber dado inicio a los sistemas asimétricos, en la práctica sólo es válido para el intercambio de claves simétricas, y con esta funcionalidad es muy usado en los diferentes sistemas seguros implementados en Internet, como SSL (Secure Socket Layer) y VPN (Virtual Private Network).

En 1991 fue cuando el NIST propuso un sistema de firma nuevo para el sistema Diffie-Hellman, que denominaron **DSS** (*Digital Standard Signature*), entonces nació el algoritmo completo que hoy conocemos como **DH/DSS**.

2.1.2.2 Algoritmo RSA [HUID05]

Fue desarrollado en 1977 por Rivest, Shamir y Adleman, la seguridad de este algoritmo se basa en el hecho de que no existe una forma eficiente de factorizar números que sean productos de dos grandes primos; así las claves pública y privada se calculan a partir de un número que se obtiene como producto de dos números primos grandes.

Este algoritmo es de un nivel de seguridad alto pero deben tenerse en cuenta las siguientes precauciones.

- Claves demasiado cortas: si bien una clave de 768 bits proporciona un nivel de

seguridad aceptable, se recomienda no utilizar claves menores de 2048 bits.

- Ataques de intermediario: en realidad, puede darse en cualquier algoritmo asimétrico consiste en un tercero que intercepta los mensajes e inserta su propia clave para poder cifrar el envío de lo que se está transmitiendo; este punto se explicará más adelante con mayor detalle.
- No firmar un mensaje después de codificarlo ya que existen ataques que aprovechan este hecho.
- Si se utiliza RSA para conseguir secreto y como firma digital, entonces es preferible que cada usuario use claves distintas para cada uno de los dos propósitos. De esta forma, cada usuario tendría asignada una clave en el directorio público de claves de cifrado y otra distinta en el directorio público de firmas digitales. Esta separación es útil para dos propósitos. En primer lugar, ayuda a evitar el problema que surge cuando el módulo del emisor es mayor que el del receptor. En segundo lugar, dado que el RSA es débil frente a algunos ataques con texto escogido, tales ataques pueden verse facilitados si se utiliza la misma clave para ambos fines y, en consecuencia, es preferible evitarlo.

2.1.2.3 Algoritmo ELGamal

Fue descrito por Taher Elgamal en 1984 Fue diseñado en un principio para producir firmas digitales, pero posteriormente se extendió también para codificar mensajes. Se basa en el problema de los logaritmos discretos, que está íntimamente relacionado con el de la factorización, y en el de Diffie-Hellman.[LUCE09]

2.1.2.4 Algoritmo de rabin

El sistema de clave asimétrica de Rabin se basa en el problema de calcular raíces cuadradas módulo de un número compuesto. Este problema se ha demostrado que es equivalente al de la factorización de dicho número.

2.1.3 Encriptación basada en password (PBE) [LOPE07]

En el ámbito de la encriptación simétrica, PBE surge como la necesidad de

ofrecer una clave sencilla de recordar para él usuario a lo cual denominaremos "password".

El "password" estará formado por una o más palabras al cual después se le aplicará una función hash para obtener una clave binaria, la cual se usará para la encriptación simétrica; este proceso se encuentra esquematizado en la siguiente figura.

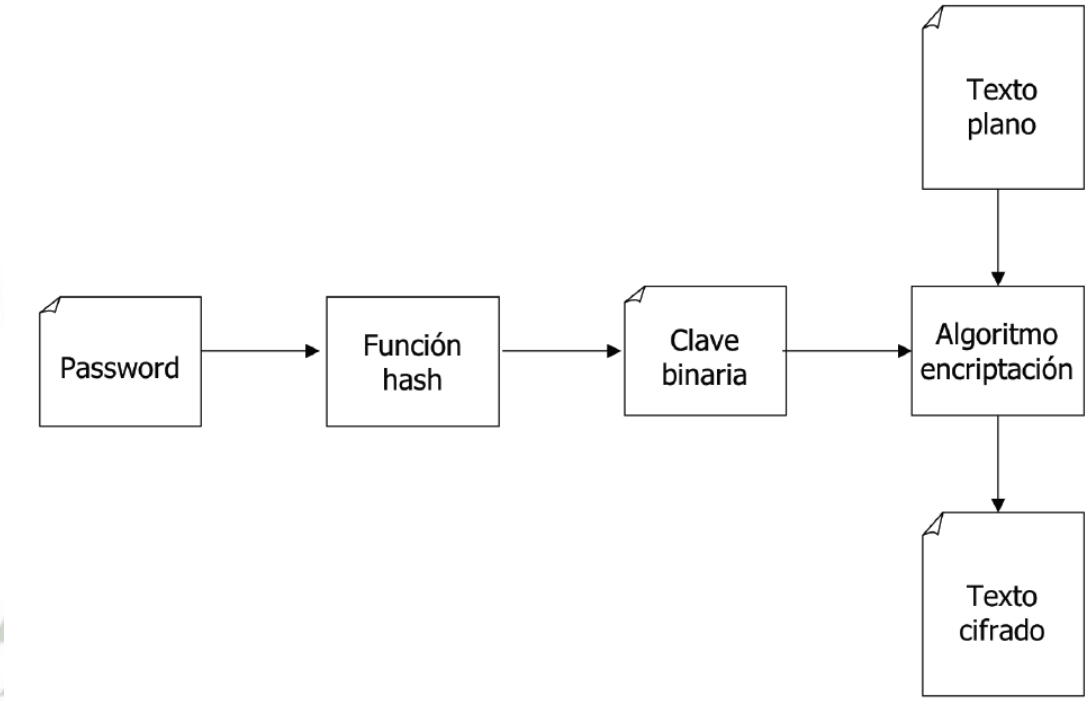


Figura 2.1 Obtención de clave binaria a través de un password.
Fuente: *"Seguridad criptografía y comercio electrónico con java" macprogramadores.org.*

El problema que tienen los passwords es que son mucho más fáciles de atacar por fuerza bruta que las claves binarias. Mientas que una clave binaria de 128 bits tiene 2^{128} combinaciones distintas, un password típico de 8 letras tiene 26^8 combinaciones (26 letras) que es aproximadamente 2^{33} combinaciones distintas.

A esto se añade que el usuario tiende a utilizar palabras comunes en el idioma, con lo que se vuelve muy efectivo un *ataque por diccionario* (ataque de fuerza bruta que a través de una base de datos de palabras comunes empieza a probar palabra por palabra hasta dar con la clave). Para dificultar este ataque se suelen usar tres estrategias:

1. **Passphrase:** El programa sugiere al usuario que en vez de dar una palabra se escriba una frase entera, esto con el propósito de aumentar el número de combinaciones.
2. **Salt:** Es un valor aleatorio que se concatena al password antes de pasarlo por la función hash, el salt se almacena sin encriptar dentro del mensaje encriptado, ya que es necesario conocerlo para calcular la clave binaria que se está usando.
3. **Iteration count:** Es utilizado con el fin de aumentar el tiempo necesario para calcular el hash de cada password, básicamente el número de veces que hay que hacer el hash al salt junto con el password para calcular la clave binaria; por ejemplo si su valor es 1000, el ataque por fuerza bruta es 1000 veces más costoso.

2.1.4 Ataques de intermediario [LUCE09]

El ataque de intermediario (figura 2.2) puede darse con cualquier algoritmo asimétrico, dando lugar a un grave peligro del que hay que ser consciente, y tratar de evitar a toda costa. Supongamos que A quiere establecer una comunicación con B, y que C quiere espiarla. Cuando A le solicite a B su clave pública K_B, C se interpone, obteniendo la clave de B y enviando a A una clave falsa K_C creada por él. A partir de ese momento puede pasar lo siguiente:

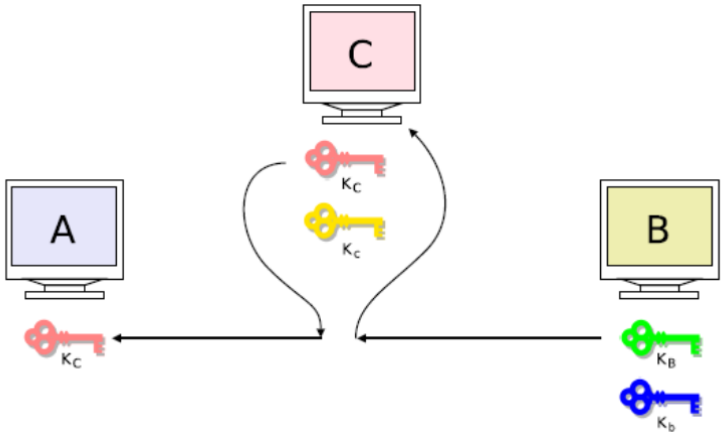


Figura 2.2 Ataque intermediario para un algoritmo asimétrico.
Fuente: *"Criptografía y seguridad en computadores" Universidad de Jaén.*

- Cualquier documento firmado digitalmente por C será interpretado por A como procedente de B.
- Si A cifra un mensaje para B, en realidad estará generando un mensaje cifrado para C, que podrá interceptarlo, descifrarlo con su propia clave privada, volverlo a cifrar con la clave K_B correcta, y reenviárselo a B. De esta forma C tendrá acceso a toda la información cifrada que viaje de A hasta B sin que ninguna de sus víctimas advierta el engaño.

  La única manera de evitar esto consiste en buscar mecanismos para poder garantizar que la clave pública que recibe A pertenece realmente a B. Para ello la solución más obvia consiste en que K_B esté firmada digitalmente por un amigo común, que certifique la autenticidad de la clave. Si A y B carecen de amigos comunes, pueden recurrir a los llamados anillos de confianza, que permiten certificar la autenticidad de las claves a través de redes sociales, en las que cada usuario está relacionado con unos cuantos y decide en quiénes confía, sin necesidad de centralizar el proceso. Por eso se nos suele recomendar, cuando instalamos paquetes de cifrado asimétrico, como PGP, que firmemos todas las claves sobre las que tengamos certeza de su autenticidad, y únicamente esas.

2.1.5 Ventajas y desventajas entre criptografía simétrica y asimétrica

	Ventajas	Desventajas
Criptografía simétrica	• Posee altas velocidades en los procesos de encriptación. • Las claves simétricas son relativamente cortas. • Pueden ser compuestos para crear cifrados más fuertes	• En una comunicación de dos partes se debe transmitir una clave la cual debe permanecer secreta en ambos lados. • En grandes redes habrán muchos pares de claves a ser gestionadas. • La práctica dice que la clave debe ser cambiada frecuentemente y posiblemente para cada sesión de comunicación.
Criptografía asimétrica	• Solamente la clave privada debe permanecer en secreto. • Dependiendo del modo de uso, un par de claves público y privada deben permanecer sin cambios durante un periodo considerable de tiempo. • En grandes redes, el número de claves necesario puede ser considerablemente más pequeño que un escenario de claves simétricas.	• Las velocidades de encriptación son mucho más lentos que los simétricos. • El tamaño de las claves son típicamente más grandes que las claves simétricas.

Tabla 2.1 Ventajas y desventajas entre criptografía simétrica y asimétrica.

Fuente: *Elaboración propia*

2.1.6 Curvas elípticas en criptografía [LUCE09]

La Criptografía de Curva Elíptica es una de las disciplinas más prometedoras en el campo de los cifrados asimétricos. Las curvas elípticas constituyen un formalismo matemático conocido y estudiado desde hace más de 150 años, y presentan una serie de propiedades que da lugar a problemas difíciles análogos a los que presentaba la aritmética modular. Si bien su estructura algebraica es algo compleja, su implementación suele resultar tanto o más eficiente que la aritmética modular, y además con claves mucho más cortas se puede alcanzar el mismo nivel de seguridad que con otras técnicas.

2.1.7 Gestión de claves [HUID05]

Para lograr un sistema criptográfico exitoso es necesaria la protección y gestión de claves. La gestión de claves consta de técnicas de generación, almacenamiento, distribución y mantenimiento, aplicada a la información almacenada y transmitida a través de las redes de ordenadores.

2.1.7.1 Generación de claves [HUID05]

Se deben generar claves pseudoaleatorias lo más impredecibles posible, existen generadores aleatorios de bits, los generadores mediante registros de desplazamiento, mediante la utilización de algoritmos matemáticos, generadores de secuencias, etc.

2.1.7.2 Almacenamiento de claves [HUID05]

Las claves deben conservarse en un lugar lo suficientemente seguro, existen varias soluciones entre las cuales tenemos, almacenar las claves en una tarjeta de banda magnética, una llave de plástico con un chip ROM o una tarjeta inteligente, una solución más simple sería almacenar la clave en una memoria flash del usuario.

Otra manera de almacenar las claves es encriptarlas con una clave fácil de recordar, como por ejemplo almacenar una clave privada RSA cifrada mediante una clave DES.

En los sistemas de clave pública existen un par de claves; la clave privada que

debe mantenerse en secreto y la clave pública que debe ser accesible por todo el mundo, por lo tanto dicha clave pública deberá residir en un lugar de máxima accesibilidad. El problema puede surgir para asegurar que las claves públicas pertenecen a quien dicen ser; para esto surge una entidad certificadora que almacena dichas claves y se comporta como un notario que asegura la identidad de los propietarios de las claves públicas; por otra parte si un usuario pierde su clave privada este deberá comunicarlo a la entidad certificadora para que sea revocada, otra forma de mantener la integridad de la clave privada es asignarle un periodo de validez.

2.1.7.3 Distribución de claves [HUID05]

Muchas veces el emisor debe enviar la misma clave con la que cifro el mensaje para que el receptor pueda descifrarlo; esta acción requerirá utilizar como medio de transporte canales seguros, así mismo también se deberá poder garantizar la identidad del origen de la clave, mantener su integridad y en el caso de claves secretas su confidencialidad.

En los sistemas de clave pública es vital evitar la suplantación de las claves usadas, el emisor obtiene una clave privada que es mantenida en secreto; y una clave pública que es almacenada por la entidad de certificación, la entidad certificadora se encarga de la autenticación de dicha clave además de aplicar métodos criptográficos incorporando marcas de tiempo y aplicando firmas digitales para que cualquiera pueda verificar que la clave pública del emisor es auténtica.

Otro método para asegurar la identidad de una clave pública es a través de anillos de confianza en donde cada usuario firma digitalmente las claves públicas en las que confía dando de esta manera un respaldo a dicha clave y creando un círculo de confianza entre usuarios asociados, en esta forma de autenticación se mantienen niveles de seguridad que pueden ir desde inseguridad, confianza parcial y confianza total.

2.1.7.4 Mantenimiento de claves [HUID05]

Se refiere al cambio periódico de las claves y a las acciones a tomar cuando son reveladas o robadas. Una clave nunca debería usarse por un tiempo indefinido. Debe tener una fecha de caducidad, se deben cambiar con cierta frecuencia y, en el caso de claves respaldadas por una entidad de certificación, comunicárselo a la entidad certificadora para que valide la nueva generación de claves.

2.2 FUNCIONES HASH

Junto a la criptografía asimétrica se utilizan en la firma digital las llamadas funciones hash o funciones resumen. Los mensajes que se intercambian pueden tener un gran tamaño, lo cual dificulta el proceso de cifrado, por ello no se cifra el mensaje entero sino un resumen del mismo obtenido aplicando al mensaje una función hash. [WWW01]

Partiendo de un mensaje determinado que puede tener cualquier tamaño, dicho mensaje se convierte mediante la función hash en un mensaje con una dimensión fija (generalmente de 160 bits). Para ello, el mensaje originario se divide en varias partes cada una de las cuales tendrá ese tamaño de 160 bits, y una vez dividido se combinan elementos tomados de cada una de las partes resultantes de la división para formar el mensaje-resumen o hash, que también tendrá una dimensión fija y constante de 160 bits. Este resumen de dimensión fija es el que es cifrado al momento de realizar una firma digital. [WWW01]

De lo anterior una función Hash debe cumplir con los siguientes criterios:

- La función Hash debe ser de longitud fija independientemente de la longitud del mensaje.
- Dado el mensaje debe ser fácil calcular la función hash, pero dada una función hash debe ser computacionalmente intratable recuperar el mensaje
- No pueden haber diferentes mensajes que tengan como resultado una misma función hash.

Entre los algoritmos de funciones Hash tenemos:

2.2.1 SHA-1

SHA-1 fue ideado por el NIST (National Institute of Standards and Technology) en 1994 como ampliación al algoritmo SHA. Se trata de una función criptográfica de tipo hash que acepta una entrada de 2^64 bits como máximo (2048 Terabytes) y devuelve como salida una cadena de **160 bits.**

Dado el tiempo en que los rumores puedan ser ciertos sobre la vulnerabilidad de SHA-1, los criptoanalistas recomiendan migrar los algoritmos de hash de las aplicaciones de firma electrónica hacia funciones RIPEMED-160 (recomendación de ETSI) o SHA-2 (SHA-1 con más bits).

2.2.2 SHA-2

NIST publicó cuatro funciones hash adicionales (SHA224, SHA256, SHA384, SHA512) conocidas colectivamente como SHA-2, estas variantes del hash fueron nombradas de acuerdo a la cantidad de bits de salida que dan como resultado.

2.2.3 MD2

Desarrollado por Ron Rivest, Optimizado para máquinas de 8 bits se considera insegura dada la cantidad de colisiones que posee.

2.2.4 MD5

Desarrollado por Ron Rivest, supone la evolución de los algoritmos MD2 y MD4, ha sido uno de los algoritmos hash más usados. Procesa mensajes de una longitud arbitraria en bloques de 512 bits generando un compendio de 128 bits. Debido a la capacidad de procesamiento actual esos 128 bits son insuficientes, además de que una serie de ataques criptoanalíticos han puesto de manifiesto algunas vulnerabilidades del algoritmo, por lo que ya no se aconseja su uso desde el punto de vista del criptoanálisis.

2.2.5 RIPEMD160

RIPEMD-160 fue ideado por Hans Dobbertin, Antoon Bosselaers, y Bart Preneel como ampliación del algoritmo RIPEMD. Se trata de una función criptográfica de tipo hash que acepta como entrada un mensaje de cualquier

longitud y devuelve como salida una cadena de 160 bits.

A pesar de haberse desarrollado mucho más libre que SHA-1, no es muy popular y tampoco ha sido muy estudiado por criptólogos. No obstante existen dos extensiones de este algoritmo (que son menos usadas aún) denominadas RIPEMD-256 y RIPEMD-320. Las longitudes de sus salidas son respectivamente 256 y 320 bits, con lo que se reducen significativamente las colisiones débiles y fuertes.

2.2.6 GOST3411

Desarrollado por " Federal Agency Government Communication and Information" y "All-Russian Scientific and Research Institute of Standardization", produce una salida hash de 256 bits y es usado solamente con el algoritmo **GOST3410** para realizar firmas digitales.

2.3 FIRMAS DIGITALES

Una firma digital es una secuencia de bits que se añade a una pieza de información cualquiera, y que permite garantizar su autenticidad de forma independiente del proceso de transmisión, tantas veces como se desee, presenta una analogía directa con la firma manuscrita, y para que sea equiparable a esta última debe cumplir los siguientes principios:

- **Garantizar la identidad del firmante,** Para garantizar la identidad del firmante se emplea la tecnología de par de claves vinculada a los datos que identifican al titular del certificado. De este modo, cuando se firma un documento se emplea un número único que sólo pertenece al firmante. El receptor del documento verifica la firma con la parte pública de la clave, de este modo, si el proceso de validación es positivo, debe concluirse que el firmante del documento es el titular del certificado.

- **Garantizar que el documento no ha sido modificado tras ser firmado**, La integridad del documento no se refiere al hecho de validar el contenido, sino de garantizar que el documento no ha sido modificado tras su firma. Para garantizar

esto no es necesario que un tercero custodie una copia del documento sino que se realiza generando un código único del documento a partir de su estructura interna en el momento de ser firmado. Cualquier alteración del contenido del documento provocará que al aplicar de nuevo la función de generación de código único sea imposible reproducir el original, por tanto, quedará rota la integridad del contenido.

- **No repudio**, Jurídicamente implica que el firmante no pueda negar haber firmado. Entre otros, los elementos que garantizan el no repudio son los siguientes:
  - La clave privada vinculada al certificado y que confiere unicidad a los documentos firmados, que sólo esté en posesión del firmante desde el mismo momento de generar dichas claves y vincularlas a sus datos de identificación.
  - El certificado y los dispositivos de firma empleados deben basarse en tecnologías y procesos seguros que eviten el uso o sustracción de la clave por parte de terceros y que se encuentren homologados por la Autoridad de Certificación emisora del certificado empleado.
  - Que el certificado esté activo en el momento de ser empleado. Esto equivale al estado de las tarjetas de crédito que también pueden ser revocadas por el interesado y caducar con el tiempo.
  - Que los receptores de documentos firmados dispongan de un instrumento de verificación seguro que no permita suplantar identidades del firmante o de la Autoridad de Certificación que realiza la validación.

- **Confidencialidad,** considerada como una característica opcional trata de que la información contendida haya sido cifrada, y la voluntad del emisor solo permite, que el receptor que él determine pueda descifrarla.

  La forma más extendida de calcular firmas digitales consiste en emplear una combinación de cifrado asimétrico y funciones resumen (hash) lo cual explicaremos a continuación.

Los conceptos de cifrado asimétrico que dan inicio a la firma digital fueron reconocidos muchos años antes de que cualquier realización práctica estuviera disponible. El primer método descubierto fue el esquema de firma RSA, el cual permanece hasta ahora como uno de las técnicas más prácticas y versátiles. Subsecuentes investigaciones han resultado en muchas técnicas alternativas de firma digital, algunas ofrecen ventajas significativas en términos de implementación y funcionalidad. **[MENE96]**

2.3.1 Proceso de firma digital

2.3.1.1 Firma digital de un mensaje electrónico **[WWW01]**

1. *Angel* (emisor) crea o *redacta un mensaje* electrónico determinado (por ejemplo, una propuesta comercial).
2. El emisor (Angel) aplica a ese mensaje electrónico una *función hash* (algoritmo), mediante la cual obtiene un resumen de ese mensaje.
3. El emisor (Angel) *cifra ese mensaje-resumen* utilizando su clave privada.

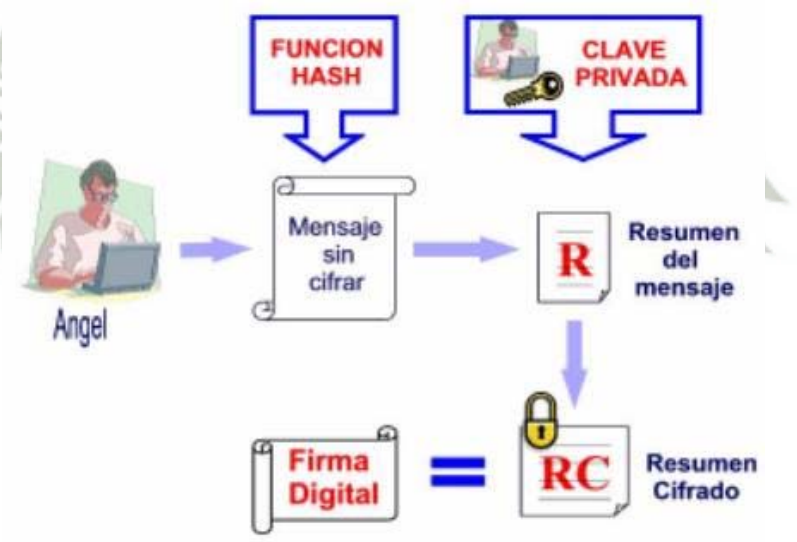


Figura 2.3 Firma digital de un documento.
Fuente: *Tramitación telemática de boletines de telecomunicaciones de FENIE.*

4. *Angel* envía a *Blanca* (receptor) un correo electrónico que contiene los siguientes elementos:
    a. El *cuerpo* del mensaje, que es el mensaje en claro (es decir, sin cifrar). Si se desea mantener la confidencialidad del mensaje, éste se cifra también pero utilizando la clave pública de Blanca (receptor).

b. La *firma* del mensaje, que a su vez se conforma del certificado digital y el resumen cifrado.

Figura 2.4 Envío de documento firmado digitalmente.
Fuente: *Tramitación telemática de boletines de telecomunicaciones de FENIE.*

2.3.1.2 Verificación del receptor de la firma digital del mensaje [WWW01]

1. Blanca (receptor) recibe el correo electrónico que contiene todos los elementos mencionados anteriormente.
2. Blanca en primer lugar *descifra el certificado digital* de Angel, incluido en el correo electrónico, utilizando para ello la clave pública del *prestador de servicios de certificación* que ha expedido dicho certificado. Esa clave pública la tomará Blanca, por ejemplo, de la página web del Prestador de Servicios de Certificación en la que existirá depositada dicha clave pública a disposición de todos los interesados.
3. Una vez descifrado el certificado, Blanca podrá acceder a la clave pública de Angel, que era uno de los elementos contenidos en dicho certificado. Además podrá saber a quién corresponde dicha clave pública, dado que los datos personales del titular de la clave (Angel) constan también en el certificado.
4. Blanca utilizará la clave pública del emisor (Angel) obtenida del certificado digital para *descifrar el hash* o mensaje-resumen creado y cifrado con la clave privada por *Angel***.**

5. Blanca *aplicará al cuerpo del mensaje*, que aparece en claro o no cifrado, que también figura en el correo electrónico recibido, la misma *función hash* que utilizó *Angel* con anterioridad, obteniendo igualmente *Blanca* un mensaje-resumen. Si el cuerpo del mensaje también ha sido cifrado para garantizar la confidencialidad del mismo, previamente *Blanca* deberá descifrarlo utilizando para ello su propia clave privada (recordemos que el cuerpo del mensaje había sido cifrado con la clave pública de *Blanca*).
6. *Blanca comparará* el mensaje-resumen o hash recibido de *Angel* con el mensaje-resumen o hash obtenido por ella misma. Si ambos mensajes-resumen o hash coinciden totalmente significa lo siguiente:
    a. El mensaje-resumen descifrado por Blanca con la clave pública de Angel ha sido necesariamente cifrado con la clave privada de Angel y, por tanto, proviene necesariamente de Angel.
    b. Como el certificado digital nos dice quién es Angel, podemos concluir que el mensaje ha sido firmado digitalmente por Angel, siendo Angel una persona con identidad determinada y conocida.

Por el contrario, si los mensajes-resumen no coinciden quiere decir que el mensaje ha sido alterado por un tercero durante el proceso de transmisión, y si el mensaje-resumen descifrado por Blanca es ininteligible quiere decir que no ha sido cifrado con la clave privada de Angel. En resumen, que el mensaje no es auténtico o que el mensaje no ha sido firmado por Angel sino por otra persona

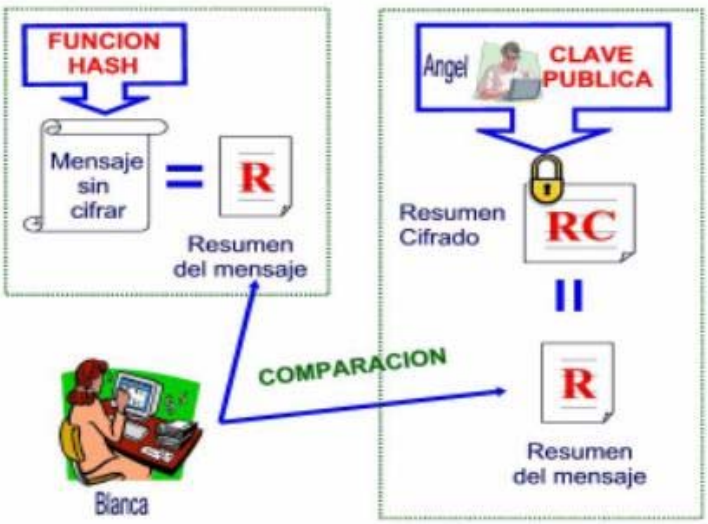


Figura 2.5 Recepción de mensaje firmado digitalmente.
Fuente: *Tramitación telemática de boletines de telecomunicaciones de FENIE.*

Finalmente, hay que tener en cuenta que las distintas fases del proceso de firma y verificación de una firma digital que han sido descritas no se producen de manera manual sino automática e instantánea, por el simple hecho de introducir la correspondiente tarjeta magnética en el lector de tarjetas de nuestro ordenador y activar el procedimiento.

2.3.2 Esquemas de firma digital [MENE96]

Existen dos esquemas de firma digital:

- Con apéndice, en el cual se necesita el mensaje original como parámetro de verificación, actualmente este esquema es el más usado y es en el que profundizaremos, dado al uso conjunto con funciones hash es menos propenso a ataques de falsificación.
- Con recuperación de mensaje, en este esquema el mensaje firmado puede ser recuperado de la firma digital, es más usado en mensajes cortos y de longitud fija.

2.3.3 Tipos de ataques en esquemas de firma digital [MENE96]

La meta de un atacante es falsificar firmas; lo cual consiste en producir firmas las cuales sean aceptadas como las que pertenecen a alguna otra entidad, los puntos siguientes proveen un criterio sobre lo que significa romper un esquema

de firma digital.

- **Quiebre total:** El atacante es capaz de computar la información de la clave privada del firmante, o encuentra un eficiente algoritmo de firma equivalente al algoritmo de firma válido.
- **Falsificación selectiva:** El atacante es capaz de crear una firma válida para un mensaje o conjunto de mensajes escogidos a priori. Crear la firma no involucra directamente al firmante legítimo.
- **Falsificación Existencial:** El atacante es capaz de falsificar una firma para al menos un mensaje. El atacante tiene poco o falta de control sobre el mensaje del cual la firma es obtenida, y el firmante legítimo podría no ser involucrado en el engaño.

2.3.4 Estándares de firma digital [WWW03]

Dentro de los estándares más usados tenemos PKCS (Public Key Cryptography Standards) el cuál se refiere a un grupo de estándares de criptografía pública concebidos y publicados por los laboratorios RSA en California, los estándares se especifican desde PKCS#1 hasta PKCS#15 dentro de estos haremos énfasis principalmente en 3 los cuales se usarán en el desarrollo del proyecto:

2.3.4.1 PKCS7

Define un conjunto de normas para firmar y encriptar documentos, su principal uso es en el mail para la privacidad y autenticidad de datos contenidos en ellos, normalmente la extensión usada en este fichero es ".p7s".

De objetos PKCS #7 podemos encontrar varios tipos:

- **Data:** Sólo datos, usado para enviar datos sin encriptar.
- **Signed sata:** datos firmados, usado para autentificación del remitente; dentro de signed data a su vez encontramos 2 tipos:
  - **Attached:** En el cual, los datos se adjuntan al fichero de firma resultante. Con lo que para una posterior verificación no se necesita ningún archivo adicional.
  - **Detached:** En el cual, los datos no se adjuntan en el fichero de firma resultante. Por lo tanto para una posterior verificación se deberá facilitar el documento original.

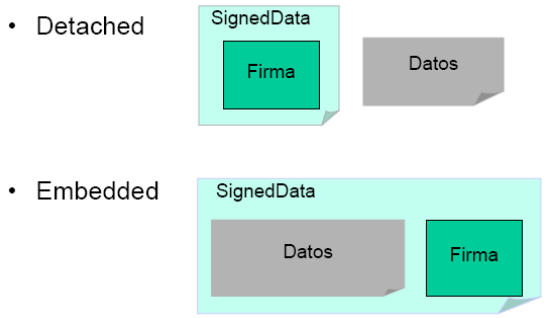


Figura 2.6 Tipos de formato mensajes firmados.
Fuente: *Elaboración propia*

- **Enveloped data:** datos juntos - o enfundados - que pueden ser datos o datos firmados o datos encriptados o varios de ellos a la vez, utilizados para confidencialidad.
- **Signed-and-enveloped data:** datos firmados y enfundados. Para autenticidad y confidencialidad.
- **Digested data:** datos resumidos, para comprobar la integridad del mensaje.
- **Encrypted data:** datos encriptados, utilizados para confidencialidad.

El formato PKCS7 se puede esquematizar de la siguiente manera:

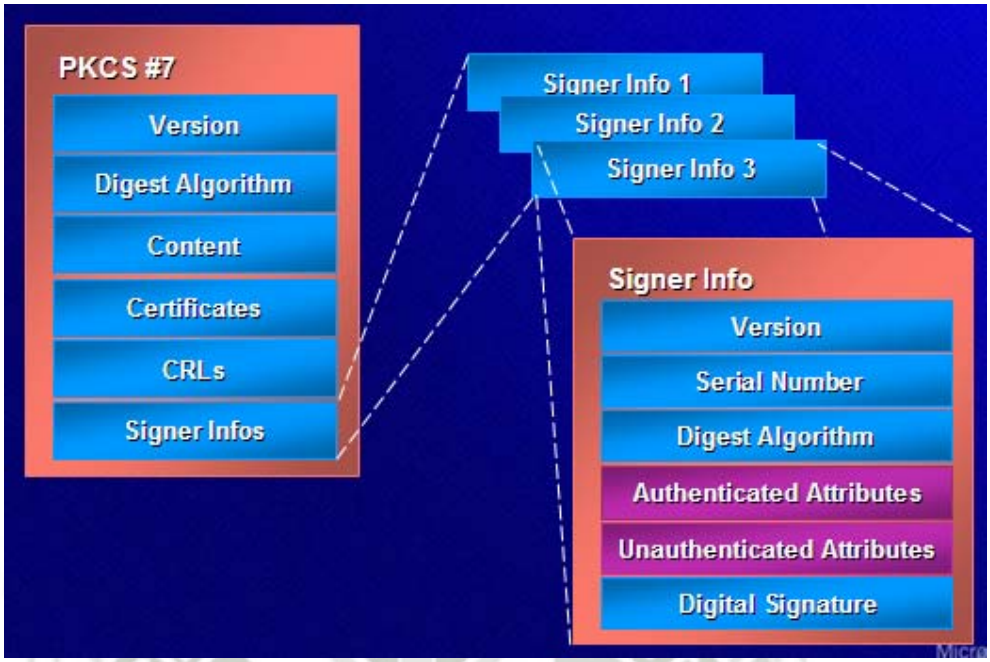


Figura 2.7 Formato PKCS7.
Fuente: *Microsoft TechNet*

2.3.4.2 PKCS10[WWW04]

Es el formato más común usado para una petición firmada de certificado o CSR (Certificate Signing Request), el cual consiste en un mensaje enviado a una autoridad de certificación para solicitar un certificado para firma digital.

Los procedimientos para generar un CSR son los siguientes:

- Antes de generar un CSR el solicitante debe generar un par de claves asimétricas, manteniendo la privada en secreto.
- El CSR contiene información que identifica al solicitante y la clave pública del solicitante todo esto debe ser firmado con la clave privada.
- El CSR puede ir acompañado de otras credenciales o pruebas de identidad requeridas por la autoridad de certificación, también puede darse el caso que la autoridad de certificación se contacte con el solicitante para obtener mayor información.

La información más típica en un CSR es la siguiente:

Información	Descripción
Nombre Propio	El nombre del solicitante o dominio web que se quiera certificar, por ejemplo: Juan Florez, en caso de dominio, www.ejemplo.com.
Nombre de Negocio / Organización	Usualmente el nombre legal de un empresa y debería incluir los sufijos: Ltd., Inc., o Corp.
Nombre de Departamento / Unidad Organizacional	Nombre del departamento que ocupa el solicitante en la empresa por ejemplo: Recursos Humanos, Finanzas, IT , etc.
Ciudad	Ciudad donde se reside por ejemplo: Londres, Paris, Nueva York, etc.
Provincia, Región o estado	No debería ser abreviado por ejemplo: Nueva Jersey , Normandía, etc.
País	El código ISO de dos letras del país donde se reside por ejemplo: FR, US, GB, etc.
Dirección de Correo Electrónico	Correo electrónico del solicitante para contacto.

Tabla 2.2 Información típica de una petición CSR
Fuente: Elaboración propia

2.3.4.3 PKCS12 [WWW05]

Define un formato de fichero usado para almacenar claves privadas con sus certificados x509; normalmente este fichero es protegido por una clave usando encriptación simétrica basada en password (PBE); la extensión comúnmente usada para este fichero es ".p12"; PFX es el sucesor de este formato propuesto por Microsoft.

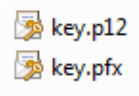


Figura 2.8 Íconos de archivos PKCS12
Fuente: *Elaboración propia.*

2.3.5 Legislatura en el Perú de la firma digital y electrónica [ROME05]

La firma electrónica es un género, caracterizado por el soporte a todo modo de identificación de auditoría basado en medios electrónicos; luego vienen las especies, que en general, se caracterizan por agregar elementos de seguridad que la sola firma electrónica no posee. Las legislaciones reconocen el género de la firma electrónica y luego eligen una especie que denominan firma digital, que es la que utiliza un sistema, generalmente criptográfico, que da seguridad.

El objetivo de la ley peruana con respecto a la firma electrónica, es regular su utilización otorgándole la misma validez y eficacia jurídica que el uso de una firma manuscrita u otra análoga que conlleve manifestación de voluntad.

El Art 1 de la ley de Firmas y Certificados Digitales, Ley N° 27269, define la Firma Electrónica como “cualquier símbolo basado en medios electrónicos utilizado o adoptado por una parte con la intención precisa de vincularse o autenticar un documento cumpliendo todas o algunas de las funciones características de una firma manuscrita”.

Esta firma presenta un nivel de seguridad jurídica particularmente débil ya que su falsificación aparece tremendamente sencilla, la autenticidad de los documentos no puede ser garantizada, por consiguiente, tal empresa tendrá que preferir para sus transacciones en línea el uso de un sistema de firma digital.

El Art. 3 de la misma ley define la Firma Digital como “Aquella firma electrónica que utiliza una técnica de criptografía asimétrica, basada en el uso de un par de claves único, asociadas una clave privada y una clave pública relacionadas matemáticamente entre sí, de tal forma que las personas que conocen la clave pública no puedan derivar de ella la clave privada”.

Varios problemas se plantean en cuanto a la fiabilidad de estas operaciones, como garantizar la autenticidad de la identidad de aquél que ha efectuado la operación o garantizar la integridad del documento, si este no ha sido

modificado o falsificado durante su transmisión, una firma digital ofrece todas las garantías.

2.4 CERTIFICADO DIGITAL

EL certificado digital es una pieza de información en la que se asocia el nombre de una entidad con su clave pública durante un periodo de validez, y que es firmado por una Autoridad Certificadora (CA). **[CARR04]**

Tanto emisor y receptor confiarán en esa CA, el usuario que tenga un certificado expedido por ella se autenticará ante el otro, en tanto que su clave pública está firmada por dicha autoridad. **[RAMI06]**

Es importante hacer notar que el certificado es una pieza de información segura en sí misma que, en principio, puede ser conocida por todos sin ningún género de restricciones. Su fortaleza se obtiene de la firma de la CA correspondiente y de la confianza que el receptor del certificado tiene con esa CA. **[CARR04]**

La confianza en el certificado proviene de dos cosas: **[CARR04]**

a. La confianza en la CA, todas las entidades provenientes de un determinado dominio se fían de que cuando expide un certificado de la entidad A, antes se ha comprobado adecuadamente la identidad de A y el valor de su clave pública. Confían, además, en que no les va a engañar, bien porque han participado de la decisión de quien administra la CA, bien porque tienen que aceptarlo si desean entrar a formar parte de un determinado dominio.
b. Todos los miembros del dominio de seguridad conocen la clave pública de la CA y están seguros de que es válida, gracias a eso, al verificar la firma de la CA están seguros de la validez de la clave pública de la entidad certificada.

2.4.1 Formato de certificado X.509

Cuando los comités de ISO y de ITU-T encargados de especificar los servicios y protocolos de seguridad en redes telemáticas contemplaron la necesidad de

definir el certificado digital, pensaron que el sitio adecuado para almacenar esta pieza de información seria el Servicio de Directorio, DS (Directory Service), más conocido como X.500, su nombre de serie en ITU. Fue concebido como el sistema distribuido que soportaría una base de datos de rango planetario, la DIB (Directory Information Base), destinada a contener toda la información de las entidades comunicantes que fuese relevantemente a los efectos de los restantes servicios telemáticos. **[CARR04]**

En X.509 se define un framework (una capa de abstracción) para suministrar servicios de autenticación a los usuarios del directorio X.500. **[RAMI06]**

El estándar X.509 sólo define la sintaxis de los certificados, por lo que no está atado a ningún algoritmo en particular, el formato X.509 tuvo que pasar por distintas versiones y poder evolucionar a la versión 3 que es la más usada y difundida.

2.4.1.1 Versiones 1 y 2 del certificado X.509 [CARR04]

En la figura 2.10 se presenta el esquema de la estructura del certificado X.509 conforme a las dos primeras versiones aparecidas en 1988 y 1993 respectivamente. El nombre y contenido de los 7 primeros campos es el siguiente:

- **Versión**

  Indica la versión del certificado conforme a la cual están definidos formalmente sus distintos campos. El valor 0 indica versión 1 y el valor 1, versión 2.

- **Número de serie**

  Cada CA deberá ir numerando correlativamente todos los certificados que emita, de forma que el número de serie sirve de identificador único para este certificado.

- **Algoritmo de firma del certificado (signature)**

  Representa el algoritmo de cifrado que ha utilizado la Autoridad Certificadora (CA) en la firma del certificado. Normalmente será RSA o

DSA, pero tal y como está concebido el certificado X.509 puede utilizar cualquier otro algoritmo de encriptación asimétrica.

- **Nombre de la CA emisora (issuer)**

  En la especificación del certificado está previsto que este campo recoja el nombre X.500 de la CA.

- **Validez**

  Indica el comienzo y el final del período de tiempo durante el cual el certificado es válido.

- **Nombre del usuario o titular (subject)**

  Es el nombre X.500 de la entidad adscrita a esta CA a la que se la ha expedido el certificado. Puede tratarse de una CA a la que otra CA la haya generado un certificado.

- **Información de la clave pública (subject public key)**

  Es el componente principal del certificado, consta de 2 elementos:

  - **Algoritmo con el que será usada:** identificador del algoritmo con el que se ha previsto que la clave pública sea usada.
  - **Valor de la clave pública (subject public key):** es el valor de la clave pública de la entidad propietaria del certificado.

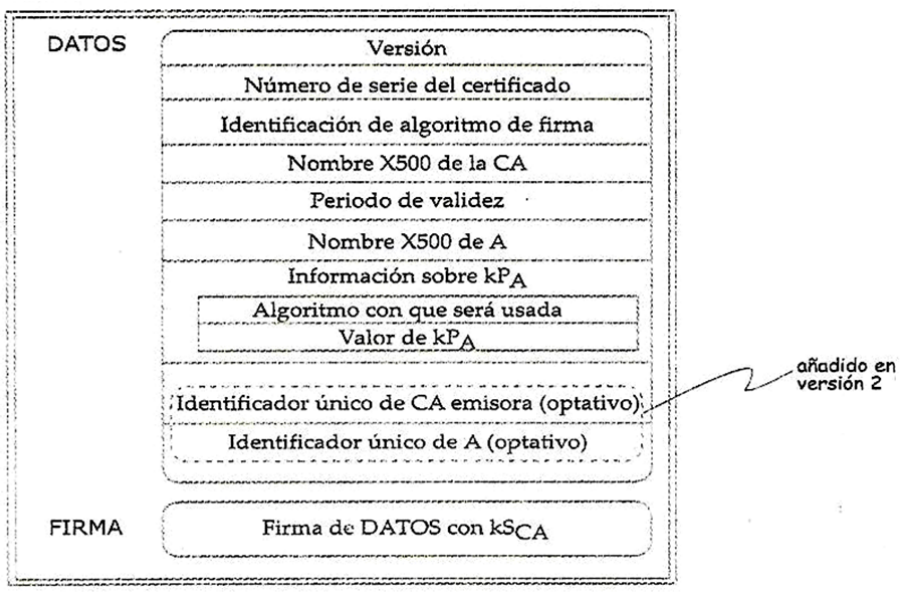


Figura 2.9 Estructura de las versiones 1 y 2 del certificado X.509
Fuente: *"Seguridad en redes telemáticas", Carracedo Justo.*

Años después se pensó que sería conveniente aprovechar la existencia de una pieza de información tan robusta y segura para incluir al menos, algunos aspectos con la adecuada utilización del certificado, y fue por este criterio que por un "sí pero poco" se aprobó la versión 2 del certificado incluyendo los siguientes campos **optativos**.

- **Identificador único de CA emisora**
  Se trata de una cadena de bits, sin formato específico, que de forma opcional sirve para contener información adicional sobre la CA emisora del certificado.
- **Identificador único de la entidad propietaria**
  Análogamente al caso anterior, serviría para contener información adicional sobre la entidad a nombre de la cual se ha expedido el certificado.

Al ser estos campos una cadena de bits (bit String), sin sintaxis ni semántica puntualmente definida, podían ser usados por gestores de dominio de acuerdo a reglas internas para guardar datos que se consideren convenientes, de tal manera que una entidad ajena al dominio no podría interpretar la información de los campos. Por otra parte al ser optativos contribuía a no perder la validez global del certificado, ya que los demás campos si podrán ser interpretados por cualquier entidad que conociese la estructura del certificado X.509 V1.

Como ha podido observarse para la identificación de las entidades (CA emisora y entidad adscrita a dicha CA). Sólo se contempla el **nombre X.500**. Cuando se emitió la norma se esperaba que el servicio de directorio se estableciese a lo largo y ancho de las redes de todos los países de tal manera que este nombre fuese un nombre distintivo DN (Distinguished Name) compuesto por campos jerárquicamente organizados, cada uno de estos campos diferenciadores de denomina RDN(Relative Distinguished Name).

Supongamos que el nombre de un empleado es Alicia Gómez, que está

laborando como secretaria de Recursos Humanos en el diario “El Comercio”. El campo de mayor jerarquía correspondería al del país en el que está ubicada la organización a la que pertenece la entidad comunicante nombrada. El identificador de este campo es <<C<< (del inglés country) y en este caso sería C = pe. El segundo corresponde a la *Organización*, que aquí suponemos que es: O = El Comercio. A continuación puede aparecer un número indeterminado de campos correspondientes a diversas unidades organizacionales jerárquicamente dependientes de la institución catalogada como <<organización>>, el identificador de estos campos es <<OU>>, del inglés *Organization Unit*. En este ejemplo tenemos dos de esas unidades organizativas: Recursos Humanos y Secretaría. Los valores de estos dos RDNs serían: OU = RR. HH. y OU = Secretaría. Por último aparecería el RDN que contiene el nombre propio (Common Name) del usuario, es decir CN= Alicia Gómez. El conjunto de los distintos RDNs es el nombre X.500 completo:

{C = pe, O = El Comercio, OU = RR. HH., OU = Secretaría, CN = Alicia Gómez}.

2.4.1.2 Versión 3 del certificado X.509

A pesar de la aprobación de la versión 2 la comunidad científica continuó trabajos conducentes a una nueva configuración que supliese las insuficiencias de anteriores versiones. La opción que se consideró como la más adecuada fue la inclusión del campo de extensiones, que permite definir un número indeterminado de campos adicionales donde recoger los datos que resulten de interés para la política de seguridad definida en el dominio para el que el certificado es generado.

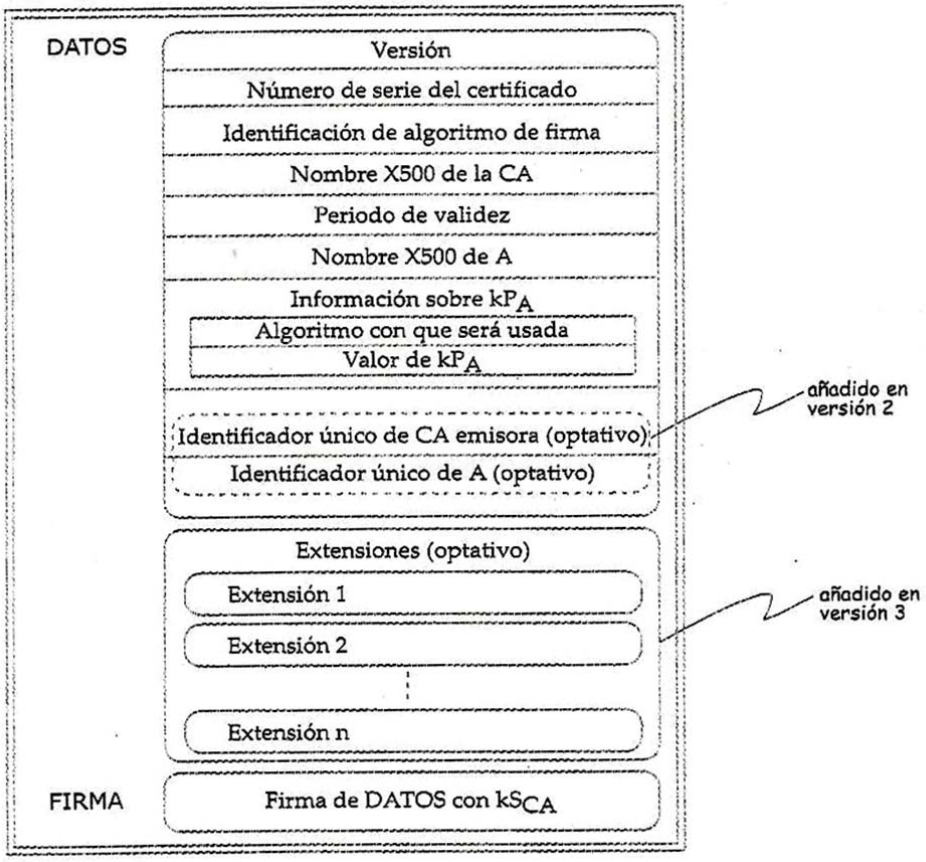


Figura 2.10 Estructura del certificado X.509 V3
Fuente: *"Seguridad en redes telemáticas", Carracedo Justo.*

Además de ser optativas en la definición de cada extensión existe un campo de valor binario que ilustra si la extensión es o no crítica. En un determinado entorno cuando una entidad recibe un certificado con una extensión crítica debe ser capaz de interpretar dicho campo y tenerla en cuenta de otro modo cualquier acción desempeñada tomando como base el certificado será considerada como no válida, por ejemplo si se dice que una determinada clave sólo puede ser usada para el acceso seguro a través de internet, cualquier otro uso que quiere dársele puede considerarse como ilícito e incluso traer consigo repercusiones penales. Por otro lado si la extensión está descrita como no crítica el hecho que una entidad quiera o no tener en cuenta el valor de la extensión no le quita validez al uso del certificado.

Por último el tercer campo representa el valor de la extensión propiamente dicha.

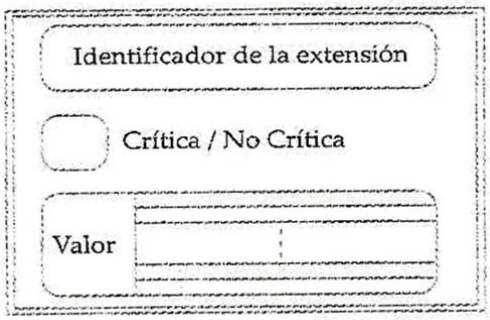


Figura 2.11 Estructura de las extensiones del certificado X.509 V3
Fuente: *"Seguridad en redes telemáticas", Carracedo Justo.*

2.4.2 Revocación y suspensión de certificados

El certificado es una pieza de información segura en sí misma. Sin embargo, existen circunstancias que provocan la invalidez de un certificado aunque no haya concluido el periodo de validez que aparece explícitamente reflejado en uno de sus campos. Algunas de estas causas son de tipo organizativo, como el hecho de que un determinado usuario ha dejado de pertenecer a alguna organización, pero las más graves son aquellas que se producen cuando el propietario de un certificado teme que la clave privada, pareja de pública certificada, no está tan segura como debía a causa de un descuido en su custodia o ante el temor de los efectos de un ataque que haya sufrido. **[CARR04]**

La entidad con capacidad para revocar o suspender un certificado es la CA que lo generó. El certificado revocado o suspendido puede pertenecer a una entidad usuaria final o a otra CA cuya clave haya sido certificada. En este último caso, los efectos de la revocación se expanden hacia todos los certificados que se hayan emitido por ella.

Las causas por las que puede ser revocado un certificado son las siguientes:

- Clave de entidad propietaria comprometida.
- Clave de CA comprometida.

- Cambio de afiliación.
- Certificado reemplazado.
- Cese de operación.
- Borrado de la CRL.
- Suspensión de certificado.

Hay dos estados diferentes de revocación definido en RFC 3280:

- **Revocado (Revoked):** Un certificado es irreversiblemente revocado si, por ejemplo, la CA (Autoridad de Certificación) ha descubierto que ha emitido inapropiadamente un certificado, o si se piensa que la seguridad de una clave privada ha sido comprometida, por ejemplo el usuario denuncia un robo o cree que no es el único con acceso al contenido de la clave privada. Los certificados también pueden ser revocados por falla en la identidad del propietario del certificado de acuerdo a las políticas requeridas, como la publicación de documentos falsos o violación de cualquier otra política especificada entre la CA y el cliente.
- **Retenido (Hold):** Este estado es reversible y puede ser usado para invalidar temporalmente el certificado, por ejemplo el usuario esta inseguro sobre si la clave privada está perdida, si en nuestro ejemplo se encuentra que nadie tuvo acceso a la clave privada, el estado del certificado puede ser reinstaurado y cambiado a activo, por ende el certificado es removido de futuras CRLs.

2.4.2.1 CRL (listas de revocación de certificados)

En la operación de algunos sistemas criptográficos, usualmente los de infraestructura de clave pública (PKI), una CRL es una lista de certificados (más concretamente sus números de serie) que han sido revocados, ya no son válidos y en los que no debe confiar ningún usuario del sistema. **[WWW07]**

Una CRL es una lista de números de serie de certificados digitales revocados por una autoridad de certificación concreta. Dicha lista está firmada digitalmente por la propia autoridad de certificación.

Cuando un tercero desea comprobar la validez de un certificado debe descargar una CRL actualizada desde los servidores de la misma autoridad de certificación que emitió el certificado en cuestión. A continuación comprueba la autenticidad de la lista gracias a la firma digital de la autoridad de certificación. Después debe comprobar que el número de serie del certificado cuestionado está en la lista realizando una búsqueda secuencial. En caso afirmativo, no se debe considerar el certificado como revocado.

Estrictamente hablando, no es necesario descargar una CRL cada vez que se verifica un certificado, ya que estas se actualizan con cada cierta periodicidad, con lo cual una CRL deberá descargarse una vez sea actualizada.

Otra forma de averiguar si un certificado fue revocado es utilizando el protocolo OCSP (Online Certificate Protocol, RFC 2560) que da un resultado más exacto. Mediante este último se pueden realizar consultas sobre certificados específicos sin la necesidad de descargar una CRL.

2.4.2.2 OCSP (protocolo en línea del estado del certificado)

Online Certificate Status Protocol (OCSP) es un método para determinar el estado de revocación de un certificado digital X.509 usando otros medios que no sean el uso de CRL (Listas de Revocación de Certificados). Este protocolo se describe en el RFC 2560 y está en el registro de estándares de Internet. **[WWW08]**

Los mensajes OCSP se codifican en ASN.1 y habitualmente se transmiten sobre el protocolo HTTP. La naturaleza de las peticiones y respuestas de OCSP hace que a los servidores OCSP se les conozca como "OCSP responders". **[WWW08]**

2.4.2.2.1 Petición OCSP

Una petición OCSP está formada por:

- Versión del protocolo.
- Identificadores de los certificados a validar, un identificador puede estar formado por: número de serie del certificado, hash del Distinguised Name (DN) del emisor del certificado y el hash de la clave pública del

mismo.

- La firma electrónica de la petición es opcional y depende de lo que decida la autoridad de validación OCSP.

En una petición se pueden solicitar la consulta del estado de varios certificados, incluso pertenecientes a diferentes CA.

2.4.2.2.2 Respuesta OCSP

Un Responder OCSP puede devolver las siguientes respuestas, las cuales serán para cada uno de los certificados de los que se ha solicitado en la consulta:

- Respuesta firmada indicando el estado del certificado consultado que puede ser: bueno (good), revocado (revoked) o desconocido (unknown, en OCSP v.1 puede significar que el propietario del certificado es desconocido o que el estado de revocación del certificado es desconocido).
- Respuesta no firmada indicando un código de error.

2.4.2.2.3 Ventajas de OCSP sobre CRL

OCSP fue creado para solventar ciertas deficiencias de las CRL. Cuando se despliega una PKI (Infraestructura de Clave Pública), es preferible la validación de los certificados mediante OCSP sobre el uso de CRL por varias razones.

- OCSP proporciona una versión más reciente del estado de revocación de un certificado.
- OCSP elimina la necesidad de que los clientes tengan que obtener y procesar las CRL, de este modo ahorrando el tráfico de red y procesamiento en el cliente.
- El contenido de las listas CRL puede considerarse sensible en cambio en OCSP sólo muestra la información del estado de revocación del certificado consultado la cual no es sensible.
- OCSP soporta el encadenamiento de confianza de las peticiones OCSP entre los "responders". Esto permite que los clientes se comuniquen con un "responder" de confianza para lanzar una petición a una autoridad de certificación alternativa dentro de la misma PKI.
- Las CRLs pueden ser archivos de gran tamaño que necesitan recorrer la

lista secuencialmente, en cambio un responder OCSP usa un motor de base de datos para consultar el estado del certificado solicitado, con todas las ventajas y estructura para facilitar las consultas.

2.5 AUTORIDAD DE CERTIFICACIÓN

Una Autoridad de Certificación (CA), es una entidad privada o pública, que por sí misma o por medio de una Autoridad de Registro (RA), verifica la identidad del solicitante de un certificado antes de su expedición o, en caso de certificados expedidos con la condición de revocados, elimina la revocación de los certificados al comprobar dicha identidad.

La CA es una tercera parte de confianza (TTP) en la que confían los participantes en la comunicación que son miembros del dominio de seguridad de que se trate. Es utilizada para garantizar la propiedad y validez de la clave pública mediante la generación de un certificado de la clave pública firmado por la CA. **[CARR04]**

Las funciones de una Autoridad de Certificación deben ser, entre otras, las siguientes:

- Generación y Registro de claves.
- Identificación de Peticionarios de Certificados.
- Emisión de certificado.
- Almacenamiento en la AC de su clave privada.
- Mantenimiento de las claves vigentes y revocadas (CLRs y OCSP).
- Servicios de directorio.

Elementos de una CA

- Política de certificación.
- Certificado de la CA.
- Certificados de los usuarios (X.509).
- Protocolos de autenticación, gestión y obtención de certificados.

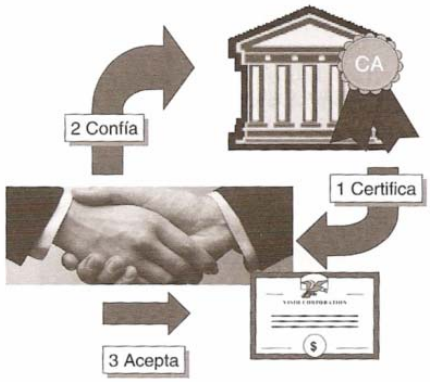


Figura 2.12 Modelo jerárquico de confianza
Fuente: "Seguridad en redes y sistemas criptográficos", Huidobro José M. Roldán.

2.5.1 Rutas de certificación

Los usuarios de aplicaciones de firma digital deben tener confianza que la clave pública de un propietario de certificado es genuina, para eso es que existen las Autoridades de Certificación (CA) que emiten los certificados y los firman con su clave privada dejando constancia de su generación. Si el usuario no posee una copia confiable de la clave pública de una CA que firmó la clave pública del propietario del certificado, entonces otra clave pública que correspondiente al ente firmante es requerida. Esta lógica puede ser aplicada recursivamente hasta que una cadena de certificados (o ruta de certificación) es descubierta, desde la clave pública de una autoridad Raíz (certificado de la CA más confiable) hasta el certificado del propietario. En términos generales una ruta de certificación es una lista ordenada de certificados, usualmente formadas desde el certificado del propietario, cero o más certificados y finalmente la clave pública de la CA más confiable.

La Figura 2.13 Ilustra una ruta de certificación una clave pública desde la clave pública de CA más confiable (CA1) hasta el propietario del certificado (Alice). La ruta de certificación establece la confianza en la clave pública de Alice a través de una CA intermediaria nombrada CA2.

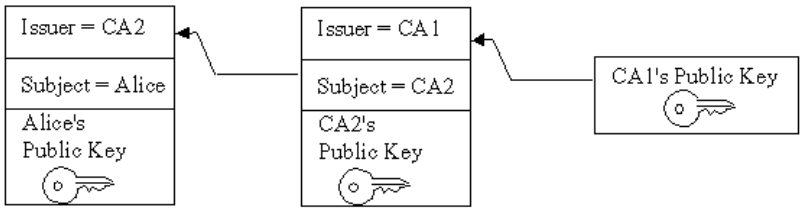


Figura 2.13 Ejemplo ruta de certificación
Fuente: *"Guía programación PKI", http://docs.oracle.com/.*

Antes de confiar en el certificado público de una persona se debe evaluar la ruta de certificación, para ello la "RFC 5280" define un algoritmo estandarizado de validación de ruta para certificados X.509. El algoritmo toma las siguientes entradas:

- La ruta de certificación a ser evaluada.
- El certificado Raíz de la ruta de certificación.
- Fecha y hora actual.
- Lista de Identificadores de Objeto (OID) de las Políticas de Certificados aceptadas por la CA.
- Indicadores ya sea sobre si el mapeo de políticas es permitido y como/cuando/o si un OID es tolerado.

En el algoritmo estandarizado se realizan los siguientes pasos (Esto es un resumen explicativo mas no una reproducción rigurosa detallada de los pasos), comenzando desde el certificado raíz. Si la evaluación falla en uno de los certificados, el algoritmo termina y la validación de la ruta falla.

- Se evalúa los parámetros y algoritmo de la clave pública.
- Se compara la fecha actual para corroborar que se encuentre dentro del periodo de validez del certificado.
- Se evalúa el estado de revocación, ya sea CRL u OCSP, o algún otro mecanismo.
- Se verifica que el que el nombre del emisor del certificado sea igual que

el nombre del propietario del anterior certificado en la ruta de certificación.

- Se verifica la restricción de nombre, para asegurarse que el nombre del propietario se encuentra en las lista de sub-árboles permitidos de todos los certificados CA previos.
- Los Identificadores de Objeto (OID) de Políticas de Certificados declarados son evaluados contra los OIDs permitidos del certificado previo, incluyendo cualquier mapeo equivalente declarado por el certificado previo.
- Las restricciones de básicas y de política son evaluadas, para asegurarse que cualquier requerimiento de política explícita no está siendo violada.
- Se asegura que la longitud de la ruta de certificación no exceda el límite máximo declarado en el certificado previo.
- Se evalúa la extensión "uso de clave" para asegurarse que la firma de certificados está permitida.
- Finalmente cualquier otra extensión crítica es reconocida y procesada.

Si este procedimiento llega hasta el último certificado de la cadena, sin restricciones de nombre, violaciones de políticas o cualquier otra condición error, entonces la validación de la ruta de certificación termina satisfactoriamente.

2.6 SECURE SOCKET LAYER (SSL)

Secure Socket Layer (SSL) es un protocolo criptográfico que proporciona comunicación segura a través de una red.

El protocolo de SSL opera en la capa de transporte de TCP/IP un nivel debajo de los protocolos específicos de aplicación tales como HTTP (web), SMTP (email) y NNTP (news), dando lugar en el primero de los casos a los servidores web seguros, cuya URL comienza por el prefijo https://. **[HUID05]**

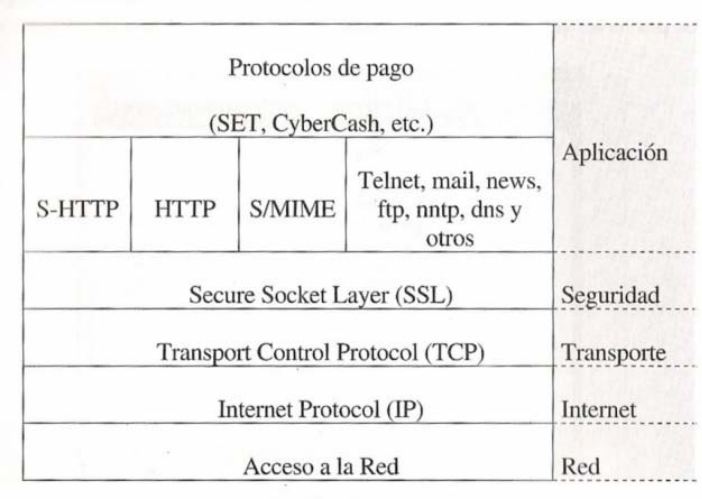


Figura 2.14 Pila de protocolos de SSL
Fuente: *"Seguridad en redes y sistemas criptográficos", Huidobro José M. Roldán.*

SSL utiliza llaves públicas y privadas para proveer un esquema de encriptación flexible que puede ser configurado al momento de iniciar la transacción segura.

Una de las ventajas de emplear un protocolo de comunicaciones en lugar de un algoritmo o algoritmos concretos, es que ninguna de las fases del protocolo queda atada a ningún algoritmo, por lo que si en el futuro aparecen algoritmos mejores, o alguno de los que se emplean en un momento dado quedara comprometido, el cambio se puede hacer sin modificar el protocolo. En la actualidad, las implementaciones típicas de SSL soportan algoritmos como RSA, Diffie-Hellman o DSA para la parte asimétrica y RC2, RC4, IDEA, DES, TripleDES o AES para la simétrica, y como funciones resumen SHA-1 o MD5.[LUCE09]

2.6.1 Funcionamiento del protocolo SSL

Para establecer una comunicación segura utilizando SSL, se tiene que seguir una serie de pasos. Primero, se debe presentar una solicitud de seguridad. Tras haberla hecho, se debe establecer los parámetros que se utilizarán para SSL. Esta parte se conoce como SSL Handshake. Una vez establecida una comunicación segura, se deben efectuar verificaciones periódicas para garantizar que la comunicación seguirá siendo segura a medida que se transmiten datos. Cuando

la transacción haya sido completada, se termina SSL. [IBEA00]

2.6.1.1 Solicitud de SSL

Antes de que se establezca SSL, se debe hacer una solicitud de un URL a un servidor que soporte SSL. SSL acepta solicitudes por un puerto distinto al utilizado normalmente para ese servicio. [IBEA00]

Una vez se ha hecho la solicitud, el cliente y el servidor empiezan a negociar la conexión SSL, es decir llevan a cabo el SSL Handshake. [IBEA00]

2.6.1.2 SSL Handshake

Durante el handshake se cumplen varios propósitos: Se autentifica el servidor y, opcionalmente, el cliente; se determina qué algoritmos de criptografía serán utilizados, y se genera una llave secreta que será utilizada durante el intercambio de los subsiguientes mensajes durante la comunicación SSL. [IBEA00]

La siguiente figura ilustra el proceso handshake

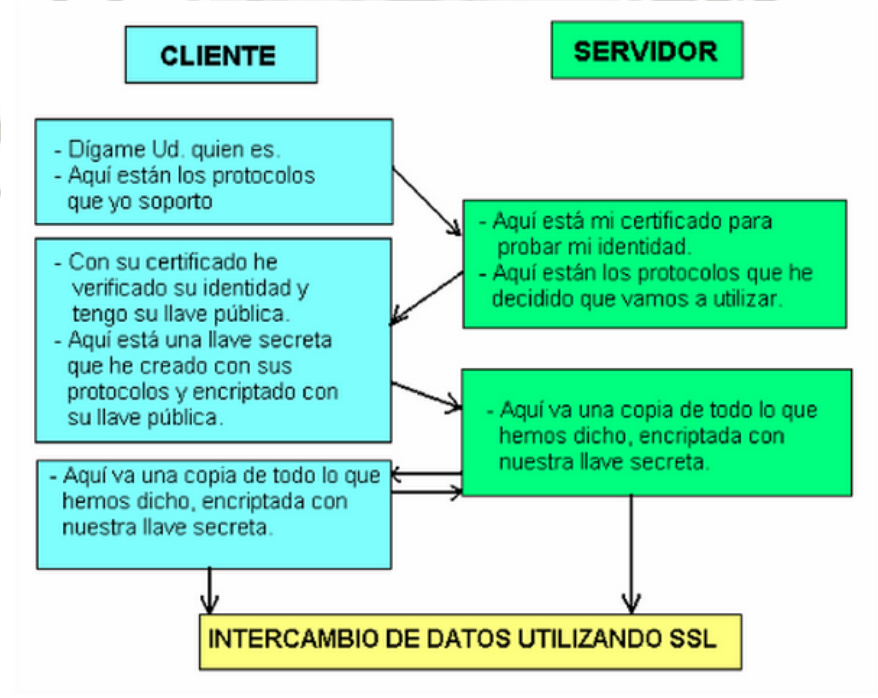


Figura 2.15 Proceso de HandShake en el protocolo SSL.
Fuente: *"E-Logistics(I) Nuevas Tecnologías de la Información", Ángel Ibeas Portilla, José Maria Días Pérez, Daniel de la Hoz Sánchez*

De acuerdo a la Figura 2.15 al final ambas partes se envían una copia de las últimas transacciones encriptadas con la clave secreta esto con el motivo de realizar una última verificación para comprobar que la información transmitida hasta el momento no ha sido alterada. Si ambas partes confirman la validez de las transacciones, el handshake se completa; si no es así, se reinicia el proceso.[IBEA00]

2.6.1.3 Intercambio de datos utilizando SSL

Ahora que se ha establecido un canal de transmisión seguro SSL, ya es posible llevar a cabo el intercambio de datos. [IBEA00]

Cuando el servidor o cliente desea enviar un mensaje al otro, se genera un digest (utilizando un algoritmo de hash de una vía acordado durante el handshake); luego se encripta el mensaje utilizando la clave privada previamente generada y finalmente se envía el mensaje. [IBEA00]

2.6.1.4 Termino de sesión SSL

Cuando el cliente abandona una sesión SSL, generalmente la aplicación presenta un mensaje en el que se advierte de que la comunicación no es segura y se confirma que el cliente, en efecto, desea abandonar la sesión SSL. [IBEA00]

CAPÍTULO 3: DESARROLLO DEL SISTEMA

3.1 METODOLOGÍAS ÁGILES DE DESARROLLO DE SOFTWARE

Es cierto que las metodologías tradicionales son efectivas en proyectos de larga duración y grandes recursos, pero para los proyectos pequeños estas normas son difíciles de seguir dado lo cambiante de sus requerimientos, la cantidad de tiempo disponible y los recursos a su alcance (físico, lógico y humano), al ser más complejas de aplicar se tiende a distorsionar las reglas y entonces se pueden producir contrariedades en el desarrollo.

Es por esto la necesidad de utilizar una metodología ágil de desarrollo, que intenta establecer un orden para los nuevos proyectos.

Cuatro cosas que valora una metodología ágil de desarrollo son:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.
- Desarrollar software que funcione más que conseguir una buena documentación.
- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

A continuación se elabora una comparación de cuatro metodologías agiles de desarrollo de software que fueron investigadas para el desarrollo de este proyecto.

	SCRUM	Extreme programming	OpenUP	AgileUP
Fases	Planificación de iteración Ejecución de iteración Inspección y adaptación	Exploración Planificación de entrega Iteraciones Producción Mantenimiento	Concepción Elaboración Construcción Transición	Concepción Elaboración Construcción Transición
Fundamentos	Comunicación Requisitos priorizados Desarrollo incremental Timeboxing Potenciación de equipo	Comunicación Simplicidad Retroalimentación Tenacidad	Requisitos priorizados. Colaboración Retroalimentación Enfoque en la arquitectura	Simplicidad Agilidad Actividades de valor primero Independencia de herramientas
Roles	Product owner Team Scrum master	Cliente Programador Tester Tracker Coach Big Boss	Analista Arquitecto Programador Administrador de proyecto Stakeholder Tester	DBA Modelador Programador Administrador de proyecto Stakeholder Administrador de pruebas
Artefactos de documentación	Product backlog Sprint backlog Sprint burn down chart	Realease planing Iteration planing Tarjeta de tarea Tarjeta CRC	Arquitectura Desarrollo Administración de proyecto Requerimientos Pruebas	Modelo de requerimientos Modelo de diseño Documentación del sistema Suite de pruebas de regresión

Tabla 3.1 Comparación de metodologías de desarrollo de software.
Fuente: *Elaboración propia*

De estas cuatro metodologías se seleccionó SCRUM debido a la simplicidad de su aplicación, además que no detallaba como elaborar ciertas tareas, brindando libertad a los participantes de crear su propio método de trabajo para las áreas de diseño, codificación y pruebas. Se enfoca principalmente en la obtención de resultados concentrándose más en el desarrollo del software que en la documentación, mas no ata a los desarrolladores con tareas específicas en las áreas ya mencionadas.

3.1.1 SCRUM

Es una metodología de desarrollo de software ágil, que se enfoca en el trabajo en equipo y consecución de objetivos en el menor tiempo posible.

Se realizan entregas parciales y regulares del resultado final del proyecto, priorizadas por el beneficio que aportan a receptor del proyecto.

Los proyectos que utilizan esta metodología son especialmente aquellos que presentan entornos complejos, en donde los objetivos son difusos o cambiantes y se requieren resultados rápidamente.

3.1.1.1 Actores de SCRUM

- Product Owner, es quien se relaciona con el User y obtiene los requerimientos del proyecto, organizándolos por prioridad y así generar el Product Backlog.
- Scrum Master, es quien se encarga de eliminar los obstáculos que se le presente al Scrum Team para la consecución de los objetivos del Sprint, además regula que se cumplan con las reglas de Scrum. No es el líder de proyecto.
- Scrum Team, es el equipo de desarrollo encargado de cumplir con los objetivos pactados en el Sprint Backlog.
- Users/Clients, son los que reciben el proyecto, y observando los avances colaboran con ideas y requerimientos.

3.1.1.2 Actividades dentro de SCRUM

Como ya se mencionó SCRUM se basa en iteraciones (Sprints), de tal forma el avance del proyecto es de forma incremental.

Existen tres actividades definidas dentro de cada iteración:

- Planificación de la Iteración (Scrum Planning Meeting)

Reunión con una duración de 8 horas máximo divida en 2 partes:

- Selección de requisitos (4 horas máximo): a partir de una lista de requisitos priorizada del producto (Product Backlog), el equipo de desarrollo (Scrum Team) selecciona los requisitos con mayor prioridad, que se comprometen a cumplir al finalizar el Sprint, y además realizan preguntas al cliente sobre sus dudas.
- Planificación de la iteración (4 horas máximo): se elabora la lista de requerimientos a cumplir en la iteración (Sprint Backlog) con los requerimientos seleccionados previamente. Se estima el esfuerzo de desarrollo de manera conjunta y se auto asignan las tareas.

- Ejecución de la iteración (Daily Scrum Meeting)

  Diariamente dentro al ejecución de la iteración (Sprint) se realiza una reunión de 30 minutos como máximo entre el Scrum Master y el Scrum Team en donde se elaboran las siguientes preguntas a cada integrante del equipo desarrollador:

  - ¿Qué hiciste ayer?
  - ¿Qué piensas hacer hoy?
  - ¿Qué ayuda necesitas?

  Finalmente el Scrum Master es el encargado de eliminar a lo largo del día los obstáculos o problemas que impidan el desarrollo normal del proyecto.

- Inspección y adaptación (Sprint Review y Sprint Retrospective)

  Ya pasado el tiempo acordado para el Sprint se realizan un par de reuniones:

  - Sprint Review: se reúnen el Scrum Team, Scrum Master, Product Owner y cualquier interesado, en ella se muestra el resultado de la iteración realizada. Es aquí que los participantes pueden dar a conocer sus opiniones, posibles mejoras, etc.
  - Sprint Retrospective: participan el Scrum Team, Scrum Master y Product Owner. Se analiza que se puede mejorar a partir de lo aprendido en la iteración pasada, y así mejorar el modo de

trabajo.

Finalizado esta actividad, se reinicia nuevamente el ciclo del SCRUM.

3.1.1.3 Artefactos de SCRUM

- **Product backlog**

  El product backlog es un documento de alto nivel para todo el proyecto. Contiene descripciones genéricas de todos los requerimientos, funcionalidades deseables, etc. priorizadas según su valor para el negocio (business value). Es el qué va a ser construido. Es abierto y cualquiera puede modificarlo. Contiene estimaciones grosso modo, tanto del valor para el negocio, como del esfuerzo de desarrollo requerido. Esta estimación ayuda al product owner a ajustar la línea temporal y, de manera limitada, la prioridad de las diferentes tareas. Por ejemplo, si dos características tienen el mismo valor de negocio la que requiera menos tiempo de desarrollo tendrá probablemente más prioridad, debido a que su ROI será más alto.

- **Sprint backlog**

  El sprint backlog es un documento detallado donde se describe el cómo el equipo va a implementar los requisitos durante el siguiente sprint. Las tareas se dividen en horas con ninguna tarea de duración superior a 16 horas. Si una tarea es mayor de 16 horas, deberá ser rota en mayor detalle. Las tareas en el sprint backlog nunca son asignadas, son tomadas por los miembros del equipo del modo que les parezca oportuno.

- **Burn down**

  La burn down chart es una gráfica mostrada públicamente que mide la cantidad de requisitos en el Backlog del proyecto pendientes al comienzo de cada Sprint. Dibujando una línea que conecte los puntos de todos los Sprints completados, podremos ver el progreso del proyecto. Lo normal es que esta línea sea descendente (en casos en que todo va bien en el sentido de que los requisitos están bien definidos desde el principio y no varían nunca) hasta llegar al eje horizontal, momento en el cual el proyecto se ha terminado (no hay más requisitos pendientes de ser completados en el Backlog). Si durante el proceso se añaden nuevos

requisitos la recta tendrá pendiente ascendente en determinados segmentos, y si se modifican algunos requisitos la pendiente variará o incluso valdrá cero en algunos tramos.

3.2 ELECCIÓN DE PLATAFORMA

El sistema que desarrollaremos estará orientado al desarrollo de una aplicación web para que cada usuario por medio de una conexión a internet pueda acceder a su cuenta desde cualquier parte necesitando tener instalado en su computador un navegador web IE o Firefox.

Para escoger la plataforma web se ha considerado los siguientes criterios con su respectivo porcentaje de ponderación.

- **Costo (35%):** Se analiza el costo monetario de las herramientas que usa en la plataforma tanto para su desarrollo como para su despliegue.
- **Seguridad (27%):** Se analiza que tan segura y resistente a ataques sería la aplicación desarrollada en la plataforma escogida.
- **Herramientas y frameworks (15%):** Se analiza la variedad de herramientas y frameworks que ofrece la plataforma para agilizar el desarrollo de la aplicación.
- **Robustez (15%):** Se analiza las herramientas y servicios que provee la plataforma para que la aplicación desarrollada no cause errores críticos y despliegue mensajes de error apropiados en situaciones no contempladas en los requerimientos.
- **Velocidad de respuesta (13%):** Se analiza que tan rápido responde la aplicación, desplegada en una determinada plataforma, a peticiones del usuario web.

	Costo (30%)	**Seguridad (27%)**	**Herramientas y frameworks (15%)**	**Robustez (15%)**	**Velocidad de respuesta (13 %)**	**Total**
Java	5	5	4	4	3	4.44
ASP.Net	2	5	5	4	4	3.82
PHP	5	3	3	3	4	3.73

Tabla 3.2 Comparación de plataformas para desarrollo de software
Fuente: *Elaboración propia*

Por lo tanto la plataforma escogida es java debido al poco costo que es necesario para conseguir acceso a las distintas herramientas y proceder al despliegue de la aplicación, la seguridad que nos ofrece la máquina virtual, la variedad de frameworks que permite agilizar el desarrollo del proyecto, la robustez que nos dan los servidores para desplegar aplicaciones java y por último la velocidad de respuesta a la petición web de un usuario.

3.3 ANÁLISIS

3.3.1 Casos de uso

3.3.1.1 Módulo de documentos

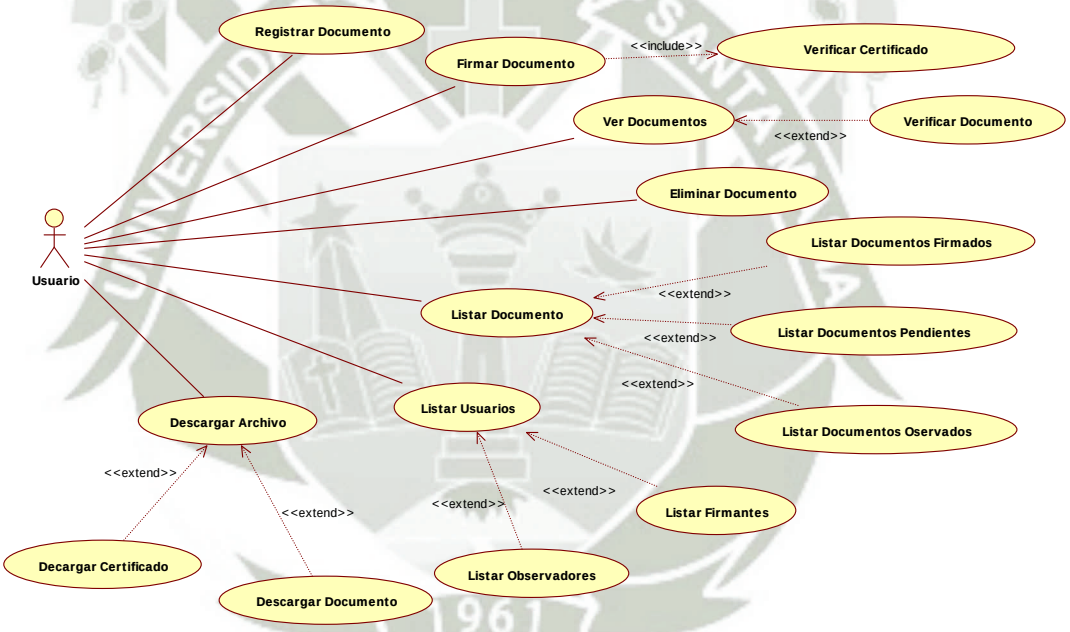


Figura 3.1 Diagrama de casos de uso – módulo de documentos
Fuente: *Elaboración propia.*

CU0001	**Registrar documento**
Flujo de Eventos	**Flujo principal** • El usuario selecciona el documento a registrar, selecciona los firmantes desde su lista de contactos, selecciona

	observadores desde su lista de contactos, finalmente ingresa una descripción y un mensaje para los firmantes y para los observadores. • Una vez ingresados los datos el usuario presiona el botón Guardar, luego de esto un mail es enviado a los firmantes y contactos junto con sus respectivos mensajes. • Si el documento ingresado es un PDF este es modificado y se colocan los respectivos contenedores de firma para cada uno de los firmantes asignados, estos contenedores serán creados de acuerdo al estándar de firmas digitales de Acrobat Reader.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	Se ha insertado el registro del documento en la BD.

CU0002	**Firmar documento**
Flujo de Eventos	**Flujo principal** • El usuario selecciona el archivo (.p12, .pfx) que contiene su clave privada, el usuario ingresa la contraseña de su clave privada y presiona el botón Aceptar. • Si la clave privada o la contraseña son inválidos ir a **FA1**, caso contrario se procede a verificar si el certificado de la clave privada no ha sido revocado y no ha expirado. • Si el certificado ha sido revocado o ha expirado ir a **FA2**, caso contrario se procede a firmar digitalmente el documento. • Si el documento a firmar es un PDF ir a **FA3**, caso contrario se procede a crear la firma del documento en el sistema y aparece un mensaje indicando que el documento ha sido firmado satisfactoriamente.

	FA1 • Si la clave privada o la contraseña son inválidos se cancela la firma del documento y se muestra el mensaje indicando el error o problema correspondiente. **FA2** • Si el certificado ha sido revocado o ha expirado se cancela la firma del documento y se muestra el mensaje correspondiente a la revocación o expiración del documento. **FA3** • Si el documento a firmar es un PDF se procede a crear la firma del documento en el sistema y adicionalmente se modifica el documento original para incrustar la firma en el contenedor de firma previamente creado de acuerdo a los estándares de firmas digitales de Acrobat Reader, finalmente se muestra el mensaje indicando el error o problema correspondiente.
Precondiciones	El usuario debe ser uno de los firmantes del documento escogido, el usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	Se ha insertado el registro de la firma del documento en la BD.

CU0003	**Ver documento**
Flujo de Eventos	**Flujo principal** • Se muestra, nombre, descripción, fecha de creación del documento. • Se generara links para que el usuario pueda ver los observadores y los firmantes. • Se muestran los botones Descargar, Reportar e Historial. • Si el usuario que revisa el documento es un firmante aparecerá al botón Firmar.

	• Si el documento ha sido firmado por algún usuario se procederá a realizar la verificación por cada firma realizada. o Si la verificación de firma es incorrecta ir a **FA1**, caso contrario se muestra una región de color "verde", dentro de esta región aparecen datos del usuario: Foto Registrada del Usuario, Nombre del Usuario, Email, Fecha en la que se realizó la firma, Imagen de la rúbrica; luego aparecen datos obtenidos del certificado: Nombre del propietario del certificado: Autoridad de Certificación, Fecha de Expedición del certificado, Fecha de Expiración del Certificado, Un enlace para Descargar el Certificado Digital y finalmente el estado de la verificación el cual debe decir "Verificación Correcta". **FA1** o Si la verificación es incorrecta se muestra una región de color "rojo", dentro de esta región aparecen datos del usuario: Foto Registrada del Usuario, Nombre del Usuario, Email, Fecha en la que se realizó la firma, Imagen de la rúbrica; luego se muestra el Estado de Verificación el cual debe describir el error ocurrido al momento de verificar la firma del documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0004	**Eliminar documento**
Flujo de Eventos	**Flujo principal** • El usuario presiona el botón eliminar del documento

	seleccionado de un listado. • Se muestra una ventana emergente, preguntando al usuario si desea eliminar el documento, también se muestran los botones Aceptar y Cancelar. • Si el usuario presiona Aceptar se procede con la eliminación del documento. • Si el usuario presiona Cancelar no se elimina el documento y se cierra la ventana emergente.
Precondiciones	El usuario debe ser la persona que registró el documento, el usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	Se elimina el registro en BD del documento eliminado.

CU0005	**Listar documentos firmados**
Flujo de Eventos	**Flujo principal** • Se listan todos los documentos que han sido firmados por el usuario, ordenados por la fecha de firma en forma descendente. • En el listado de documentos se muestran los siguientes campos: Nombre del documento, fecha de firma, Link para mostrar los personas que firmaron el documento, Link para mostrar los observadores asignados al documento, finalmente un ícono el cual permitirá Ingresar al Caso de Uso Ver Documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0006	**Listar documentos pendientes**
Flujo de Eventos	**Flujo principal** • Se listan todos los documentos que requieren la firma del

	usuario, ordenados por la fecha de creación del documento, adicionalmente al listado se muestra el botón Firmar el cual permitirá Firmar los Documentos seleccionados del listado. • En el listado de documentos se muestran los siguientes campos: Checkbox para seleccionar el documento, nombre del documento, fecha de firma, Link para mostrar los personas que firmaron el documento, Link para mostrar los observadores asignados al documento, finalmente un ícono el cual permitirá Ingresar al Caso de Uso Ver Documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0007	**Listar documentos observados**
Flujo de Eventos	**Flujo principal** • Se listan todos los documentos donde el usuario ha sido asignado como observador, ordenados por la fecha de creación del documento. • En el listado de documentos se muestran los siguientes campos: Nombre del documento, fecha de firma, Link para mostrar los personas que firmaron el documento, Link para mostrar los observadores asignados al documento, finalmente un ícono el cual permitirá Ingresar al Caso de Uso Ver Documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0008	**Listar firmantes**
Flujo de Eventos	**Flujo principal** • El link para mostrar Firmantes mostrará: “Número de

	Personas que Firmaron" / "Número Total de Firmantes". • Cuando el usuario presione el link descrito se mostrará una ventana emergente donde se mostrarán: Nombre de los Firmantes del documento, Estado de la Firma respectiva a cada firmante dicho estado puede ser Pendiente o Firmado; Finalmente en la parte inferior se tendrá un Progress Bar donde se indicará el porcentaje de avance en la firma del documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0009	**Listar observadores**
Flujo de Eventos	**Flujo principal** • El link para mostrar Observadores mostrará el Número Total de Observadores. • Cuando el usuario presione el link descrito se mostrará una ventana emergente donde se mostrarán los Nombres de los Observadores asignados al documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0010	Descargar documento
Flujo de Eventos	**Flujo principal** • El Usuario presiona el botón Descargar, si el documento es un PDF se descargará un documento modificado que tendrá contenedores de firma de acuerdo a los estándares de firma digital de Acrobat Reader; en caso que el documento "no sea PDF" se descargará la versión original del documento.
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

CU0011	Descargar certificado
Flujo de Eventos	**Flujo principal** • El usuario presiona el link Descargar dentro de la zona de verificación de Firmas realizadas, seguidamente se procede con la descarga del certificado Digital del Firmante (.cer).
Precondiciones	El usuario debe estar logueado, el sistema ha iniciado una conexión segura SSL.
Poscondiciones	

3.3.1.2 Módulo de conexión SSL

Figura 3.2 Diagrama de casos de uso – módulo de conexión SSL.
Fuente: *Elaboración propia.*

CU0012	Iniciar conexión segura SSL

Flujo de Eventos	**Flujo principal** • El sistema inicia una conexión segura utilizando el protocolo SSL con un certificado previamente creado.
Precondiciones	
Poscondiciones	

3.3.1.3 Módulo de usuarios

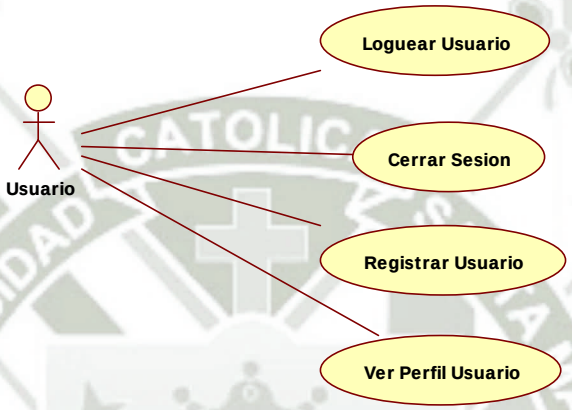


Figura 3.3 **Diagrama de casos de uso – módulo de usuarios**
Fuente: *Elaboración propia.*

CU0013	**Loguear usuario**
Flujo de Eventos	**Flujo principal** • El usuario ingresa su nombre de usuario (correo electrónico registrado) y su contraseña. • Si el correo electrónico o la contraseña son incorrectos ir a **FA1**, caso contrario se procede a ingresar al sistema. **FA1** • Si el correo electrónico o la contraseña son incorrectos se cancela el ingreso del usuario y se muestra el mensaje indicando el error o problema correspondiente.
Precondiciones	El usuario se debe encontrar previamente registrado en la BD del sistema a través de la interfaz de registro de usuarios nuevos.
Poscondiciones	Se crea una sesión en el sistema con el usuario logueado.

CU0014	**Registrar usuario**
Flujo de Eventos	**Flujo principal** • El usuario presiona el botón Registrar de la interfaz de logueo del sistema. • El usuario ingresa los datos requeridos por la interfaz de registro de usuario nuevo: o Nombres (Requerido). o Apellidos (Requerido). o Correo Electrónico (Requerido). o Contraseña (Requerido). o DNI (Opcional). o Nro. Teléfono (Opcional). o Img. Usuario (Requerido). o Img. Firma (Requerido). • El usuario presiona el botón Guardar de la interfaz. • Si alguno de los datos ingresados son incorrectos ir a **FA1**, caso contrario se procede al registro del usuario en la base de datos. **FA1** • Si alguno de los datos ingresados son incorrectos se cancela el registro del nuevo usuario y se muestra el mensaje indicando el error o problema correspondiente.
Precondiciones	
Poscondiciones	Se registra el nuevo usuario en la BD y se envía un mail de confirmación de registro de usuario.

CU0015	**Ver perfil usuario**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Mi Perfil en el menú de opciones de usuario.

	• Se presenta una interfaz con los siguientes datos registrados por el usuario: o Nombre completo. o Correo Electrónico. o DNI. o Nro. Teléfono. o Img. Usuario. o Img. Firma.
Precondiciones	
Poscondiciones	

CU0016	**Cerrar sesión**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Cerrar Sesión en el menú de opciones de usuario.
Precondiciones	El usuario debe encontrarse previamente logueado en el sistema.
Poscondiciones	El sistema termina la sesión abierta por el usuario.

3.3.1.4 Módulo de contactos

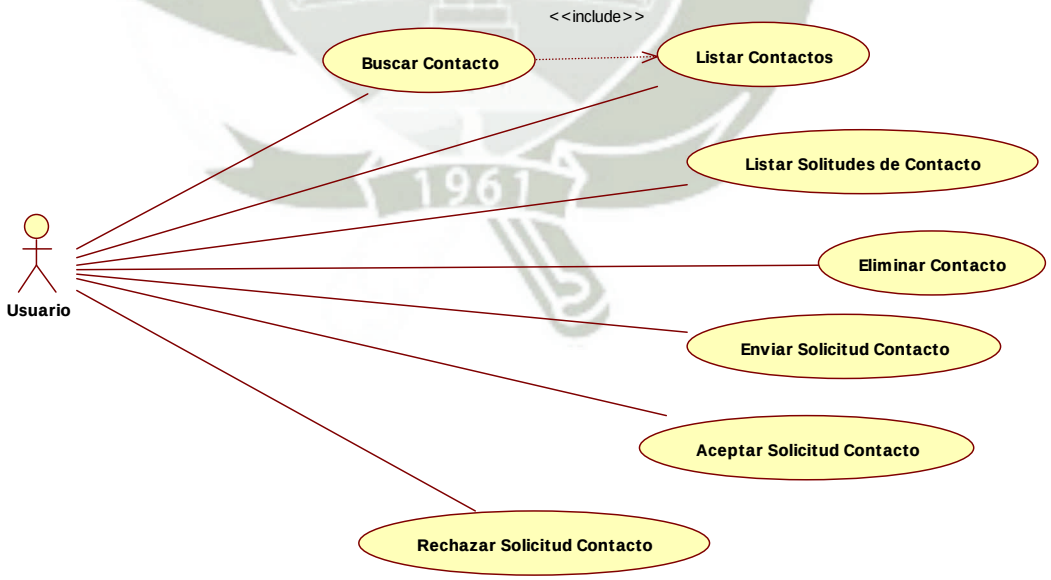


Figura 3.4 Diagrama de casos de uso – módulo contactos
Fuente: *Elaboración propia.*

CU0017	**Listar contactos**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Mis Contactos en el menú de opciones de Contactos. • El sistema le muestra al usuario una lista con los contactos registrados que posee en su cuenta.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	

CU0018	**Eliminar contacto**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Mis Contactos en el menú de opciones de Contactos. • El usuario presiona el ícono Eliminar (tacho) que se presenta al lado derecho de cada contacto. • El usuario debe confirmar esta acción presionando el botón Aceptar del mensaje emergente que se le presente.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	

CU0019	**Buscar contacto**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Añadir Contactos en el menú de opciones de Contactos. • Se le presenta al usuario una interfaz en la que debe de ingresar el nombre del contacto al que desea buscar, posteriormente debe presionar el botón Buscar. • Se le presenta al usuario una lista con los nombres de otros usuarios que cumplan con su búsqueda.

Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	

CU0020	**Enviar solicitud de contacto**
Flujo de Eventos	**Flujo principal** • El usuario realiza una búsqueda de contactos. • El usuario presiona el ícono Agregar (visto) que se encuentra al lado derecho del nombre del usuario encontrado. • El usuario confirma el envío de la solicitud de contacto presionando el botón Aceptar del mensaje emergente que se le presente.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	El sistema registra la solicitud de contacto en la BD.

CU0021	**Aceptar solicitud de contacto**
Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Mis Solicitudes en el menú de opciones de Contactos. • El sistema le muestra al usuario una lista con los nombres de otros usuarios que desean agregarlo a su lista de contactos. • El usuario acepta la solicitud presionando el ícono Aceptar (Visto) que se encuentra al lado derecho del nombre de usuario. • El usuario confirma la acción presionando el botón Aceptar del mensaje emergente que se le presente.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	El sistema registra el nuevo contacto en la BD.

CU0022	**Rechazar solicitud de contacto**

Flujo de Eventos	**Flujo principal** • El usuario presiona el enlace Mis Solicitudes en el menú de opciones de Contactos. • El sistema le muestra al usuario una lista con los nombres de otros usuarios que desean agregarlo a su lista de contactos. • El usuario rechaza la solicitud presionando el ícono Rechazar (Aspa) que se encuentra al lado derecho del nombre de usuario. • El usuario confirma la acción presionando el botón Aceptar del mensaje emergente que se le presente.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	El sistema elimina la solicitud de contacto de la BD.

3.3.1.5 Módulo de notificaciones

Figura 3.5 Diagrama de casos de uso – módulo de notificaciones
Fuente: *Elaboración propia.*

CU0023	**Ver notificaciones de documento**
Flujo de Eventos	**Flujo principal** • El usuario ingresa a la opción Ver de alguno de los documentos.

	• El usuario presiona el botón Historial que se presenta al lado derecho de la descripción del documento. • El sistema le muestra al usuario una lista con las notificaciones generadas por el documento y las fechas de las mismas, estas pueden ser: o Registro del documento. o Firma del documento. o Reporte del documento.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	

CU0024	**Registrar notificación**
Flujo de Eventos	**Flujo principal** • El usuario ingresa a la opción Ver de alguno de los documentos. • El usuario presiona el botón Reportar que se presenta al lado derecho de la descripción del documento. • El usuario ingresa en la interfaz mostrada, el motivo del reporte y finaliza presionando el botón Aceptar.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	El reporte en registrado como parte del historial del documento en la BD.

CU0025	**Ver notificaciones de usuario**
Flujo de Eventos	**Flujo principal** • El usuario presiona en el enlace Ver más de la lista de notificaciones que se le presenta luego de ingresar al sistema o de presionar en el logo del mismo. • El sistema le muestra al usuario una lista de las notificaciones recientes generadas en los distintos documentos que se encuentran a su cargo, estas

	notificaciones pueden ser de los siguientes tipos: o Registro del documento. o Firma del documento. o Reporte del documento.
Precondiciones	El usuario se debe de encontrar logueado en el sistema.
Poscondiciones	

3.4 DISEÑO

3.4.1 Ilustración del funcionamiento de la aplicación

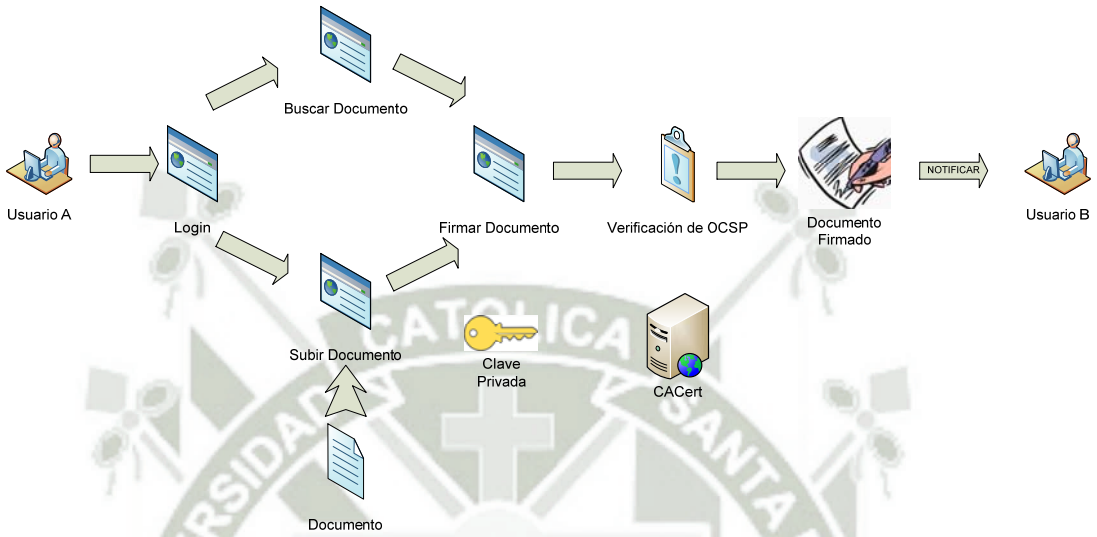


Figura 3.6 Diagrama de firma de documento.
Fuente: *Elaboración propia.*

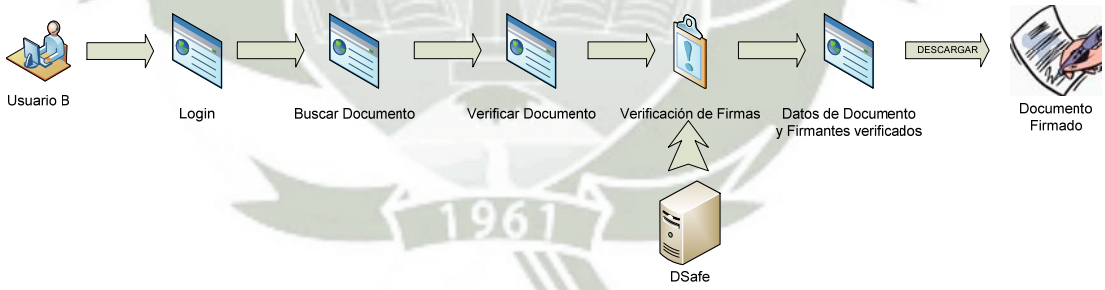


Figura 3.7 Diagrama de verificación de documento firmado.
Fuente: *Elaboración propia.*

3.4.2 Diagramas de flujos

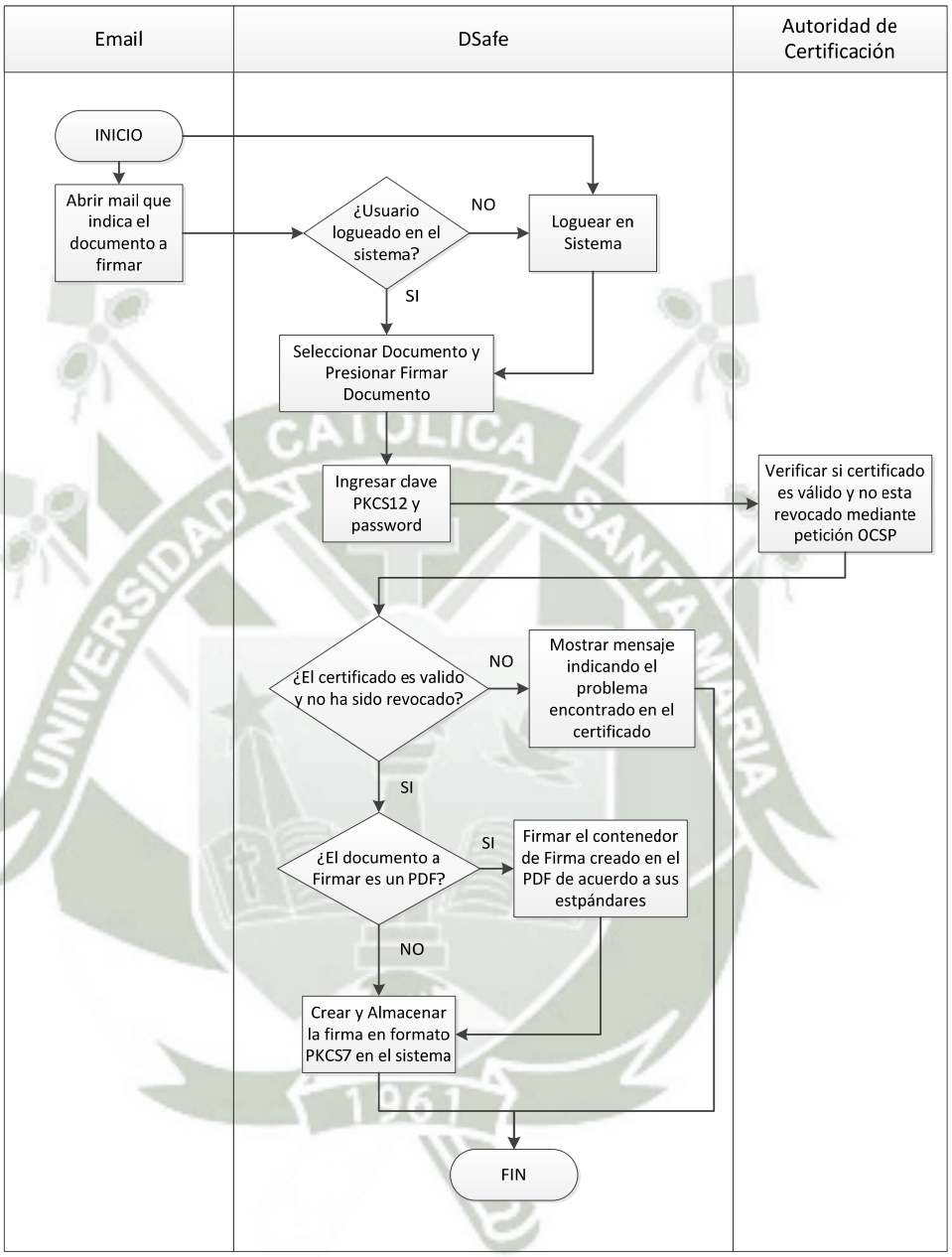


Figura 3.8 Diagrama de flujo para firmar un documento digital.
Fuente: *Elaboración propia.*

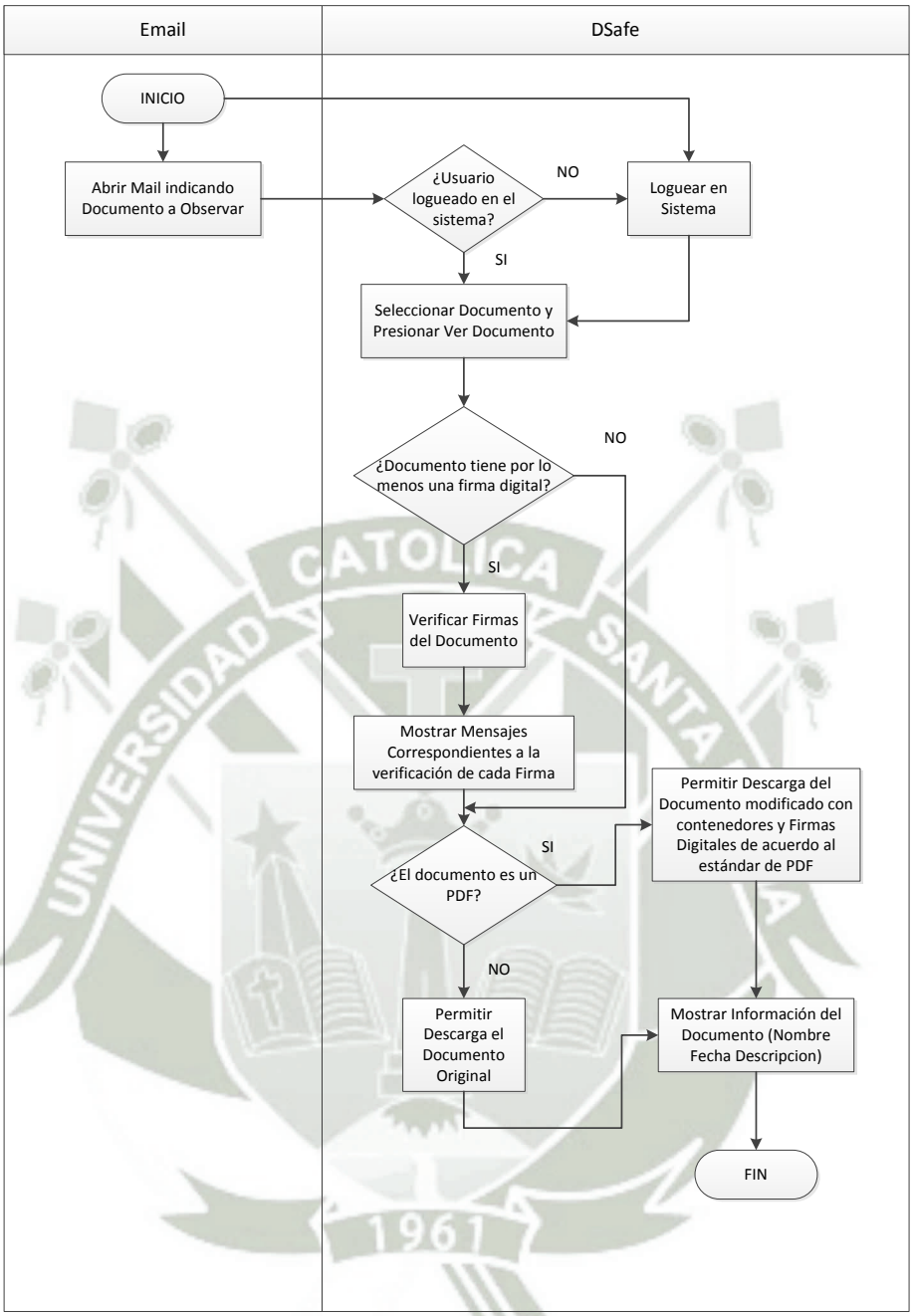


Figura 3.9 Diagrama de flujo para ver un documento.
Fuente: *Elaboración propia.*

3.4.3 Diagramas de clases

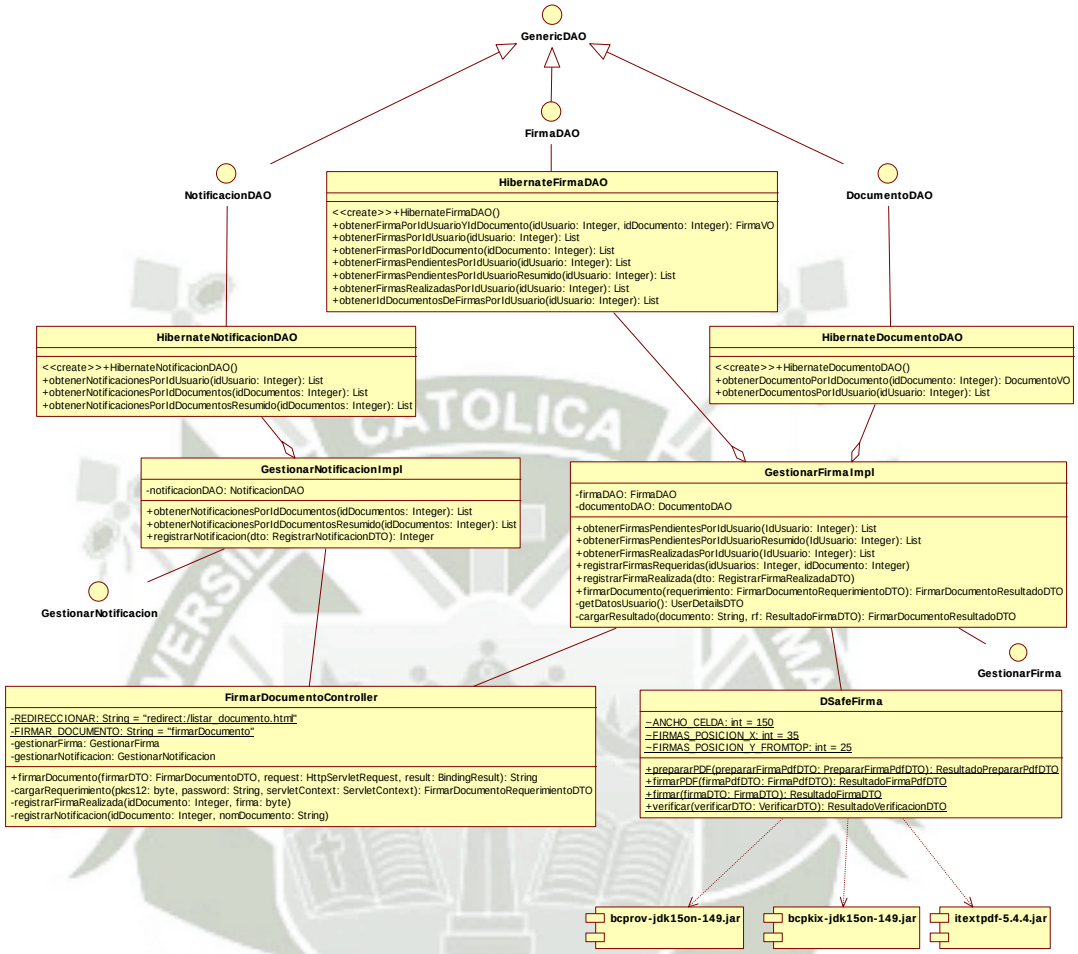


Figura 3.10 Diagrama de clases CU_FirmarDocumento.
Fuente: *Elaboración propia.*

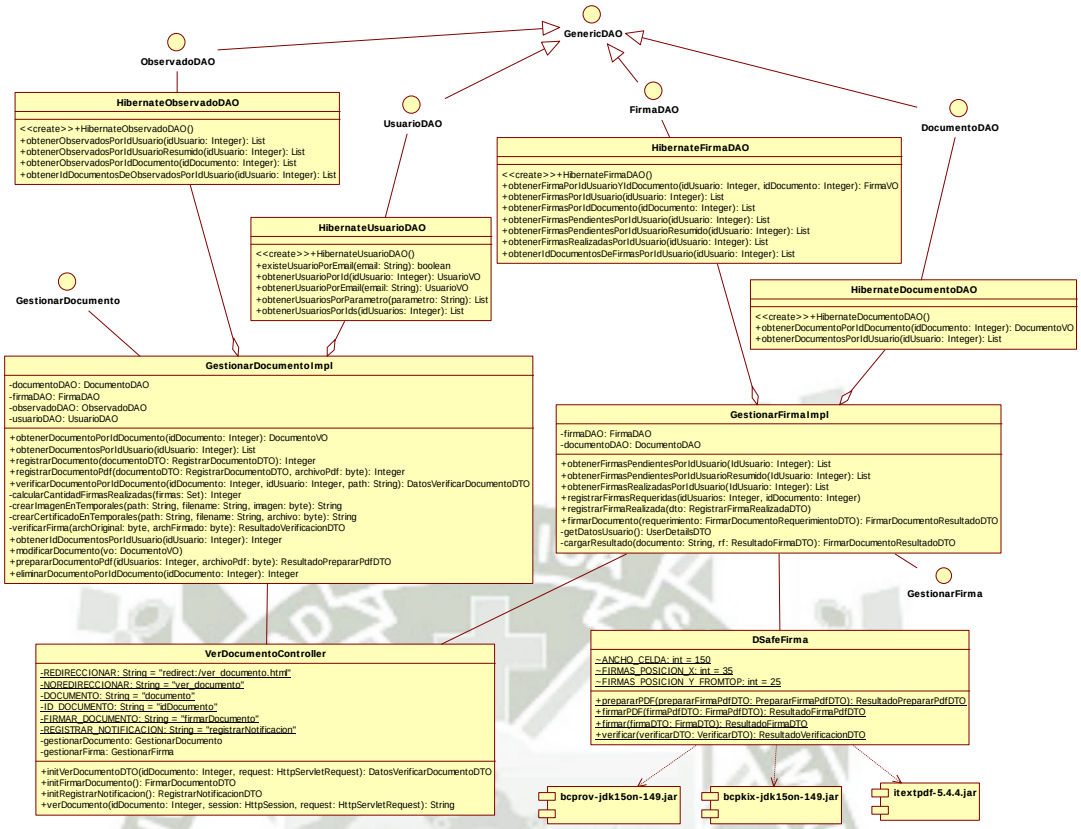


Figura 3.11 Diagrama de clases CU_VerDocumento.
Fuente: *Elaboración propia.*

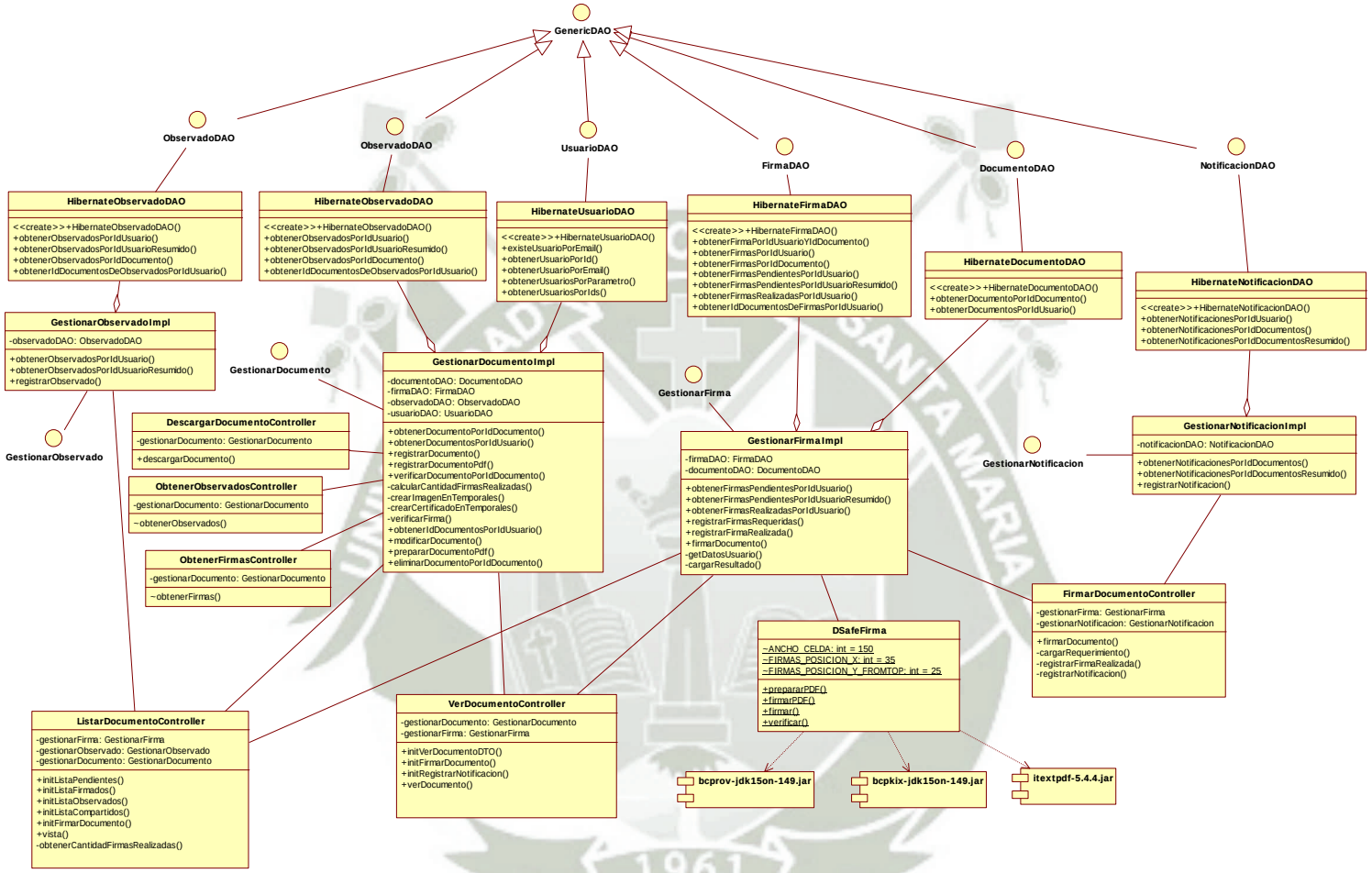


Figura 3.12 Diagrama de clases módulo de documentos.

Fuente: *Elaboración propia.*

3.4.4 Diagramas de secuencia

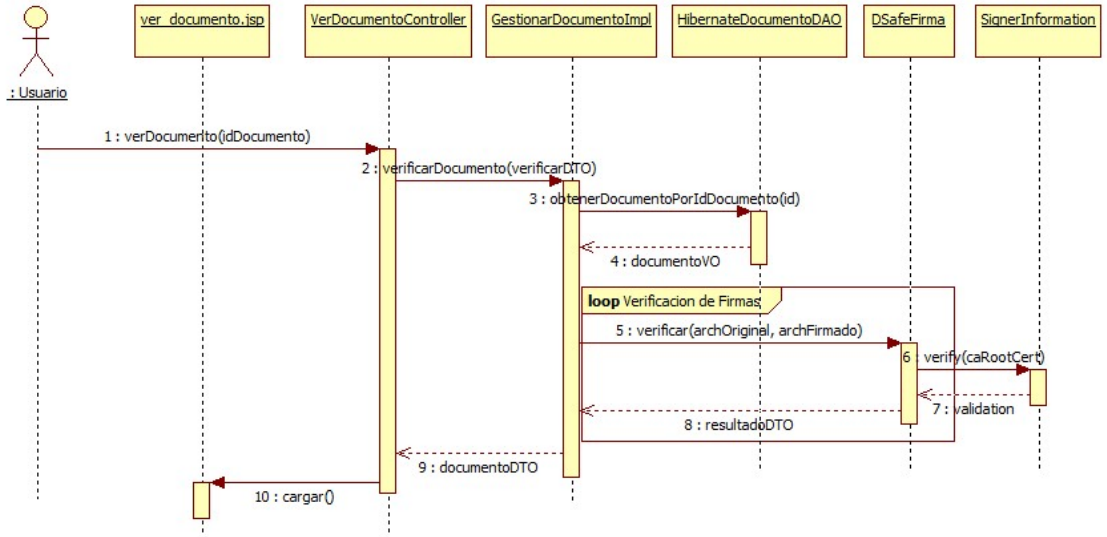


Figura 3.13 Diagrama de secuencia ver documento.
Fuente: *Elaboración propia.*

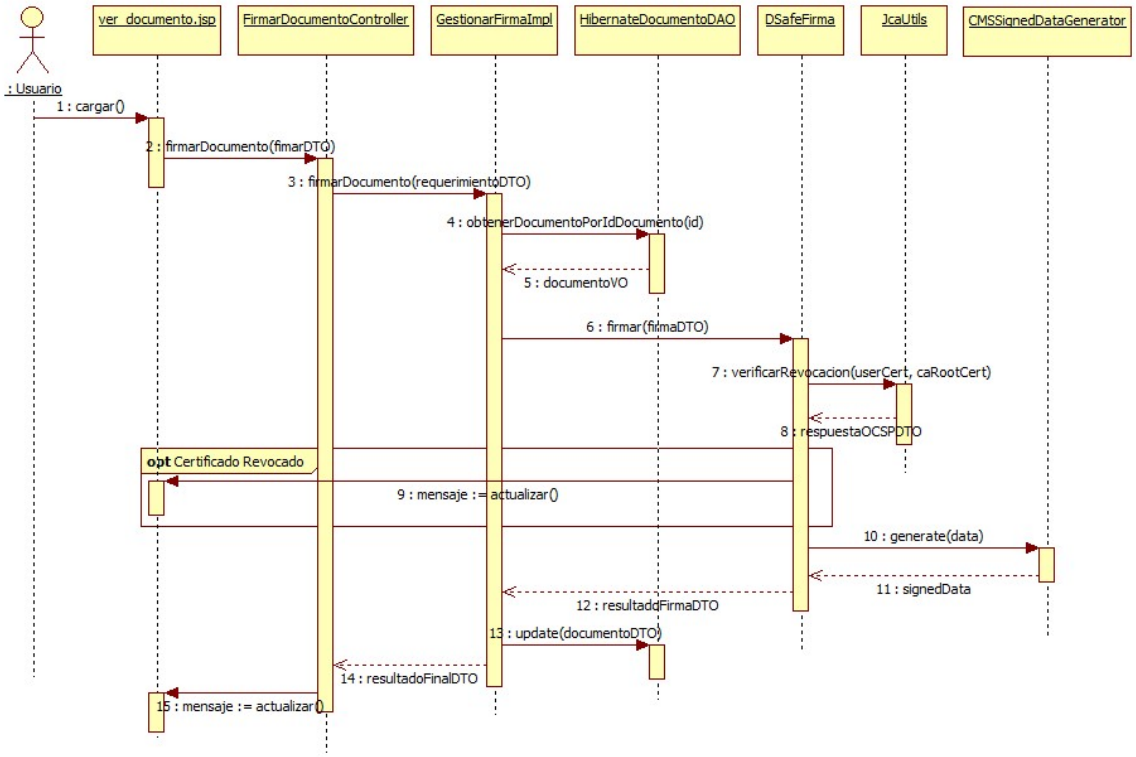


Figura 3.14 Diagrama de secuencia firmar documento.
Fuente: *Elaboración propia.*

3.4.5 Diagrama entidad-relación

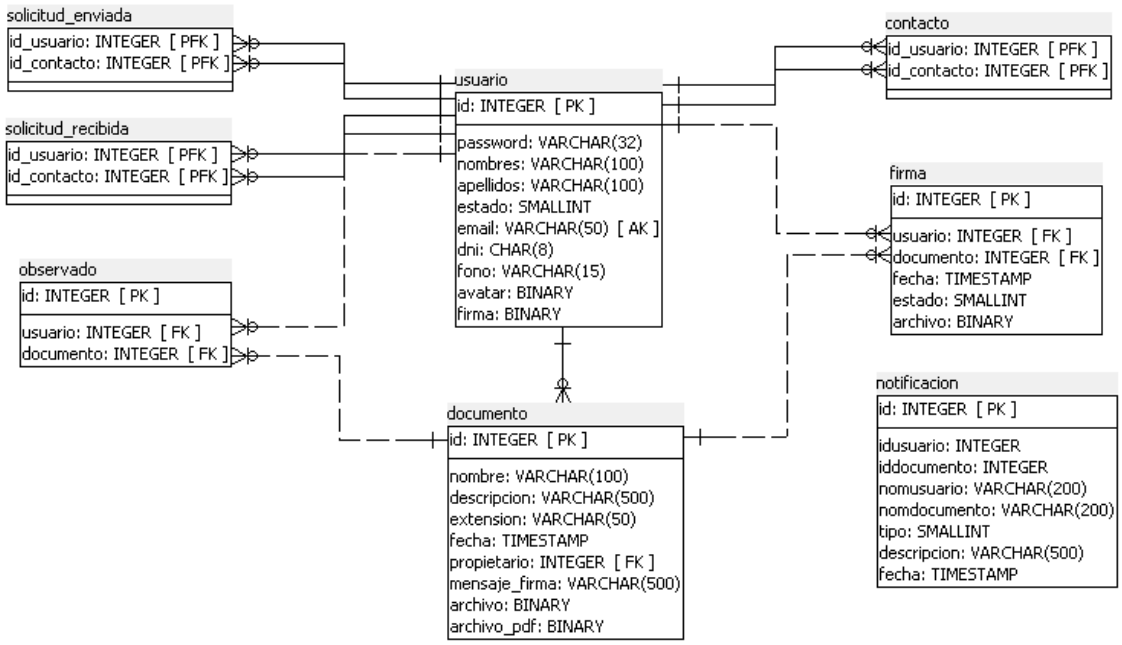


Figura 3.15 Diagrama entidad-relación.
Fuente: *Elaboración propia.*

3.4.6 Diagrama de componentes

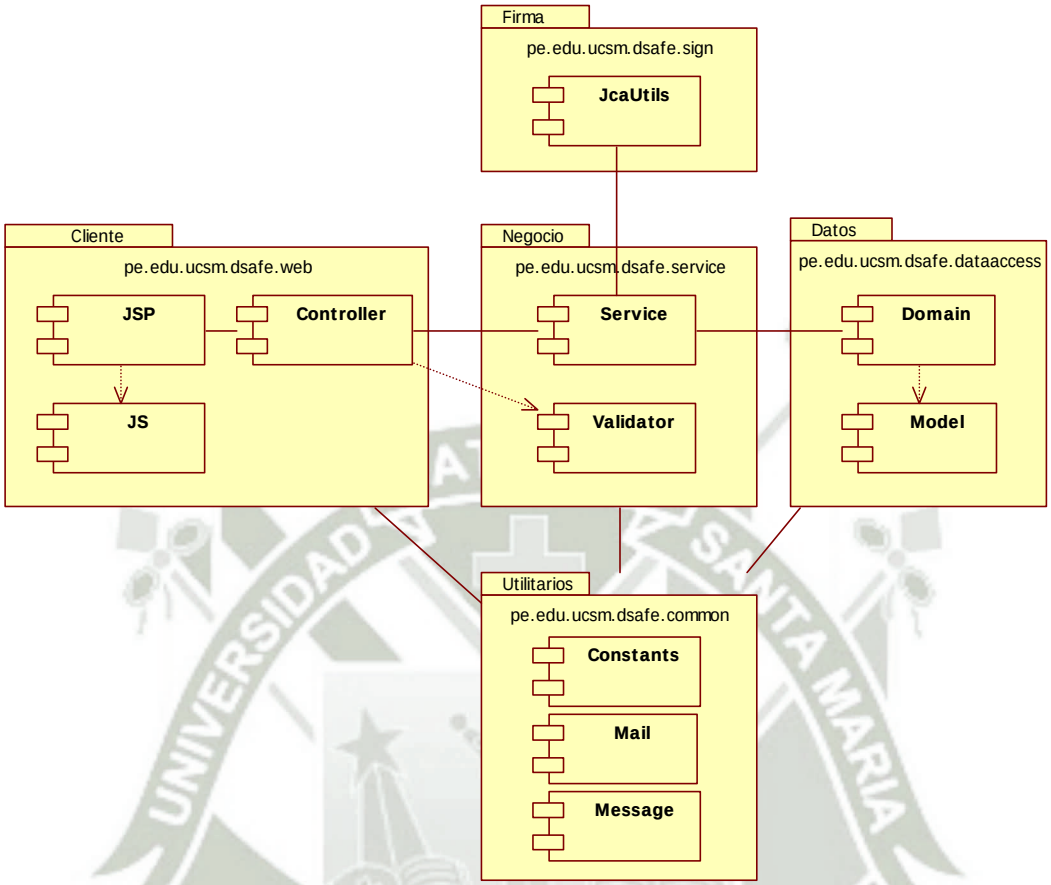


Figura 3.16 Diagrama de componentes.
Fuente: *Elaboración propia.*

3.4.7 Diagrama de despliegue

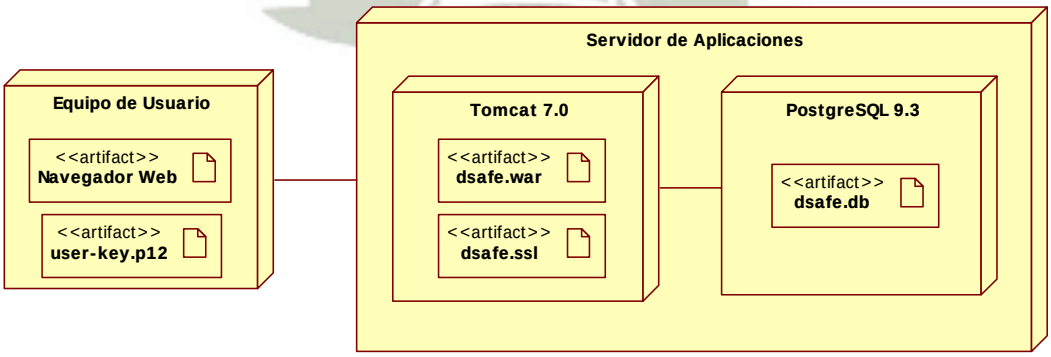


Figura 3.17 Diagrama de despliegue.
Fuente: *Elaboración propia.*

CAPÍTULO 4: PRUEBAS Y RESULTADOS

4.1 CÁLCULO Y MEDICIÓN DE LOS INDICADORES

Las mediciones se realizaron en un servidor con las siguientes características:

- **Procesador:** Intel Core i7-2630QM CPU 2.00 GHz (cuatro núcleos).
- **Capacidad de Memoria RAM:** 8 GB.
- **Sistema Operativo**: Windows 7 Home Premium
- **Disco Duro disponible**: 100 GB de disco duro disponible

4.1.1 Integridad de implementación funcional

Evaluación realizada del cumplimiento de requerimientos del proyecto.

	REQUERIMIENTO	ESTADO
CU0001	Registrar Documento	Funcional
CU0002	Firmar Documento	Funcional
CU0003	Ver Documento	Funcional
CU0004	Eliminar Documento	Funcional
CU0005	Listar Documentos Firmados	Funcional
CU0006	Listar Documentos Pendientes	Funcional
CU0007	Listar Documentos Observados	Funcional
CU0008	Listar Firmantes	Funcional
CU0009	Listar Observadores	Funcional
CU0010	Descargar Documento	Funcional
CU0011	Descargar Certificado	Funcional
CU0012	Iniciar Conexión Segura SSL	Funcional
CU0013	Loguear Usuario	Funcional
CU0014	Cerrar Sesión	Funcional
CU0015	Registrar Usuario	Funcional
CU0016	Ver Perfil Usuario	Funcional
CU0017	Buscar Contacto	Funcional
CU0018	Listar Contactos	Funcional
CU0019	Listar Solicitudes de Contacto	Funcional
CU0020	Eliminar Contacto	Funcional
CU0021	Enviar Solicitud Contacto	Funcional
CU0022	Aceptar Solicitud Contacto	Funcional
CU0023	Rechazar Solicitud Contacto	Funcional
CU0024	Ver Notificaciones de Documento	Funcional
CU0025	Ver Notificaciones Dirigidas al Usuario	Funcional

CU0026	Registrar Notificación	Funcional

Tabla 4.1 Requerimientos del sistema cumplidos
Fuente: Elaboración propia

Cálculo de métrica

A = Número de requerimientos faltantes detectados en la evaluación.
B = Número total de requerimientos.
X = 1 – A/B
X = 1 – 0/26
X = 1

Interpretación

El valor de la métrica es 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.2 Exactitud de cálculo

Número de errores encontrados en las operaciones de cálculo numérico del sistema.

Cálculo Numérico Evaluado	Número de Errores Encontrados
Número de Documentos Pendientes	0
Número de Documentos Firmados	0
Número de Documentos Observados	0
Número de Firmantes	0
Número de Observadores	0

Tabla 4.2 Errores de cálculo numérico del sistema
Fuente: Elaboración propia

Cálculo de métrica:

T: Tiempo de operación (meses): 1
A: Número de cálculos inexactos encontrados por los usuarios: 0
X: Exactitud de cálculo (lo más cercano a 0,0 es los mejor)
Xmax : Máximo de errores tolerables en 1 mes
Vm: Valor calculado de la métrica

X = A/T
X = 0/1
X = 0

Vm = 1 – X / Xmax
Vm = 1 – 0/1
Vm = 1

Interpretación:

El valor de la métrica es de 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.3 Intercambio de datos

Se realizaron peticiones OCSP usando 10 hilos concurrentes y 10 pruebas por cada hilo.

Cálculo de métrica:

A: Número de Errores encontrados en Peticiones OCSP
B: Número Total Accesos Concurrentes a Peticiones OCSP
C: Número Total de Pruebas por cada Acceso Concurrente a Peticiones OCSP

X = 1 - A/B*C
X = 1 - 0/10*10
X = 1

Interpretación:

El valor de la métrica es de 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.4 Control de acceso

Número de accesos no permitidos encontrados en el sistema.

Acciones no permitidas sin antes loguearse en el sistema	Accesos encontrados
Firmar documentos	0
Añadir documentos	0
Acceder al sistema directamente desde el enlace del mail de notificación	0
Aceptar solicitudes de contacto	0
Descargar documentos	0
Reportar observaciones	0
Verificar firmas de documento	0

Tabla 4.3 Accesos no permitidos encontrados en el sistema
Fuente: Elaboración propia

Cálculo de métrica:

A: Número de accesos en acciones no permitidas encontrados
B: Número total de acciones no permitidas
X = 1 - A/B
X = 1 - 0/7
X = 1

Interpretación:

El valor de la métrica es de 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.5 Control de acceso concurrente

Pruebas realizadas con JMeter con 20 usuarios concurrentes realizando el login en un mismo instante.

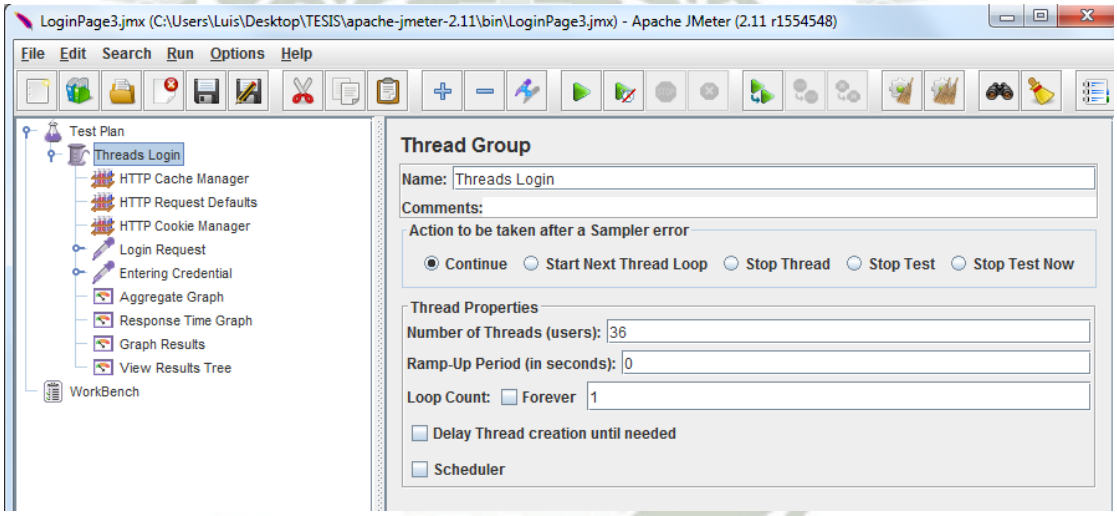


Figura 4.1 Parámetros de configuración en Jmeter
Fuente: Elaboración propia

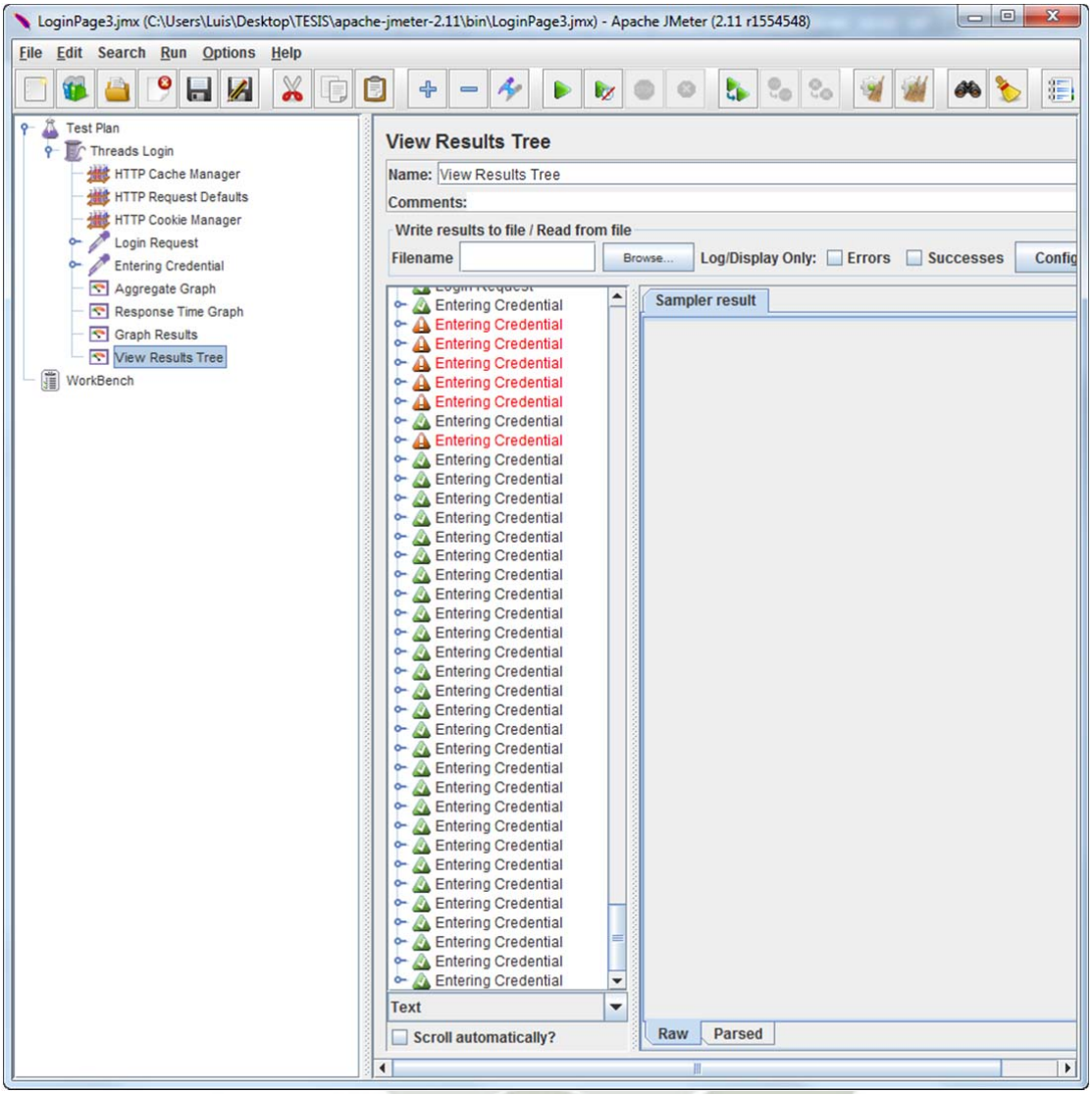


Figura 4.2 Resultados de pruebas de concurrencia en Jmeter
Fuente: Elaboración propia

Cálculo de métrica:

A: Número de errores encontrados al realizar login en el sistema.
B: Número total accesos concurrentes al login del sistema

X = 1 - A/B
X = 1 - 6/36
X = 0.83

Interpretación:

El valor de la métrica es de 0.83 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.6 Prevención de corrupción de los datos

Se realizaron pruebas de firma y verificación usando 10 hilos concurrentes y 10 pruebas sucesivas por cada hilo.

Cálculo de métrica:

A: Número de errores encontrados en procesos de firma de documentos
B: Número total de operaciones concurrentes en el proceso de firma de documentos
C: Número total de pruebas por cada operación concurrente

X = 1 - A/B*C
X = 1 - 0/10*10
X = 1

Interpretación:

El valor de la métrica es de 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.7 Densidad de fallas

Se recopilaron las fallas más resaltantes por iteración de SCRUM realizada en el proceso de programación del proyecto.

Sprint	Funcionalidades	Fallas detectadas	Relación
4	4	2	0.50
5	10	3	0.30
6	13	4	0.31
7	21	6	0.29
8	24	4	0.17
9	24	3	0.13

Tabla 4.4 Fallas detectadas en la aplicación por iteraciones de SCRUM
Fuente: Elaboración propia

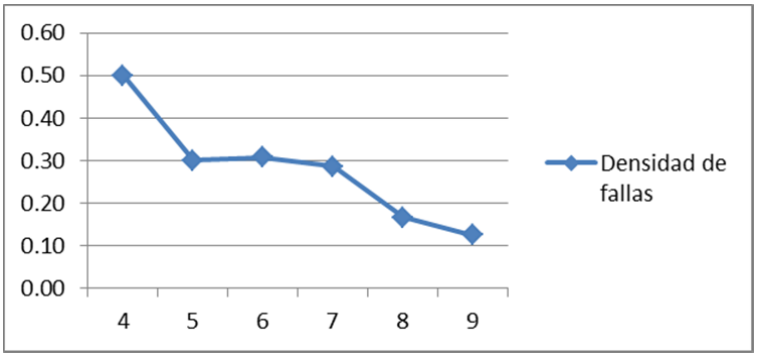


Figura 4.3 Gráfico de fallas en la aplicación por iteraciones de SCRUM
Fuente: Elaboración propia

Cálculo de la métrica:

P = Promedio entre las fallas detectadas y funcionalidades.
X = 1 – P
X = 1 – 0.13
X = 0.88

Interpretación:

El valor de la métrica es de 0.88 lo cual es favorable dado que cuanto más cercano sea a 1 es mejor.

4.1.8 Prevención de caídas

Se recopilaron las fallas que resultaron en caídas más resaltantes del sistema en cada iteración de scrum en el proceso de programación del proyecto.

Sprint	Fallas detectadas	Caídas generadas	Relación
4	2	1	0.50
5	3	2	0.67
6	4	2	0.50
7	6	2	0.33
8	4	1	0.25
9	3	0	0.00

Tabla 4.5 Fallas que resultaron en caídas del sistema por iteraciones de SCRUM
Fuente: Elaboración propia

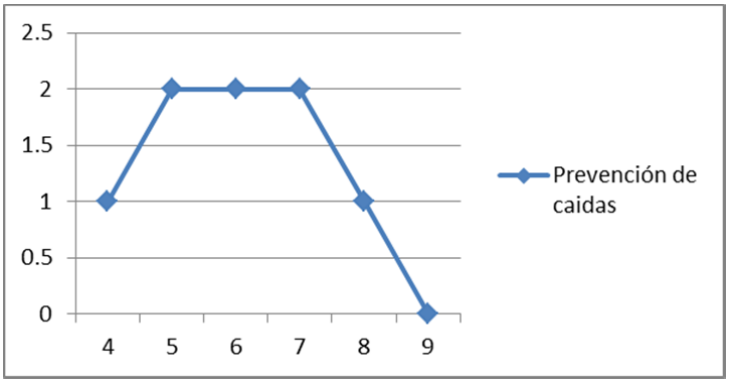


Figura 4.4 Gráfico de fallas que resultaron en caídas del sistema por iteraciones de SCRUM
Fuente: Elaboración propia

Cálculo de la métrica:

P = Promedio entre las caídas generadas y fallas detectadas.
X = 1 – P
X = 1 – 0.38
X = 0.63

Interpretación:

El valor de la métrica es de 0.63 lo cual es favorable dado que cuanto más cercano sea a 1 es mejor.

4.1.9 Prevención de operación incorrecta

Problemas encontrados en campos a validar en formularios de ingreso para prevenir errores críticos.

	Nro. de campos del formulario
Formulario de registro de usuarios	8

	No Acepta campo en blanco	Debe aceptar sólo campos numéricos	Campo debe tener formato determinado	Debe aceptar sólo una determinada longitud	Debe aceptar sólo archivos de un determinado formato
Nombre(s)	Correcto				
Apellidos(s)	Correcto				

Correo electrónico	Correcto		Correcto		
Contraseña	Correcto				
DNI		Correcto		Correcto	
Num. De Teléfono		Correcto			
Foto					Correcto
Firma					Correcto

Tabla 4.6 Problemas encontrados en campos del formulario de registro de usuarios

Fuente: Elaboración propia

	Nro. de campos del formulario
Formulario de registro de documento	6

	No Acepta campo en blanco	**Ingresa sólo Contactos del Usuario**
Archivo	Correcto	
Descripción	Correcto	
Firmantes	Correcto	Correcto
Mensaje Firmantes		
Observadores		Correcto
Mensaje Observadores		

Tabla 4.7 Problemas encontrados en campos del formulario de registro de documento

Fuente: Elaboración propia

Cálculo de métrica:

A: Número de operaciones incorrectas registradas
B: Número de casos de prevención de operaciones incorrectas contempladas

X = 1 - A/B
X = 1 - 0/15
X = 1

Interpretación:

El valor de la métrica es de 1 lo cual es muy favorable dado que es el puntaje máximo esperado en la métrica.

4.1.10 Disponibilidad

Datos recopilados durante el periodo de prueba.

Sprint	Tiempo de operación (días)	Tiempo de reparación (días)
9	5 d	1 d

Cálculo de métrica:

X: Promedio de la relación entre tiempo de reparación y tiempo de operación.
X = 0.83

Interpretación:

El valor de la métrica es de 0.83 lo cual es favorable dado que cuanto más cercano a 1 será mejor.

4.1.11 Tiempo de recuperación

Datos recopilados de cada caso de prueba realizado, tomado en segundos en cuanto tomaba reiniciar todo el sistema en su totalidad, para el peor de los casos, cuando se tenga que restaurar la base de datos.

CU	Equipo	PostgreSql	Tomcat	Total
1	25 s.	20 s.	15 s.	60 s.
2	30 s.	25 s.	16 s.	71 s.
3	26 s.	22 s.	18 s.	66 s.
4	22 s.	26 s.	14 s.	62 s.
5	35 s.	28 s.	13 s.	76 s.
6	34 s.	25 s.	20 s.	79 s.
7	25 s.	23 s.	16 s.	64 s.
8	29 s.	30 s.	15 s.	74 s.
9	26 s.	22 s.	19 s.	67 s.
10	31 s.	21 s.	14 s.	66 s.

Tabla 4.8 Tiempos de recuperación del sistema al reiniciarlo en su totalidad
Fuente: Elaboración propia

Cálculo de métrica:

Tl: Tiempo máximo en segundos propuesto para la métrica.
Tp: Promedio de los tiempos en segundos obtenidos en todos los casos de prueba realizado.

X = 1 – Tp/Tl
X = 1 – 68.5/300
X = 0.77

Interpretación:

El valor de la métrica es de 0.77 lo cual es muy favorable dado que cuanto más cercano a 1 será mejor.

4.1.12 Claridad de descripción

Datos recopilados de las encuestas realizadas a 10 personas a las que va dirigido el proyecto, con diferentes grados de instrucción, con una valoración de 1 a 5, siendo 1 lo peor y 5 lo mejor.

Sección / Encuestados	1	2	3	4	5	6	7	8	9	10
Conexión SSL	3	5	4	2	2	2	3	3	2	2
Crear cuenta de usuario	4	5	5	3	4	4	5	5	5	5
Login de usuario	5	5	5	3	5	5	5	5	5	5
Añadir contacto	5	5	5	3	5	5	5	5	5	5
Confirmar contacto	5	5	5	3	5	5	5	5	5	5
Ver mis contactos	5	5	5	3	5	5	5	5	5	5
Añadir documento	3	5	4	3	3	3	4	4	3	3
Listar documentos	5	5	5	5	5	5	5	5	5	5
Firmar documento	3	5	4	3	3	4	4	4	3	3
Verificar documento	3	5	5	3	3	3	4	4	3	3
Reportar documento	4	5	5	3	3	4	5	5	3	3
Historial de documento	4	5	5	3	3	4	5	5	3	3

Tabla 4.9 Resultados de encuesta en claridad de descripción

Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los datos obtenidos de las encuestas realizadas.
M: Valor máximo por encuesta.

X = P/M
X = 4.18/5
X = 0.84

Interpretación:

El valor de la métrica es de 0.84 lo cual es muy favorable dado que cuanto más cercano a 1 será mejor.

4.1.13 Comprensión de entrada y salida

Datos recopilados de las encuestas realizadas a 10 personas a las que va dirigido el proyecto, con diferentes grados de instrucción, con una valoración de 1 a 5, siendo 1 lo peor y 5 lo mejor.

Sección / Encuestados	1	2	3	4	5	6	7	8	9	10
Crear cuenta de usuario	3	5	5	3	4	4	5	5	5	5
Login de usuario	5	5	5	3	5	5	5	5	5	5
Añadir contacto	5	5	5	3	5	5	5	5	5	5
Confirmar contacto	5	5	5	3	5	5	5	5	5	5
Ver mis contactos	5	5	5	3	5	5	5	5	5	5
Eliminar contacto	5	5	5	3	5	5	5	5	4	4
Añadir documento	3	5	4	3	3	3	4	4	3	3
Listar documentos	4	5	5	5	5	5	5	5	5	5
Firmar documento	3	5	4	3	3	4	4	4	3	3
Verificar documento	3	5	5	3	3	3	4	4	3	3
Descargar documento	5	5	5	3	3	3	4	4	3	3
Descargar certificado	3	4	3	2	2	2	2	3	2	2
Reportar documento	3	4	5	3	3	4	5	5	3	3
Historial de documento	3	4	5	3	3	4	5	5	3	3

Tabla 4.10 Resultados de encuesta comprensión de entradas y salidas
Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los datos obtenidos de las encuestas realizadas.
M: Valor máximo por encuesta.

X = P/M
X = 4.12/5
X = 0.82

Interpretación:

El valor de la métrica es de 0.82 lo cual es muy favorable dado que cuanto más cercano a 1 será mejor.

4.1.14 Facilidad de aprender función

Datos recopilados de las capacitaciones realizadas a 10 personas a las que va dirigido el proyecto, con diferentes grados de instrucción, obteniendo los tiempos promedios en segundos que tomó el aprendizaje de cada sección.

Sección / Encuestados	1	2	3	4	5	6	7	8	9	10
Crear cuenta de usuario	120	120	120	300	120	120	120	120	180	180
Login de usuario	60	60	60	180	60	60	60	60	60	60
Añadir contacto	120	90	90	300	120	120	90	90	120	120
Confirmar contacto	60	60	60	180	90	90	60	60	90	90
Ver mis contactos	60	60	60	180	90	90	60	60	90	90
Eliminar contacto	60	60	60	180	90	90	60	60	90	90
Añadir documento	300	240	240	600	480	480	360	360	480	480
Listar documentos	60	60	60	300	90	90	60	60	90	90
Firmar documento	300	240	240	600	360	360	300	300	360	360
Verificar documento	60	60	60	240	180	180	60	60	180	180
Descargar documento	60	30	30	120	60	60	30	30	60	60
Descargar certificado	60	30	30	120	60	60	30	30	60	60
Reportar documento	180	120	120	300	180	180	120	120	180	180
Historial de documento	60	30	30	120	60	60	30	30	60	60
Promedio	**111**	**90**	**90**	**266**	**146**	**146**	**103**	**103**	**150**	**150**

Tabla 4.11 Resultados de tiempos recopilados en facilidad de aprender función

Fuente: Elaboración propia

Cálculo de métrica:

Tp: Tiempo promedio de los tiempos obtenidos de todos los usuarios.
Tm: Tiempo máximo esperado.

X = Tp/Tm
X = 135/600
X = 0.77

Interpretación:

El valor de la métrica es de 0.77 lo cual es muy favorable dado que cuanto más cercano a 1 será mejor.

4.1.15 Consistencia operacional en el uso

Evaluación realizada en las distintas secciones del sistema para determinar la claridad y compresión de los campos y mensajes presentados, siendo A “Aceptable” y I “Inconsistente”.

	Tarea	Criterio	Calificación
1	Registro de nuevo usuario.	El formulario es aceptable, las 8 cajas de texto y sus descripciones son claras, el correo enviado es consistente indicando los datos registrados.	A
2	Ingresar a la página principal después de loguearse en el sistema.	La página inicial es clara y ofrece accesos directos y simples a las funciones más utilizadas del sistema.	A
3	Añadir un contacto	Mensajes claros, y búsqueda con filtros adecuados para empezar a enviar una solicitud de contacto, aparece una ventana para confirmar el contacto que se está añadiendo.	A
4	Eliminar un contacto	El listado de nuestros contactos muestra e-mail, dni, nombres y apellidos y un ícono que da a entender la funcionalidad de eliminar un contacto, incluso tiene una descripción que dice “eliminar”.	A
5	Añadir un documento	El formulario es aceptable las 6 cajas de texto y sus descripciones son aceptables, aparece un mensaje claro indicando que el documento ha sido registrado	A

		satisfactoriamente.	
6	Listar documentos	Existen etiquetas para encontrar los 3 tipos de documentos que son “Pendientes, Firmados, Observando”, se puede ver la descripción de cada documento al pasar el mouse sobre cada ítem.	A
7	Ver documento	Los campos mostrados son claros, los mensajes de verificación por cada firmante se encuentran resaltados, además que cada verificación correcta colorea de color verde la zona de información de cada firmante.	A
8	Firmar documento	Los 2 campos requeridos para la firma del documento son claros y además de realizarse satisfactoriamente aparece un mensaje indicado que la firma del documento fue correcto	A
9	Ver historial	Se muestra claramente ordenado en forma descendente todos los eventos relacionados al documento, se colorean de color rojo aquellos eventos donde se reportaron observaciones en el documento.	A
10	Reportar documento	El mensaje es claro, este	A

		indique que se ingrese la descripción del problema encontrado.	
11	Ver firmantes	La ventana emergente muestra firmantes y su correspondiente estado indicando si este ya firmó el documento o si aún se encuentra pendiente de su firma, también existe una barra de progreso indica el porcentaje de firmado.	A
12	Ver observadores	La ventaja emergente es clara ya que muestra el listado de los observadores del documento.	A

Tabla 4.12 Claridad y comprensión de los formularios del sistema
Fuente: Elaboración propia

Cálculo de métrica:

A: Número Inconsistencias encontradas en la aplicación
B: Número de funciones o mensajes evaluados

X = 1 - 0/12
X = 1 - 0
X = 1

Interpretación:

El valor de la métrica es de 1 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.16 Corrección de errores

Datos recopilados de los tiempos requeridos para corregir los errores en los distintos casos de prueba realizados a 10 usuarios, con diferentes grados de instrucción, los resultados se encuentran en segundos.

Sección	1	2	3	4	5	6	7	8	9	10
Crear cuenta de usuario										
-No se ingresó un dato requerido.	60 s.	60 s.	60 s.	300 s.	120 s.	120 s.	60 s.	60 s.	120 s.	120 s.
Login de usuario										
-Nombre de usuario incorrecto.	30 s.	30 s.	30 s.	120 s.	60 s.	60 s.	30 s.	30 s.	60 s.	60 s.
-Contraseña de usuario incorrecto.	30 s.	30 s.	30 s.	120 s.	60 s.	60 s.	30 s.	30 s.	60 s.	60 s.
Añadir contacto										
-Nombre de usuario no registrado.	60 s.	60 s.	60 s.	180 s.	60 s.	60 s.	60 s.	60 s.	60 s.	60 s.
Confirmar contacto										
-No se confirma la agregación.	30 s.	30 s.	30 s.	120 s.	30 s.	30 s.	30 s.	30 s.	30 s.	30 s.
Eliminar contacto										
-No se confirma la eliminación.	30 s.	30 s.	30 s.	120 s.	30 s.	30 s.	30 s.	30 s.	30 s.	30 s.
Añadir documento										
-No se selecciona documento.	60 s.	30 s.	30 s.	180 s.	60 s.	60 s.	30 s.	30 s.	120 s.	60 s.
-No se seleccionan firmas requeridas.	60 s.	30 s.	30 s.	240 s.	60 s.	60 s.	30 s.	30 s.	120 s.	60 s.
-No se seleccionan observadores.	60 s.	30 s.	30 s.	180 s.	60 s.	60 s.	30 s.	30 s.	100 s.	60 s.
Firmar documento										
-No se selecciona archivo pkcs.	60 s.	60 s.	60 s.	180 s.	90 s.	90 s.	60 s.	60 s.	90 s.	90 s.
-Contraseña de pkcs incorrecta.	60 s.	60 s.	60 s.	120 s.	60 s.	60 s.	60 s.	60 s.	90 s.	90 s.
Descargar documento										
-No se acepta la descarga del documento.	30 s.	30 s.	30 s.	120 s.	60 s.	60 s.	30 s.	30 s.	60 s.	60 s.
Descargar certificado										
-No se acepta la descarga del archivo.	30 s.	30 s.	30 s.	120 s.	60 s.	60 s.	30 s.	30 s.	60 s.	60 s.
Reportar documento										
-No se ingresa un mensaje de reporte.	60 s.	60 s.	60 s.	240 s.	120 s.	120 s.	60 s.	60 s.	90 s.	90 s.
Promedio	47.14	40.71	40.71	167.14	66.43	66.43	40.71	40.71	77.86	66.43

Tabla 4.13 Tiempos que cada usuario requirió para corregir sus errores

Fuente: Elaboración propia

Cálculo de métrica:

Tp: Promedio de los tiempos obtenidos de todos los usuarios.
Tm: Tiempo máximo esperado

X = Tp/Tm
X = 65.43/300
X = 0.78

Interpretación:

El valor de la métrica es de 0.78 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.17 Entendibilidad de mensaje en uso

Datos recopilados de las encuestas realizadas a 10 usuarios a los que va dirigido el proyecto y con diferentes grados de instrucción, los resultados en una escala del 1 al 5, siendo 1 lo peor y 5 lo mejor.

Sección / Encuestados	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Crear cuenta de usuario	5	5	5	2	3	3	5	5	3	3
Login de usuario	5	5	5	3	5	5	5	5	5	5
Añadir contacto	5	5	5	3	4	4	5	5	4	4
Confirmar contacto	5	5	5	2	3	3	4	4	3	3
Ver mis contactos	5	5	5	3	3	3	5	5	4	4
Eliminar contacto	5	5	5	3	3	3	5	5	4	4
Añadir documento	5	5	5	2	3	3	4	4	3	3
Listar documentos	5	5	5	3	3	3	5	5	3	3
Firmar documento	4	4	4	2	2	2	3	3	2	2
Verificar documento	4	4	4	3	3	3	3	3	3	3
Descargar documento	4	4	4	3	3	3	4	4	3	3
Descargar certificado	4	4	4	3	3	3	4	4	3	3
Reportar documento	4	4	4	2	3	3	4	4	3	3
Historial de documento	4	4	4	2	3	3	4	4	3	3
Promedio	**4.57**	**4.57**	**4.57**	**2.57**	**3.14**	**3.14**	**4.29**	**4.29**	**3.29**	**3.29**

Tabla 4.14 Resultados de encuesta sobre entendibilidad de mensaje en uso

Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los resultados obtenidos de todos los usuarios.
M: Puntaje máximo de los ítems de la encuesta.

X = P/M
X = 3.77/5
X = 0.75

Interpretación:

El valor de la métrica es de 0.75 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.18 Interacción atractiva

Datos recopilados de las encuestas realizadas a 10 usuarios a los que va dirigido el proyecto y con diferentes grados de instrucción, los resultados en una escala del 1 al 5, siendo 1 lo peor y 5 lo mejor.

Sección / Encuestado	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Crear cuenta de usuario	5	5	4	3	4	4	5	5	4	4
Login de usuario	5	5	5	4	5	5	5	5	5	5
Pantalla principal	5	5	5	3	5	5	5	4	5	5
Añadir contacto	4	4	4	3	3	3	4	4	3	3
Ver mis contactos	4	4	4	3	3	3	4	4	3	3
Mis solicitados	4	4	4	3	4	4	4	4	3	3
Añadir documento	4	4	4	3	4	4	4	4	3	3
Mis documentos	5	5	4	3	4	4	5	5	4	4
Firmar documento	5	5	5	3	4	4	4	4	3	3
Verificar documento	5	5	5	4	5	5	4	4	5	5
Reportar documento	4	4	5	4	4	4	4	4	4	4
Historial de documento	3	3	4	3	4	4	4	4	4	4
Promedio	**4.42**	**4.42**	**4.42**	**3.25**	**4.08**	**4.08**	**4.33**	**4.25**	**3.83**	**3.83**

Tabla 4.15 Resultados de encuesta sobre interacción atractiva
Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los resultados obtenidos de todos los usuarios.
M: Puntaje máximo de los ítems de la encuesta.

X = P/M
X = 4.09/5

X = 0.82

Interpretación:

El valor de la métrica es de 0.82 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.19 Tiempo de Respuesta

Medidas tomadas por funciones de las páginas del sistema usando la herramienta de desarrollo, medición de red en Firefox.

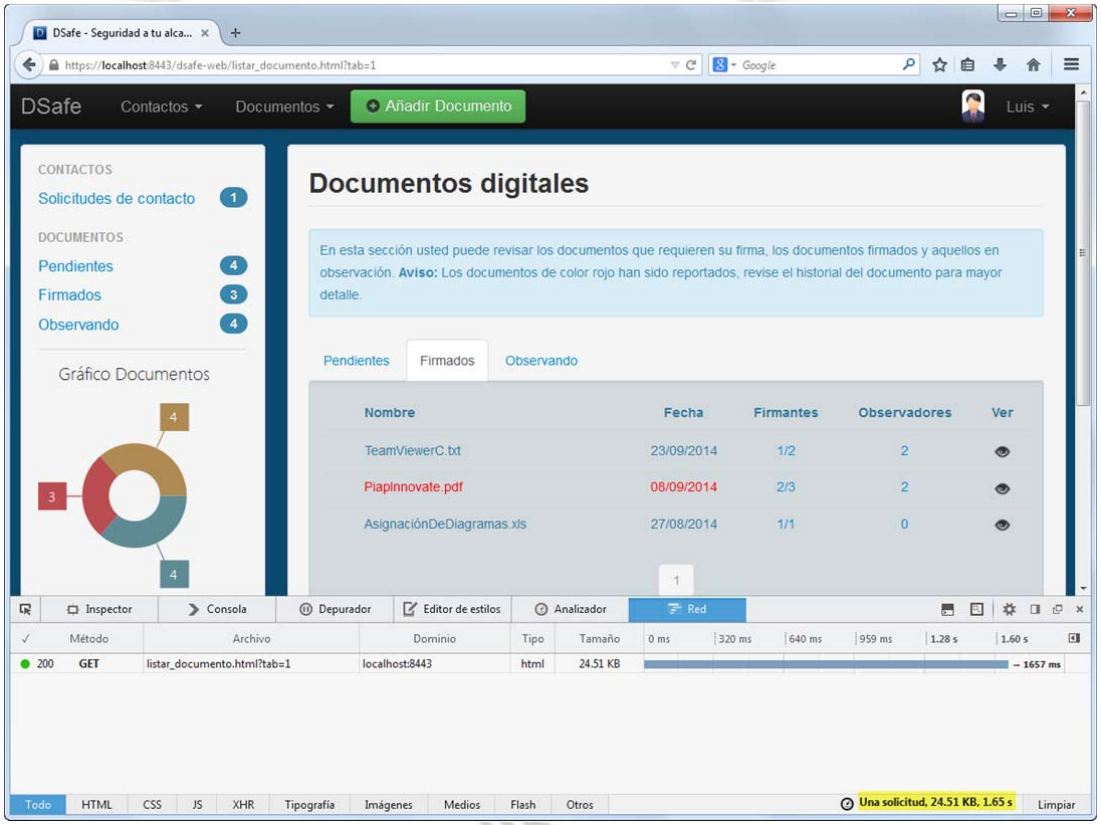


Figura 4.5 Medición de tiempo de respuesta en peticiones al sistema usando Firefox

Fuente: Elaboración propia

	Páginas y funciones del sistema	**Tiempo**
1	Carga de la página principal al loguearse	1.38 seg.
2	Cargar listado de usuarios	0.94 seg
3	Cargar listado de solicitudes de contacto	1.00 seg

4	Cargar listado de contactos aceptados	1.00 seg.
5	Aceptar contacto	1.03 seg.
6	Rechazar contacto	1.08 seg.
7	Añadir contacto	0.79 seg
8	Cargar listado de los documentos	1.65 seg.
9	Añadir documento (el formulario fue llenado y se presiona el botón guardar)	7.16 seg.
10	Eliminar documento	1.98 seg.
11	Ver documento (se incluye la verificación de las firmas del documento)	1.09 seg.
12	Cargar historial del documento	0.46 seg.
13	Reportar documento (el formulario fue llenado y se presiona el botón aceptar)	1.71 seg
14	Firmar documento (el formulario fue llenado y se presiona el botón aceptar)	2.78 seg.
15	Cargar listado de firmantes en ventana emergente	0.49
16	Cargar listado de observadores en ventana emergente	0.50

Tabla 4.16 Tiempos de respuesta por páginas y funcionalidad del sistema
Fuente: Elaboración propia

Cálculo de métrica:

Tp: Tiempo promedio obtenido
Tmax: Tiempo máximo aceptable de carga (5 seg.)

X = 1 - Tp/Tmax
X = 1 – 1.57/5
X = 0.69

Interpretación:

El valor de la métrica es de 0.69 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.20 Rendimiento

Se usó la herramienta AgileLoad, para realizar las siguientes acciones:

- Loguear en el sistema
- Añadir un documento
- Firmar un documento
- Obtener listado de documentos
- Verificar documento

Se hizo la simulación con 10 usuarios los cuales entrarán en la simulación cada 2 segundos, se ejecutaron las tareas descritas en ciclos en un tiempo límite de 15 minutos.

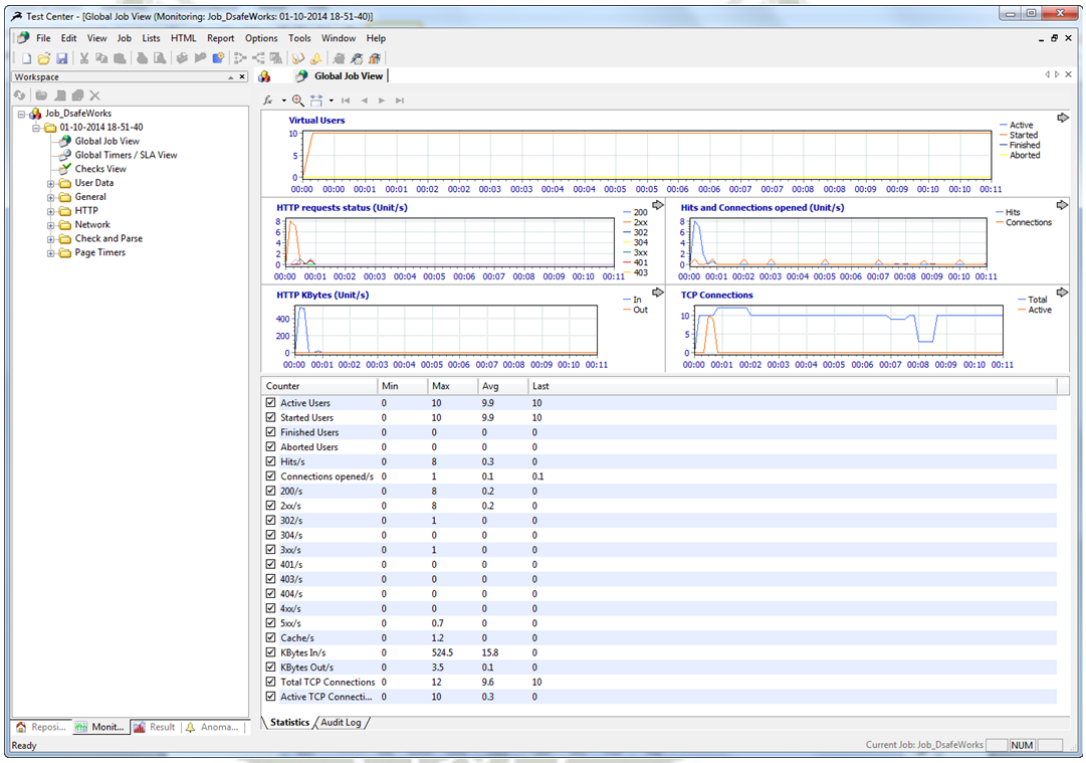


Figura 4.6 Medición de carga del sistema usando AgileLoad
Fuente: Elaboración propia

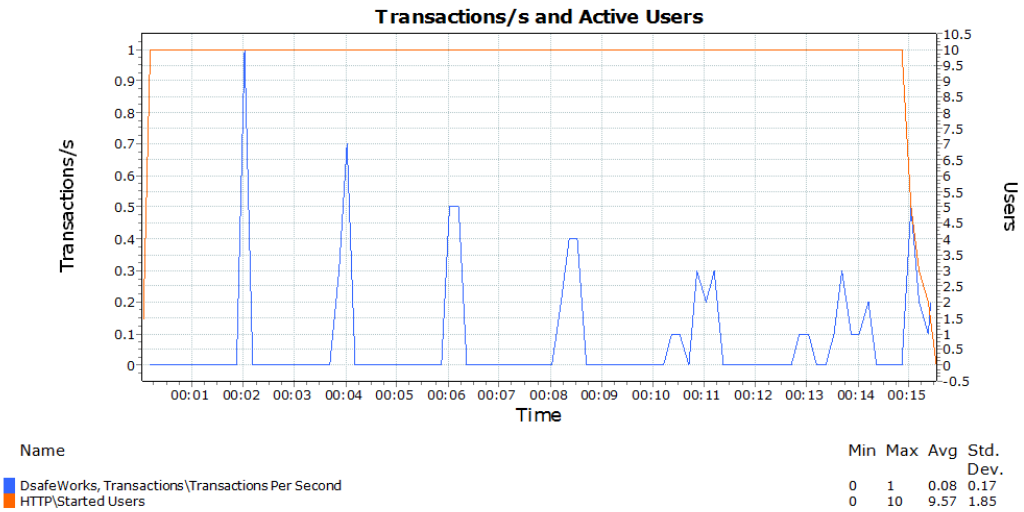


Figura 4.7 Transacciones por segundo realizadas con 10 usuarios concurrentes
Fuente: Elaboración propia

4.1.4 HTTP Summary

Rampup Step 1: 10 Virtual Users; From mié 17:58:57 To mié 18:13:47

Counter Name	Min	Max	Avg	Total
Hits/s	0	13	1.11	986
Connections opened/s	0	1	0.02	21
Failed Requests/s	0	0	0	0
Request Retries/s	0	0	0	0
Input KiloBytes/s	0	782.2	32.12	28583
Output KiloBytes/s	0	158.5	12.46	11087
200/s	0	12	0.88	783
2xx/s	0	12	0.88	783
302/s	0	1	0.22	197
304/s	0	0	0	0
3xx/s	0	1	0.22	197
401/s	0	0	0	0
403/s	0	0	0	0
404/s	0	0	0	0
4xx/s	0	0	0	0
5xx/s	0	0.1	0	1
Cache/s	0	29	7.35	6544
Total Tcp Connections/s	1	2	1.99	1770
Active Tcp Connections/s	0	1	0.33	291
Network errors/s	0	0	0	0
Parsing errors/s	0	0	0	0
Checks succeeded/s	0	1.6	0.52	463
Checks failed/s	0	0.1	0	1

Tabla 4.17 Resumen de peticiones efectuadas en el sistema con 10 usuarios concurrentes en un intervalo de 15 minutos
Fuente: Elaboración propia

Cálculo de métrica:

A: Número de operaciones realizadas en el intervalo de 15 minutos

B: Número de operaciones esperadas en el intervalo de 15 minutos

X = A/B
X = 463/500
X = 0.93

Interpretación:

El valor de la métrica es de 0.93 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.21 Capacidad de pistas de auditoria

Puntaje asignado a los registros del log para determinar su utilidad en cuanto a reconocer la ubicación donde se generó el mensaje. Valoración de 1 a 5 siendo 1 lo peor y 5 lo mejor.

Sección	ERROR	WARNING	INFO
Crear cuenta de usuario	5	4	5
Login de usuario	5	4	5
Añadir contacto	3	4	5
Confirmar contacto	4	4	5
Ver mis contactos	4	4	5
Eliminar contacto	4	4	5
Añadir documento	3	4	5
Listar documentos	5	4	5
Firmar documento	3	4	5
Verificar documento	3	4	5
Descargar documento	5	4	5
Descargar certificado	5	4	5
Reportar documento	4	4	5
Historial de documento	4	4	5
Promedio	**4.07**	**4.00**	**5.00**

Tabla 4.18 Evaluación de calidad de información mostrada en el log para reconocer la ubicación donde se generó del mensaje
Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los resultados obtenidos.
M: Puntaje máximo por cada sección evaluada.

X = P/M
X = 4.07/5

X = 0.81

Interpretación:

El valor de la métrica es de 0.81 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.22 Capacidad de análisis de falla

Puntaje asignado a los mensajes del log para determinar su utilidad en cuanto a reconocer la causa que generó el mensaje. Valoración de 1 a 5 siendo 1 lo peor y 5 lo mejor.

Sección	ERROR	WARNING	INFO
Crear cuenta de usuario	4	4	5
Login de usuario	4	4	5
Añadir contacto	3	4	5
Confirmar contacto	4	4	5
Ver mis contactos	4	4	5
Eliminar contacto	4	4	5
Añadir documento	3	4	5
Listar documentos	4	4	5
Firmar documento	3	4	5
Verificar documento	3	4	5
Descargar documento	4	4	5
Descargar certificado	4	4	5
Reportar documento	4	4	5
Historial de documento	4	4	5
Promedio	**3.71**	**4.00**	**5.00**

Tabla 4.19 Evaluación de calidad de información mostrada en el log para reconocer la causa del mensaje

Fuente: *Elaboración propia*

Cálculo de métrica:

P: Promedio de los resultados obtenidos.
M: Puntaje máximo por cada sección evaluada.

X = P/M
X = 3.71/5
X = 0.74

Interpretación:

El valor de la métrica es de 0.74 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.23 Complejidad de modificación

Datos recopilados de los registros de cambios (teniendo en cuanto la complejidad del cambio y la duración) realizados en el proyecto por cada sección importante. Valores de 1 a 5, donde 1 es lo menos complejo y 5 es lo más complejo.

Sección	**Complejidad**
Crear cuenta de usuario	2
Login de usuario	1
Añadir contacto	2
Confirmar contacto	1
Ver mis contactos	1
Eliminar contacto	1
Añadir documento	3
Listar documentos	1
Firmar documento	4
Verificar documento	3
Descargar documento	1
Descargar certificado	1
Reportar documento	1
Historial de documento	1
Promedio	**1.64**

Tabla 4.20 Complejidad en cambios realizados en el sistema por funcionalidad

Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los resultados obtenidos.
M: Puntaje máximo por cada sección evaluada.

X = 1 - P/M
X = 1 - 1.64/5
X = 0.67

Interpretación:

El valor de la métrica es de 0.67 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.24 Capacidad de controlar el cambio de software

Datos recopilados del número de versiones generadas por cada sección del proyecto.

Sección	Versiones
Crear cuenta de usuario	3
Login de usuario	2
Añadir contacto	5
Confirmar contacto	3
Ver mis contactos	3
Eliminar contacto	2
Añadir documento	5
Listar documentos	2
Firmar documento	6
Verificar documento	8
Descargar documento	2
Descargar certificado	2
Reportar documento	2
Historial de documento	3
Promedio	3.43

Tabla 4.21 Versiones generadas de código fuente por funcionalidad
Fuente: Elaboración propia

Cálculo de métrica:

P: Promedio de los resultados obtenidos.
M: Valor mínimo esperado de versiones realizadas.

X = P/M
X = 3.43/5
X = 0.68

Interpretación:

El valor de la métrica es de 0.68 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.25 Facilidad de portabilidad para el usuario

Se utilizó la herramienta SortSite para medir la compatibilidad de la aplicación web en diferentes navegadores.

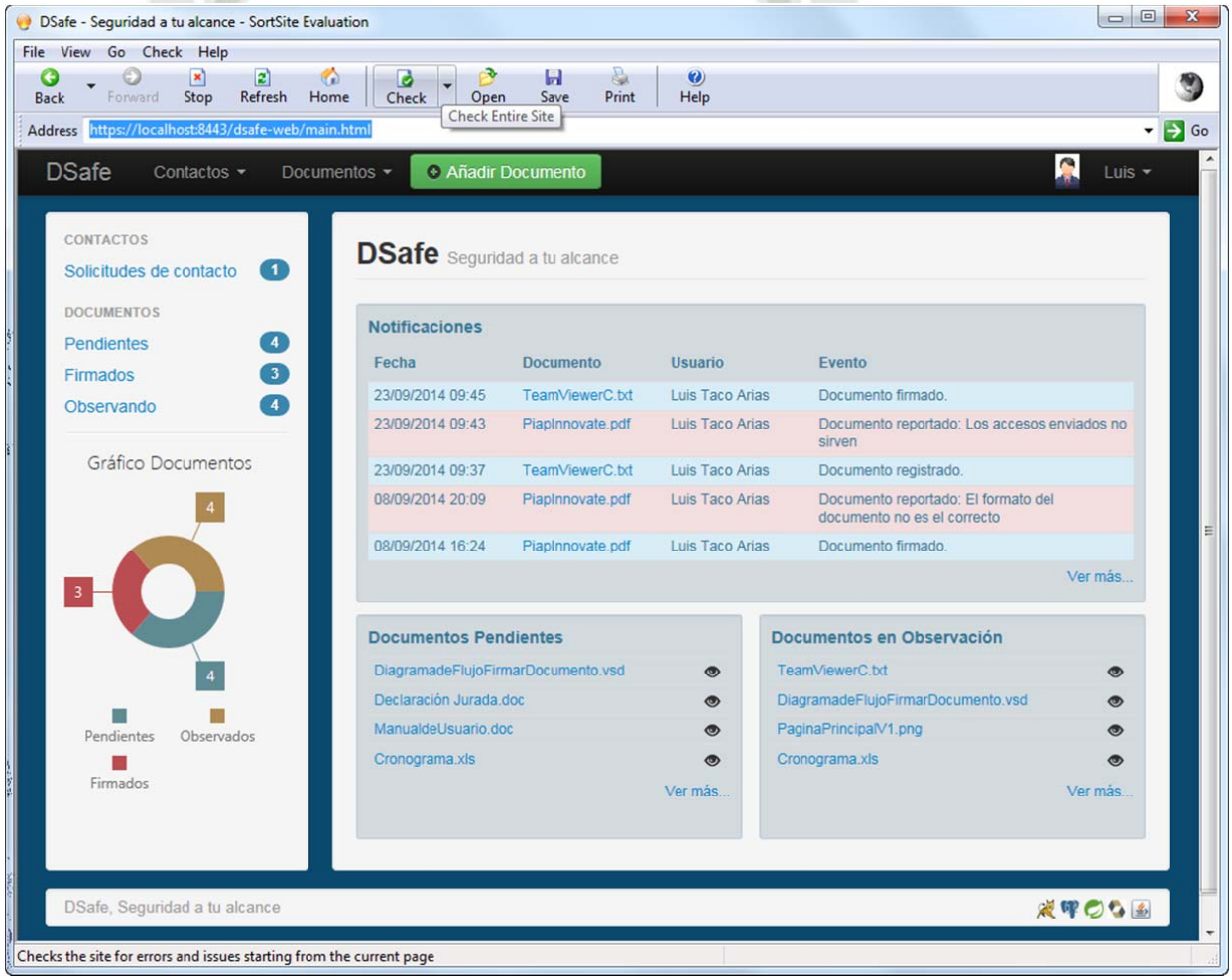


Figura 4.8 Evaluación web del sistema usando SortSite
Fuente: Elaboración propia

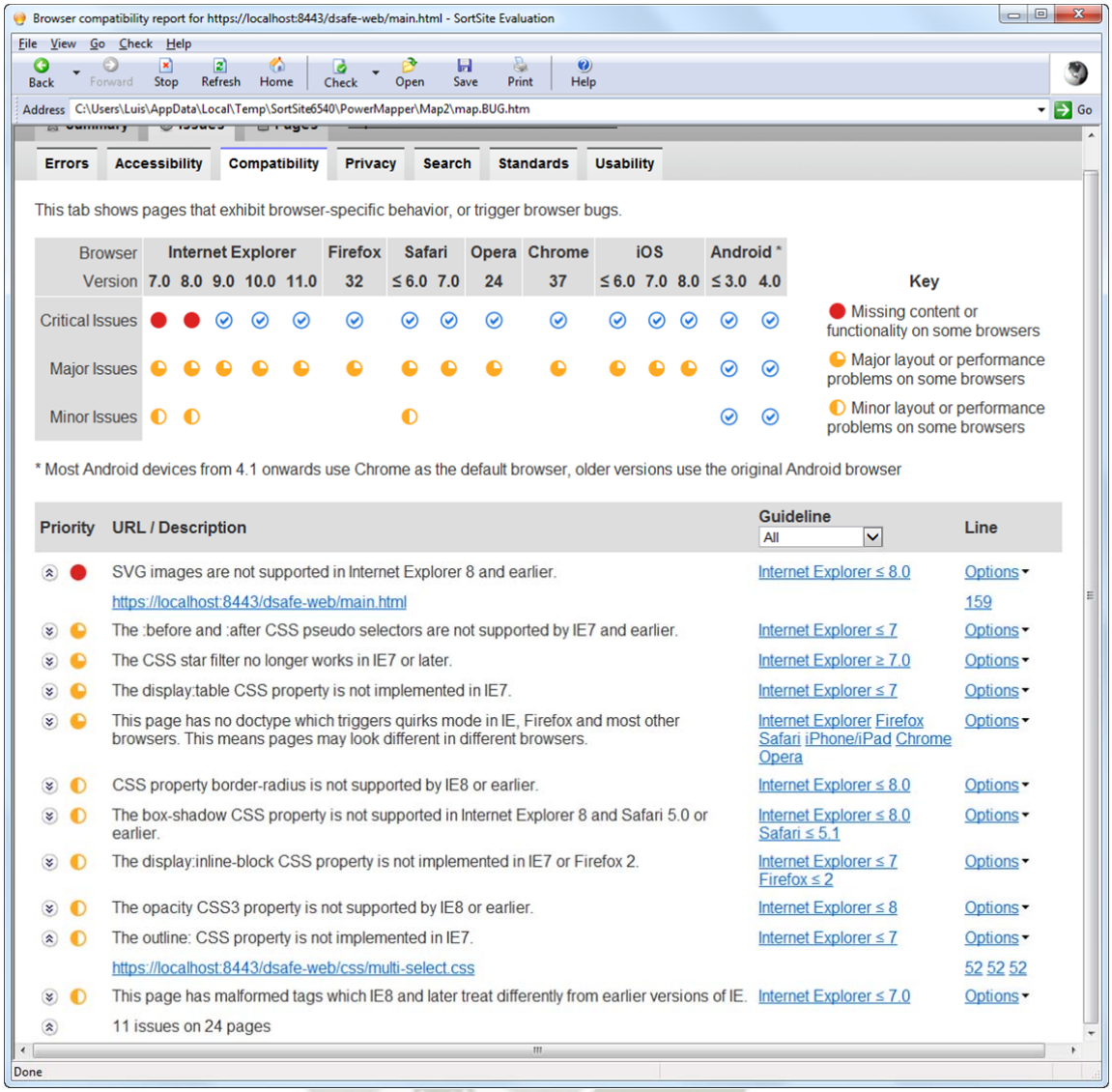


Figura 4.9 Resultados de compatibilidad web usando SortSite
Fuente: Elaboración propia

Registrar el grado de compatibilidad del sitio web de acuerdo tomando en cuenta estas referencias para el puntaje: 0 malo, 0.25 casi malo, 0.5 regular, 0.75 casi bueno y 1 bueno.

Navegador	**Puntaje obtenido**
IE 7.0	0.6
IE 8.0	0.6
IE 9.0	0.7
IE 10.0	0.7

IE 11.0	0.9
Firefox	0.9
Safari	0.85
Opera	0.9
Chrome	0.9
IOS	0.85
Android	1
PROMEDIO	0.81

Tabla 4.22 Puntaje de compatibilidad de la aplicación con distintos navegadores web

Fuente: Elaboración propia

Cálculo de métrica:

Promedio del puntaje obtenido en la compatibilidad con diferentes navegadores es 0.81

Interpretación:

El valor de la métrica es de 0.81 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.1.26 Facilidad de instalación

La instalación incluye:

- Instalación de Postgress
- Creación de la base de datos
- Instalación de Java jdk1.6.0_34
- Instalación de Apache Tomcat 7.0
- Configuración e instalación de certificado SSL
- Despliegue del sistema (archivo .war)

Instalación y configuración realizadas	**Nro. intentos**	**Errores en la instalación**
Instalación de Postgress	1	
Creación de la base de datos	1	
Instalación de Java jdk1.6.0_34	1	
Instalación de Apache Tomcat 7.0	2	• Problemas en la configuración de la versión del jdk de java, se necesitaron modificar las variables de entorno para el uso específico de jdk1.6.0_34.
Configuración e instalación de certificado SSL	4	• Problemas con el puerto que se encontraba usado por otras aplicaciones. • Problemas en el protocolo para SSL el cual debía configurarse como "org.apache.coyote.http11.Http11Protocol". • Se necesitó cambiar la configuración en server.xml de apache para indicar la dirección del nuevo certificado SSL.
Despliegue del sistema (archivo .war)	1	
TOTAL	10	

Tabla 4.23 Problemas encontrados al intentar cambiar la instalación del sistema

Fuente: Elaboración propia

Cálculo de métrica:

A: Número de casos en los que el usuario cambió exitosamente la instalación.
B: Número total de casos en los un usuario intentó cambiar la instalación.

X = A/B
X = 6/10
X = 0.60

Interpretación:

El valor de la métrica es de 0.60 lo cual es favorable dado que cuanto más cercano sea a 1 será mejor.

4.2 RESULTADOS DE LAS MÉTRICAS DE CALIDAD:

A continuación se muestra el resultado de las pruebas de acuerdo a cada uno de los indicadores planteados en la norma NTP-ISO/IEC 9126.

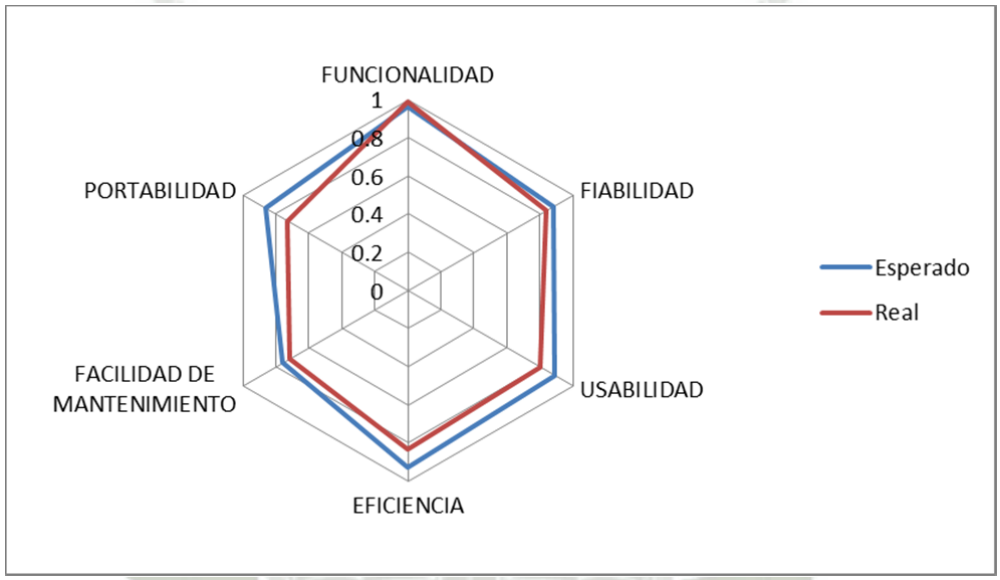


Figura 4.10 Indicadores obtenidos usando la norma NTP-ISO/IEC 9126
Fuente: *Elaboración propia*

E	R	Característica	P	E	R	SubCaracterística	P	E	R	Métrica	P	E	R
0.899	0.853	FUNCIONALIDAD	0.3	0.96	0.99	APLICABILIDAD	0.2	1	1				
										Integridad de Implementación Funcional	1	1	1
						PRECISIÓN	0.1	1	1				
										Exactitud de Cálculo	1	1	1
						INTEROPERABILIDAD	0.2	0.9	1				
										Intercambiabilidad de Datos	1	0.9	1
						SEGURIDAD	0.5	0.965	0.983	Control de Acceso	0.4	1	1
										Control de Acceso Concurrente.	0.1	0.9	0.83
										Prevención de Corrupción de los Datos	0.5	0.95	1
		FIABILIDAD	0.2	0.88	0.84	MADUREZ	0.4	0.9	0.88				
										Densidad de Fallas	1	0.9	0.88
						TOLERANCIA A FALLOS	0.4	0.875	0.815	Prevención de Caídas	0.5	0.85	0.63
										Prevención de operación incorrecta	0.5	0.9	1
						RECUPERABILIDAD	0.2	0.87	0.794	Disponibilidad	0.4	0.9	0.83
										Tiempo medio de recuperación	0.6	0.85	0.77
		USABILIDAD	0.2	0.89	0.80	ENTENDIBILIDAD	0.3	0.9	0.828	Claridad de la Descripción	0.4	0.9	0.84
										Comprensión de Entradas y Salidas	0.6	0.9	0.82
						FACILIDAD DE APRENDIZAJE	0.3	0.9	0.77	Facilidad de Aprender la Función	1	0.9	0.77
						OPERABILIDAD	0.3	0.895	0.809	Consistencia Operacional en el Uso	0.2	0.9	1
										Corrección de errores	0.3	0.8	0.78
										Entendibilidad del Mensaje en Uso	0.5	0.95	0.75
						ATRACTIVIDAD	0.1	0.8	0.82	Interacción Atractiva	1	0.8	0.82
		EFICIENCIA	0.1	0.93	0.83	COMPORTAMIENTO EN EL TIEMPO	1	0.93	0.834	Tiempo de Respuesta	0.4	0.9	0.69
										Rendimiento	0.6	0.95	0.93
		FACILIDAD DE MANTENIMIENTO	0.1	0.76	0.72	ANALIZABILIDAD	0.4	0.77	0.782	Capacidad de Pistas de Auditoría	0.6	0.75	0.81
										Capacidad de Análisis de Fallas	0.4	0.8	0.74
						CAMBIABILIDAD	0.6	0.76	0.674				
										Complejidad de Modificación	0.6	0.8	0.67
										Capacidad de Controlar el Cambio del Software	0.4	0.7	0.68
		PORTABILIDAD	0.1	0.86	0.73	ADAPTABILIDAD	0.6	0.9	0.81				
										Facilidad de Portabilidad para el usuario	1	0.9	0.81
						INSTALABILIDAD	0.4	0.8	0.6	Facilidad de Instalación	1	0.8	0.6

Tabla 4.24 Valores de métricas de calidad de software obtenidos usando la norma NTP-ISO/IEC 9126
Fuente: *Elaboración propia*

Siendo: **E** = Valor esperado, **R=** Valor real obtenido, **P** = Ponderación o peso

CONCLUSIONES

1. El empleo de Firmas Digitales, Encriptación Asimétrica y Protocolos SSL ha permitido implementar un sistema web que incremente la seguridad de los documentos digitales, con el cual se protege la integridad de los documentos después de ser firmados, se identifica de forma fehaciente a los autores y se evita el no repudio de los firmantes.
2. De todas las metodologías ágiles de desarrollo estudiadas se escogió SCRUM por su simplicidad y flexibilidad en cuanto al desarrollo de los módulos y el hecho de permitir una comunicación más fluida entre los miembros del equipo.
3. Mientras más antigua es una técnica criptográfica vigente, esta tiene un mayor respaldo en su seguridad debido el tiempo empleado en su estudio.
4. El sistema implementado resulta cómodo y fácil de usar escondiendo al usuario todo el mecanismo interno que se realiza para dar seguridad a sus documentos, quedando por ende como responsabilidad de cada usuario el saber gestionar y mantener en secreto su clave privada.
5. Los resultados obtenidos luego de realizar pruebas utilizando la norma NTP-ISO/IEC 9126 respaldan la calidad del producto desarrollado, con un puntaje de 0.853, obteniendo puntajes más altos en los indicadores de funcionalidad y fiabilidad, las cuales son los más importantes a considerar en un proyecto de este tipo.
6. El desarrollo de aplicaciones utilizando herramientas libres requiere un mayor esfuerzo comparado a utilizar herramientas licenciadas, pero a su vez nos brinda mayor control sobre lo desarrollado y conciencia sobre lo que sucede internamente, lo cual impulsa el aprendizaje de los desarrolladores.

RECOMENDACIONES

1. Realizar la integración del proyecto con alguna agencia gubernamental encargada de la emisión de certificados digitales (RENIEC), para dar un respaldo legal a las operaciones realizadas con el sistema.
2. Crear métodos de alerta por medio de dispositivos móviles los cuales se encarguen de informar al usuario sobre las acciones que se realizan sobre los documentos que ha registrado en el sistema.
3. La clave privada (PKCS12) debe ser almacenada en un llavero USB de uso exclusivo, de tal manera que sólo el usuario propietario de la clave privada pueda hacer uso de esta.
4. Extender la funcionalidad del sistema para permitir la lectura de claves privadas almacenadas en tokens, como por ejemplo lectura de tarjetas.

BIBLIOGRAFÍA

[HUID05] HUIDOBRO, José M. y ROLDAN, David. Seguridad en Redes y Sistemas Informáticos, Edición 2005

[IBEA00] IBEAS, Ángel, DÍAZ, José M. y de la HOZ, Daniel. E-Logistics(I) Nuevas Tecnologías de la Información, Logis Book Edición 2000

[MENE96] MENEZES, Alfred J, OORSCHOT, Paul C. y VANSTONE Scott A. Handbook of Applied Cryptography. CRC Press Octubre 1996.

[LUCE09] LUCENA, Manuel J. Criptografía y Seguridad en Computadores. UNIVERSIDAD DE JAÉN (España) Edición 2009.

[RAMI06] RAMIO, Jorge. Libro Electrónica de Seguridad Informática y Criptografía. UNIVERSIDAD POLITÉCNICA DE MADRID Edición 2006.

[GARC04] GARCIA, Joaquín. HERRERA Jordi. PERRAMÓN, Xavier. Aspectos Avanzados de Seguridad en Redes. Fundación para la Universidad Oberta de Catalunya Edición 2004.

[CARR04] CARRACEDO, Justo. Seguridad en Redes Telemáticas. UNIVERSIDAD POLITÉCTINA DE MADRID Edición 2004.

[MART02] MARTIN, Frank. SSL Certificates HOWTO. THE LINUX DOCUMENTATION PROJECT, Revision v0.5 2002.

[LOPE07] LÓPEZ HERNANDEZ, Fernando. Seguridad Criptografía y Comercio Electrónico en Java, Febrero 2007.

[ROME03] ROMERO FLORES, Jaime Manuel. La Seguridad Jurídica y la Contratación Electrónica en la empresa Arequipeña, Tesis presentada en la UCSM Arequipa - 2003.

REFERENCIAS

[WWW01] Portal de tramitación telemática de boletines de telecomunicaciones de FENIE (Federación Nacional de Empresarios de Instalaciones Eléctricas y de Telecomunicaciones de España).
https://www.tramitaciontelematica.es/fdigital/F_digital.htm

[WWW02] Portal de venHelix.com Inc., empresa desarrolladora de software para telecomunicaciones, SSL tutorial 2004.
http://www.eventhelix.com/realtimemantra/networking/SSL.pdf

[WWW03] Wikipedia la enciclopedia libre, PKCS.
http://es.wikipedia.org/wiki/PKCS

[WWW04] Wikipedia La enciclopedia Libre, Certificate Signing Request.
http://en.wikipedia.org/wiki/Certificate_signing_request

[WWW05] Wikipedia la enciclopedia libre, PKCS12
http://en.wikipedia.org/wiki/PKCS_12

[WWW06] RSA Laboratories
http://www.rsa.com/rsalabs/node.asp?id=2251

[WWW07] Wikipedia la enciclopedia libre, Lista Revocación de Certificados
http://es.wikipedia.org/wiki/Lista_de_revocaci%C3%B3n_de_certificados

[WWW08] Wikipedia la enciclopedia libre, Online Certificate Status Protocol.
http://es.wikipedia.org/wiki/Online_Certificate_Status_Protocol

[WWW09] Oracle Java SE Documentation, PKI Programmers Guide.
http://docs.oracle.com/javase/6/docs/technotes/guides/security/certpath/CertPathProgGuide.html

[WWW10] Wikipedia La enciclopedia Libre, Certification Path Algoritm.
http://en.wikipedia.org/wiki/Certification_path_validation_algorithm

[WWW11] Página principal de Apache JMeter.
http://jmeter.apache.org/

[WWW12] AgileLoad Herramienta de Testeo de carga y desempeño para sitios web. http://www.agileload.com/

[WWW13] SortSite herramienta de verificación de errores, links y accesibilidad para sitios web. http://www.powermapper.com/products/sortsite/

ANEXO A: GLOSARIO DE TÉRMINOS

A

- **Ataque de fuerza bruta:** se denomina ataque de fuerza bruta a la forma de recuperar una clave probando todas las combinaciones posibles hasta encontrar aquella que permite el acceso.
- **Ataque por diccionario:** Consiste en averiguar un password o contraseña probando todos los ítems de un repositorio de palabras llamado diccionario, prueba ser más eficaz que la fuerza bruta dado que la mayoría de personas utilizan palabras de una misma lengua en sus contraseñas.
- **Autoridad de Certificación (CA):** persona jurídica pública o privada que presta indistintamente servicios de producción, emisión, gestión, cancelación u otros servicios inherentes a la certificación digital. Asimismo, puede asumir las funciones de registro o verificación.
- **ASN.1:** Abstract Syntax Notation One (notación sintáctica 1, ASN. 1) es una notación para describir data que es transmitida por los protocolos de telecomunicación, sin importar el lenguaje, representación física, la aplicación que lo usa o si es complejo o simple.

B

- **Bit:** Acrónimo de Binary Digit, representa un dígito del sistema de numeración binario el cual puede tener los valores 0 ó 1.

C

- **Certificado digital:** documento electrónico generado y firmado digitalmente por una entidad de certificación el cual vincula un par de claves con una persona natural o jurídica confirmando su identidad.
- **Clave (Criptografía):** Representa una pieza de información (normalmente números o letras) que controla la operación de un algoritmo de criptografía en el aspecto que a través de estas se transforma un texto plano en uno cifrado y viceversa.
- **CRLs:** (Listas de Certificados Revocados): es una lista de números de serie de certificados digitales revocados por una autoridad de certificación concreta.

Dicha lista está firmada digitalmente por la propia autoridad de certificación.

- **CSR:** Certificate Signing Request, es una petición realizada a una autoridad de certificación con el motivo de poder obtener un certificado digital, dicha petición incluye datos de la persona solicitante, su organización y una clave pública asimétrica, el mensaje debe ser firmado con la clave privada asimétrica del solicitante.

D

- **DN:** (Distinguished Name) Nombre distintivo, es un identificador único usado en el estándar X.500.

H

- **Hash:** Se refiere a un método o función para generar claves que representen de manera casi unívoca un documento, registro o archivo.
- **Http: (**Hypertext Transfer Protocol traducido protocolo de transferencia de hipertexto) es el protocolo usado en cada transacción efectuada en la World Wide Web.

J

- **J2EE:** también conocido como java 2 Platform Enterprise Edition, es una plataforma de programación para el lenguaje de programación java, a partir del cual se desarrollan y ejecutan aplicaciones con arquitectura distribuida de n niveles, basándose ampliamente en el uso de componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

P

- **PKI:** Public Key Infraestructure (Infraestructura de clave pública) es una combinación de hardware, software, políticas y procedimientos de seguridad que permite la ejecución de operaciones criptográficas, las firmas digitales y el no repudio de transacciones electrónicas.
- **PKIX:** Public-Key Infraestructure (X.509), infraestructura de clave pública basada en el estándar de certificados X.509.
- **P7s:** extensión de un archivo que utiliza la norma PKCS7 que define la estructura de un mensaje con firma digital.

R

- **RFC:** Request for Comments, son una serie de publicaciones del Internet Engineering Task Force (IETF) que describen diversos aspectos del funcionamiento de internet y otras redes de computadoras, como protocolos, procedimientos o comentarios e ideas sobre estos.
- **ROM:** Proviene del acrónimo Read Only Memory, es una clase de medio de almacenamiento utilizado en los ordenadores y otros dispositivos electrónicos el cual permite almacenar información digital para que después esta no pueda ser modificada, permitiendo posteriormente únicamente su lectura; existen variaciones de este tipo de dispositivos que permiten más de una actualización en su contenido.

S

- **SSL:** Transport Layer Security (Protocolo de Capa de Conexión Segura), es un protocolo que proporciona privacidad y autenticación de la información entre extremos sobre internet mediante el uso de la encriptación.

T

- **Tarjeta Inteligente:** Cualquier tarjeta del tamaño de un bolsillo con circuitos integrados que permiten la ejecución de cierta lógica programada.
- **Trusted Third Party (TTP):** es una entidad que facilita interacciones entre dos participantes quienes confían en este tercero. El tercero de confianza revisa todas las comunicaciones transaccionales críticas de los miembros.

X

- **X509:** Estándar para certificados de claves públicas el cual define el formato de un certificado digital y un algoritmo de validación de la ruta de certificación.

ANEXO B: PLANEAMIENTO SCRUM

PRODUCT BACKLOG

ID	MODULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DIAS)	ITE.
1	DSafe General	Investigar metodologías de desarrollo de software.	Alta	5	1
2	DSafe General	Evaluar metodologías de desarrollo de software y herramientas relacionadas.	Alta	5	1
3	DSafe General	Desarrollar marco teórico sobre la seguridad documental.	Alta	15	2
4	DSafe General	Investigar sobre técnicas y algoritmos de encriptación.	Alta	5	2
5	DSafe General	Desarrollar el análisis del sistema.	Alta	5	3
6	DSafe General	Desarrollar el diseño del sistema.	Alta	5	3
7	DSafe Usuarios	Iniciar sesión de usuario.	Media	1	4
8	DSafe Usuarios	Registrar nuevo usuario.	Alta	2	4
9	DSafe Usuarios	Ver perfil de usuario.	Baja	1	4
10	DSafe Usuarios	Cerrar sesión de usuario.	Baja	1	4
11	DSafe Contactos	Listar contactos registrados.	Baja	1	5
12	DSafe Contactos	Eliminar contacto registrado.	Baja	1	5

13	DSafe Contactos	Buscar contacto registrado.	Alta	2	5
14	DSafe Contactos	Enviar solicitud de contacto.	Alta	3	5
15	DSafe Contactos	Aceptar solicitud de contacto.	Alta	2	5
16	DSafe Contactos	Rechazar solicitud de contacto.	Media	1	5
17	DSafe Documentos	Registrar nuevo documento.	Alta	5	6
18	DSafe Documentos	Firmar documento.	Alta	10	6
19	DSafe Documentos	Ver documento.	Alta	5	6
20	DSafe Documentos	Eliminar documento.	Baja	1	7
21	DSafe Documentos	Listar documentos firmados.	Media	1	7
22	DSafe Documentos	Listar documentos pendientes.	Media	1	7
23	DSafe Documentos	Listar documentos observados.	Media	1	7
24	DSafe Documentos	Listar usuarios firmantes.	Baja	1	7
25	DSafe Documentos	Listar usuarios observadores.	Baja	1	7
26	DSafe Documentos	Descargar documento.	Baja	1	7
27	DSafe Documentos	Descargar certificado.	Baja	1	7
28	DSafe Notificaciones	Registrar notificación a documento.	Alta	2	8
29	DSafe Notificaciones	Ver notificaciones de documento.	Media	1	8
30	DSafe Notificaciones	Ver notificaciones de usuario.	Media	1	8
31	DSafe Conexión	Establecer conexión segura SSL.	Alta	5	8

	SSL				
32	DSafe General	Elaboración de pruebas.	Alta	5	9
33	DSafe General	Elaboración de manual de usuario.	Media	5	9
34	DSafe General	Evaluación de métricas de funcionalidad.	Alta	5	10
35	DSafe General	Evaluación de métricas de fiabilidad.	Alta	5	10
36	DSafe General	Evaluación de métricas de usabilidad.	Alta	4	10
37	DSafe General	Evaluación de métricas de eficiencia.	Alta	3	10
38	DSafe General	Evaluación de métricas de fac. de mantenimiento.	Alta	2	10
39	DSafe General	Evaluación de métricas de facilidad de portabilidad.	Alta	1	10

DESARROLLO DE SPRINTS

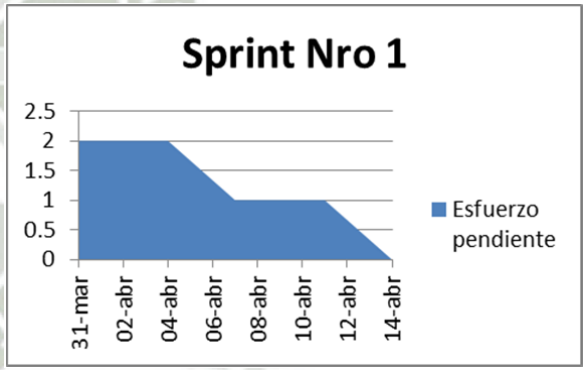
SPRINT NRO 1

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
1	DSafe General	Investigar metodologías de desarrollo de software.	Alta	5	1
2	DSafe General	Evaluar metodologías de desarrollo de software y herramientas relacionadas.	Alta	5	1

Burn down chart

SPRINT NRO 1	31-mar	02-abr	04-abr	07-abr	09-abr	11-abr	14-abr
Tareas pendientes	2	2	2	1	1	1	0



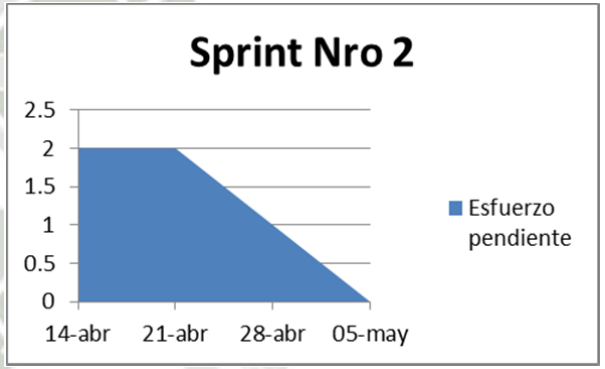
SPRINT NRO 2

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
3	DSafe General	Desarrollar marco teórico sobre la seguridad documental.	Alta	15	2
4	DSafe General	Investigar sobre técnicas y algoritmos de encriptación.	Alta	5	2

Burn down chart

SPRINT NRO 2	14-abr	21-abr	28-abr	05-may
Tareas pendientes	2	2	1	0



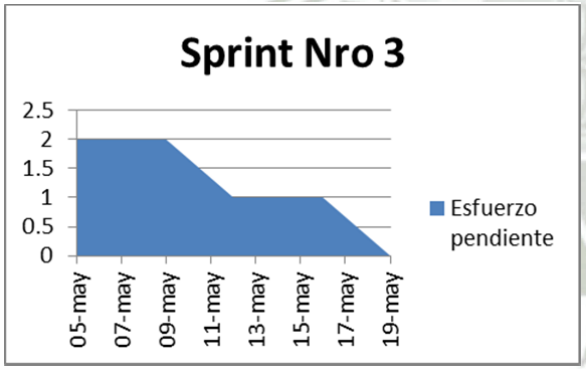
SPRINT NRO 3

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
5	DSafe General	Desarrollar el análisis del sistema.	Alta	5	3
6	DSafe General	Desarrollar el diseño del sistema.	Alta	5	3

Burn down chart

SPRINT NRO 3	05-may	07-may	09-may	12-may	14-may	16-may	19-may
Tareas pendientes	2	2	2	1	1	1	0



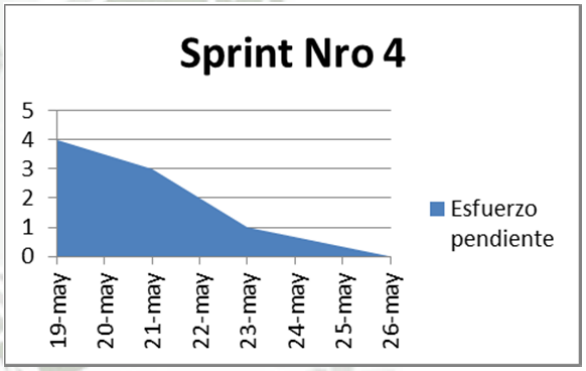
SPRINT NRO 4

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
7	DSafe Usuarios	Iniciar sesión de usuario.	Media	1	4
8	DSafe Usuarios	Registrar nuevo usuario.	Alta	2	4
9	DSafe Usuarios	Ver perfil de usuario.	Baja	1	4
10	DSafe Usuarios	Cerrar sesión de usuario.	Baja	1	4

Burn down chart

SPRINT NRO 4	19-may	21-may	23-may	26-may
Tareas pendientes	4	3	1	0



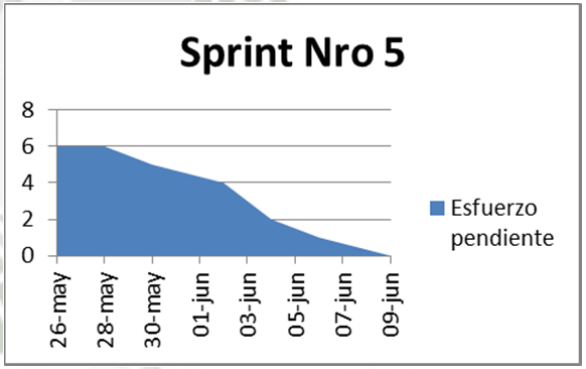
SPRINT NRO 5

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
11	DSafe Contactos	Listar contactos registrados.	Baja	1	5
12	DSafe Contactos	Eliminar contacto registrado.	Baja	1	5
13	DSafe Contactos	Buscar contacto registrado.	Alta	2	5
14	DSafe Contactos	Enviar solicitud de contacto.	Alta	3	5
15	DSafe Contactos	Aceptar solicitud de contacto.	Alta	2	5
16	DSafe Contactos	Rechazar solicitud de contacto.	Media	1	5

Burn down chart

SPRINT NRO 5	26-may	28-may	30-may	02-jun	04-jun	06-jun	09-jun
Tareas pendientes	6	6	5	4	2	1	0



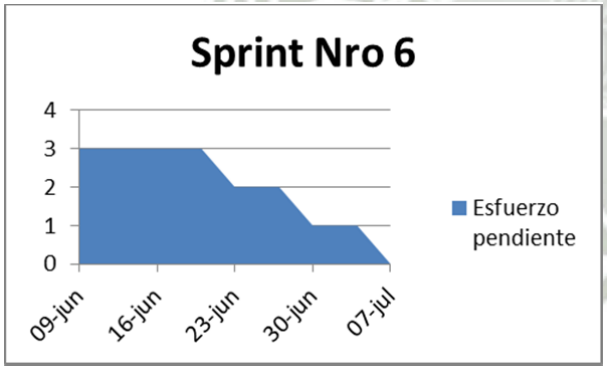
SPRINT NRO 6

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
17	DSafe Documentos	Registrar nuevo documento.	Alta	5	6
18	DSafe Documentos	Firmar documento.	Alta	10	6
19	DSafe Documentos	Ver documento.	Alta	5	6

Burn down chart

SPRINT NRO 6	09-jun	13-jun	16-jun	20-jun	23-jun	27-jun	30-jun	04-jul	07-jul
Tareas pendientes	3	3	3	3	2	2	1	1	0



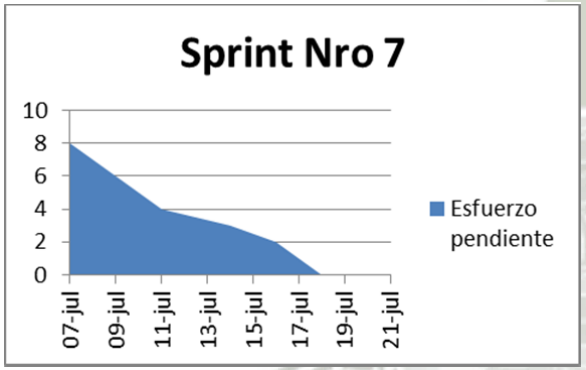
SPRINT NRO 7

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
20	DSafe Documentos	Eliminar documento.	Baja	1	7
21	DSafe Documentos	Listar documentos firmados.	Media	1	7
22	DSafe Documentos	Listar documentos pendientes.	Media	1	7
23	DSafe Documentos	Listar documentos observados.	Media	1	7
24	DSafe Documentos	Listar usuarios firmantes.	Baja	1	7
25	DSafe Documentos	Listar usuarios observadores.	Baja	1	7
26	DSafe Documentos	Descargar documento.	Baja	1	7
27	DSafe Documentos	Descargar certificado.	Baja	1	7

Burn down chart

SPRINT NRO 7	07-jul	09-jul	11-jul	14-jul	16-jul	18-jul	21-jul
Tareas pendientes	8	6	4	3	2	0	0



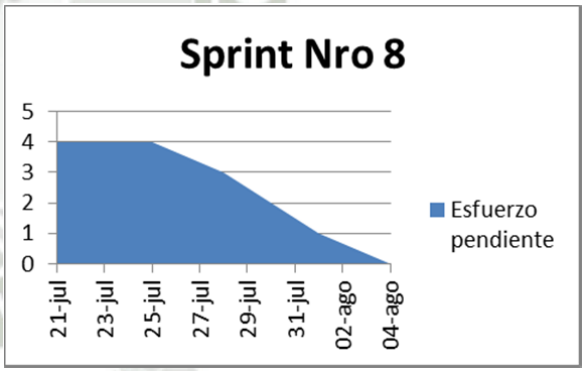
SPRINT NRO 8

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
28	DSafe Notificaciones	Registrar notificación a documento.	Alta	2	8
29	DSafe Notificaciones	Ver notificaciones de documento.	Media	1	8
30	DSafe Notificaciones	Ver notificaciones de usuario.	Media	1	8
31	DSafe Conexión SSL	Establecer conexión segura SSL.	Alta	5	8

Burn down chart

SPRINT NRO 8	21-jul	23-jul	25-jul	28-jul	30-jul	01-ago	04-ago
Tareas pendientes	4	4	4	3	2	1	0



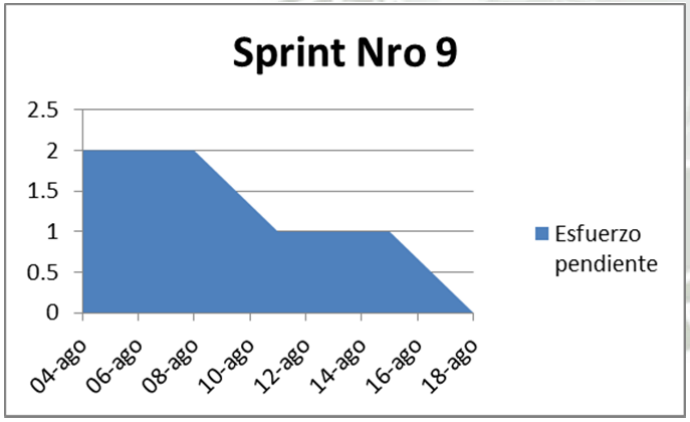
SPRINT NRO 9

Backlog

ID	MÓDULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DÍAS)	SPRINT
32	DSafe General	Elaboración de pruebas.	Alta	5	9
33	DSafe General	Elaboración de manual de usuario.	Media	5	9

Burn down chart

SPRINT NRO 9	04-ago	06-ago	08-ago	11-ago	13-ago	15-ago	18-ago
Tareas pendientes	2	2	2	1	1	1	0



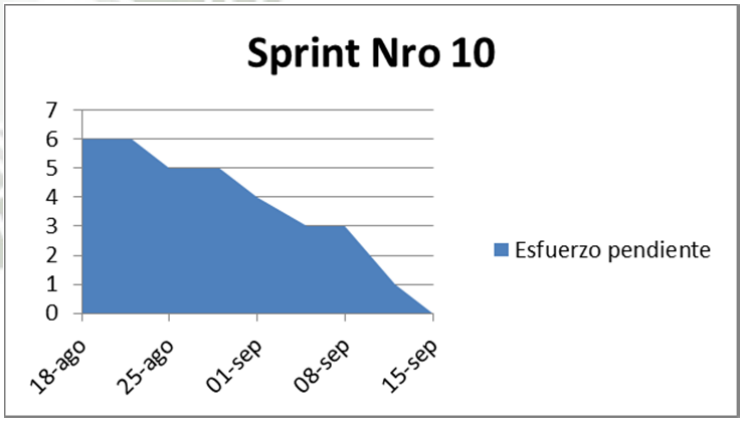
SPRINT NRO 10

Backlog

ID	MODULO	DESCRIPCIÓN	PRIORIDAD	ESTIMACION (DIAS)	ITE.
34	DSafe General	Evaluación de métricas de funcionalidad.	Alta	5	10
35	DSafe General	Evaluación de métricas de fiabilidad.	Alta	5	10
36	DSafe General	Evaluación de métricas de usabilidad.	Alta	4	10
37	DSafe General	Evaluación de métricas de eficiencia.	Alta	3	10
38	DSafe General	Evaluación de métricas de fac. de mantenimiento.	Alta	2	10
39	DSafe General	Evaluación de métricas de facilidad de portabilidad.	Alta	1	10

Burn down chart

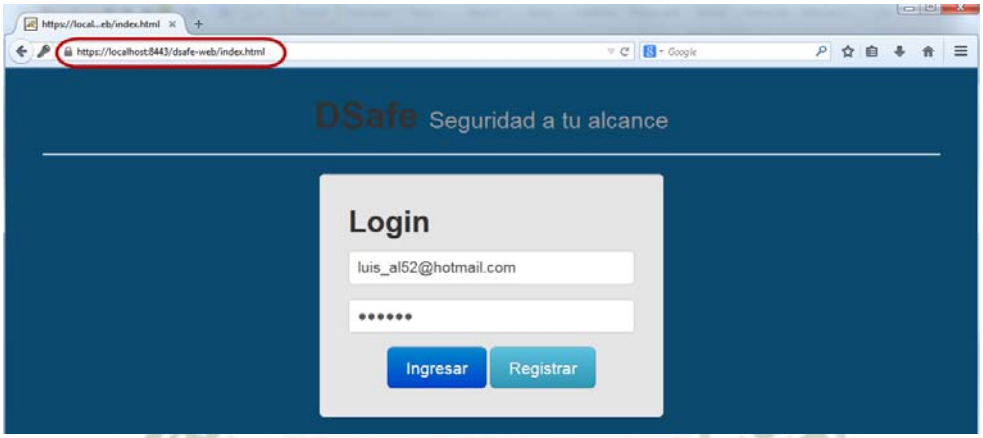
SPRINT NRO 10	18-ago	22-ago	25-ago	29-ago	01-sep	05-sep	08-sep	12-sep	15-sep
Tareas pendientes	6	6	5	5	4	3	3	1	0



ANEXO C: MANUAL DE USUARIO

CONEXIÓN SSL

Todas las operaciones realizadas en el sistema se realizan bajo un canal seguro SSL, el cual protegerá la privacidad del usuario y su respectiva clave privada usada al momento de firmar los documentos digitales.

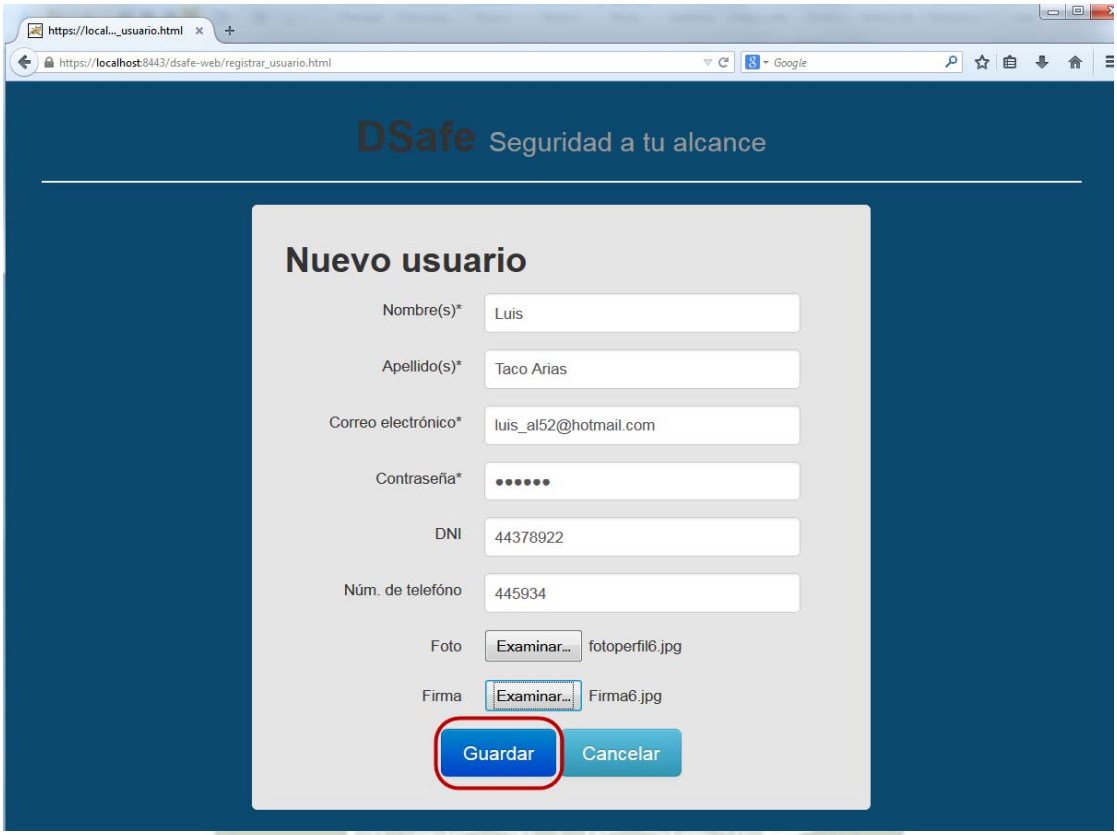


CREACIÓN DE CUENTA DE USUARIO

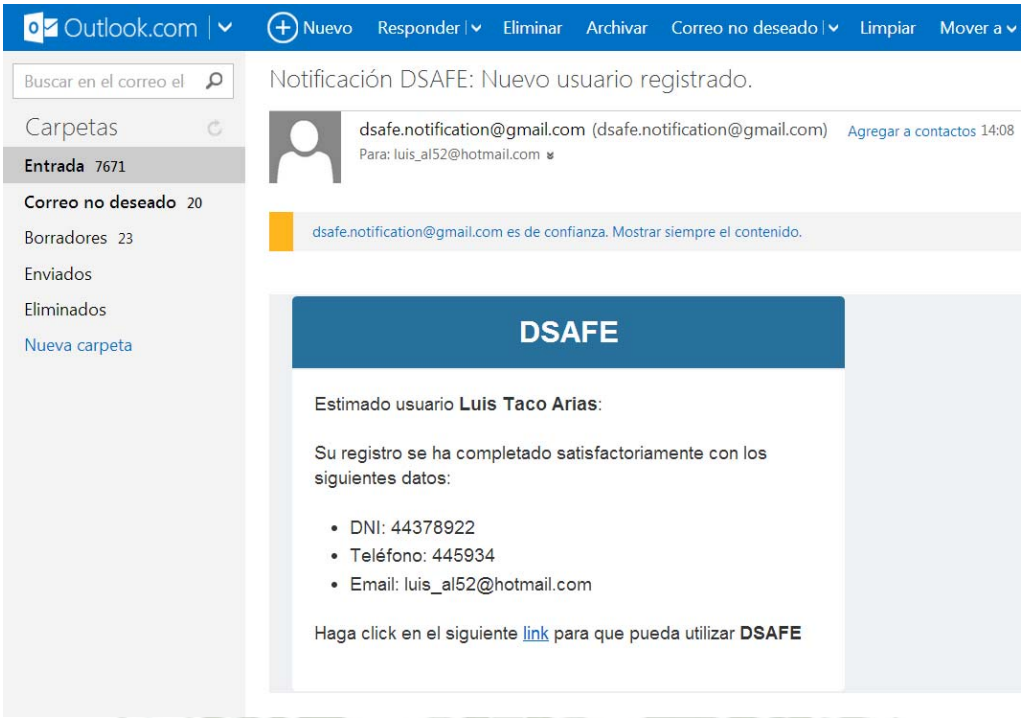
Para crear cuenta de usuario:

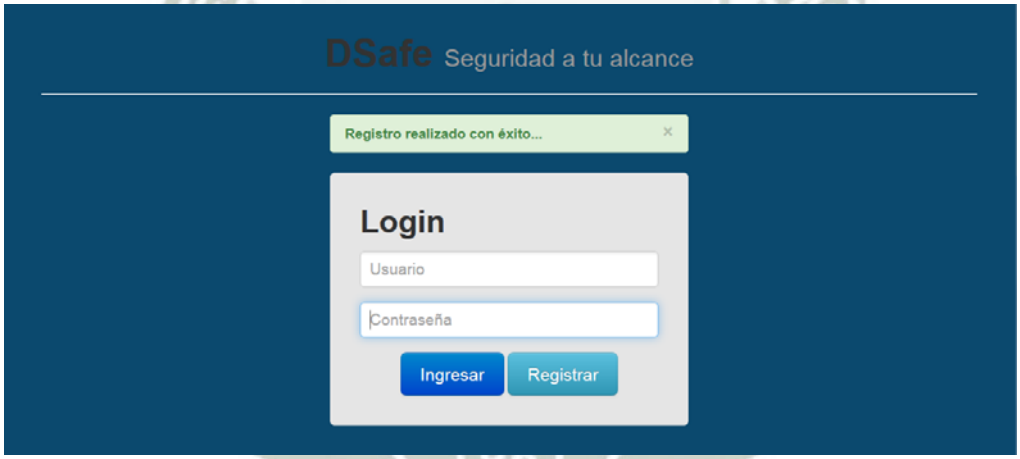
1. En la pantalla inicial del sistema presionar el botón “Registrar”.

2. Ingresar los datos del usuario junto con una foto e imagen de firma, dichas imágenes aparecerán en la pantalla de verificación cuando el usuario firme documentos, luego presionar “Guardar”.



3. Después de registrar un usuario el sistema regresa a la pantalla inicial indicando el mensaje de “Registro realizado con éxito”, también se envía un mail al correo de usuario confirmando los datos registrados.

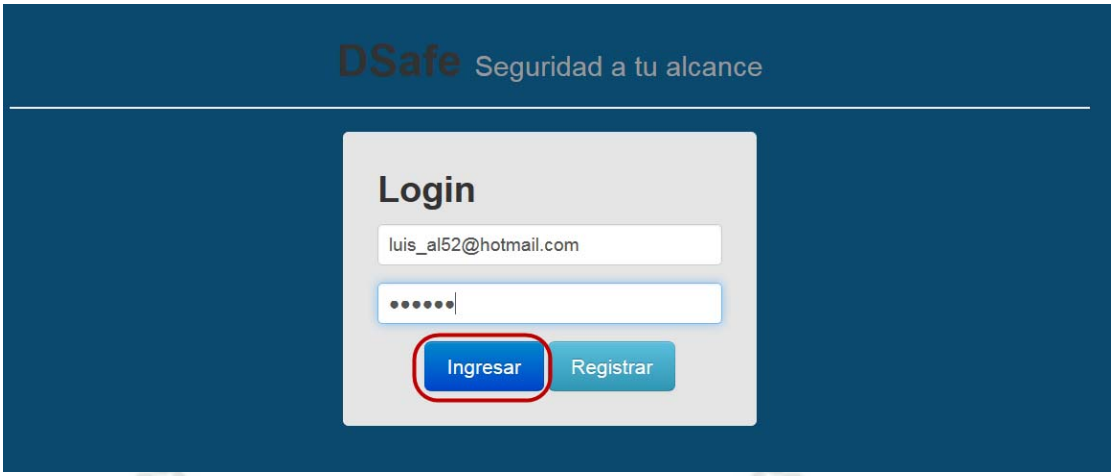




LOGIN

Para loguear en el sistema

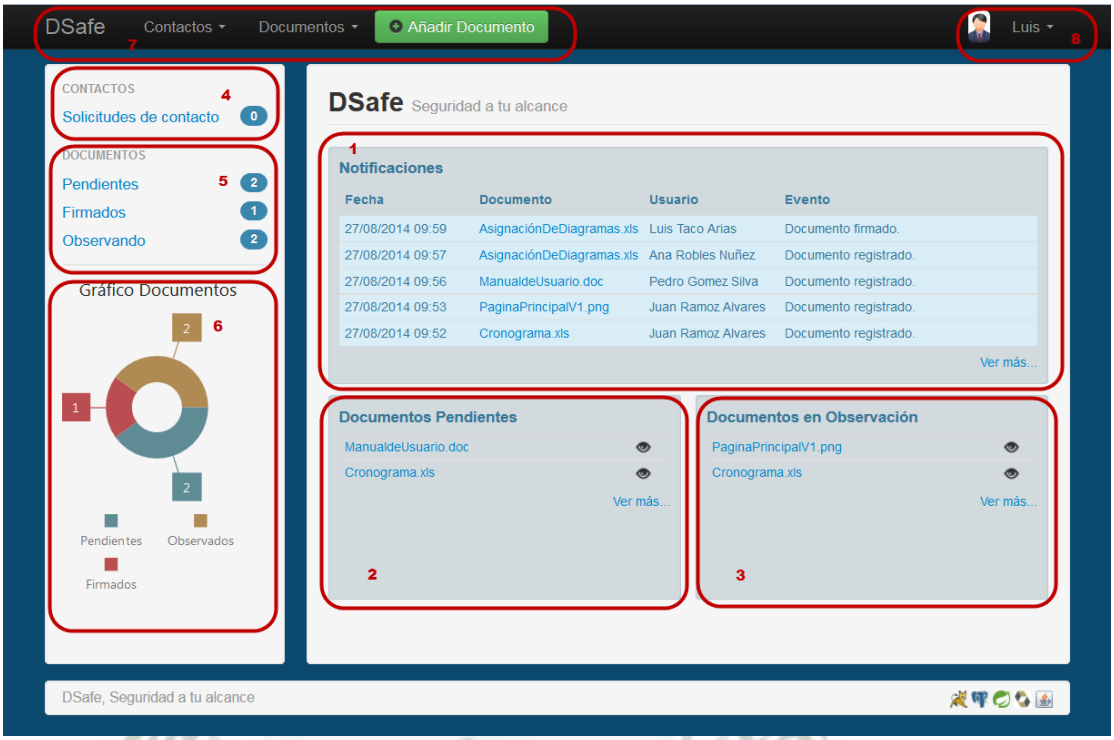
1. Ingresar el usuario (mail registrado) y contraseña en los campos correspondientes.
2. Presionar “Ingresar”.



PANTALLA PRINCIPAL DEL SISTEMA

Una vez logueado se muestra la pantalla principal del sistema en la cual tenemos los siguientes elementos:

1. Notificaciones sobre acciones realizadas en los documentos en los cuales el usuario es firmante u observador.
2. Lista de documentos pendientes de la firma del usuario.
3. Lista de documentos en los cuales el usuario es observador.
4. Número de solicitidudes de contacto recibidas esperando aprobación, esto es un link que dará acceso directo a la interfaz para aceptar las solicitudes pendientes.
5. Número de documentos: pendientes de firma, documentos donde el usuario es observador, documentos firmados, cada uno de estos números es un link que dará acceso directo a la interfaz de listado de documentos correspondiente.
6. Pie chart de las cantidades de documentos donde el usuario es observador, firmante, o requiere de su firma.
7. Barra de menú del sistema consiste de:
   7.1. Contactos: Tiene las opciones para ver contactos del usuario, solicitudes pendientes y añadir contactos.
   7.2. Documentos: Tienes las opciones para ver documentos subidos por el usuario y añadir un nuevo documento.
   7.3. Añadir Documento: botón que permite añadir un documento
8. Sesión del usuario, permite ver datos del perfil de usuario logueado y cerrar la sesión del sistema.

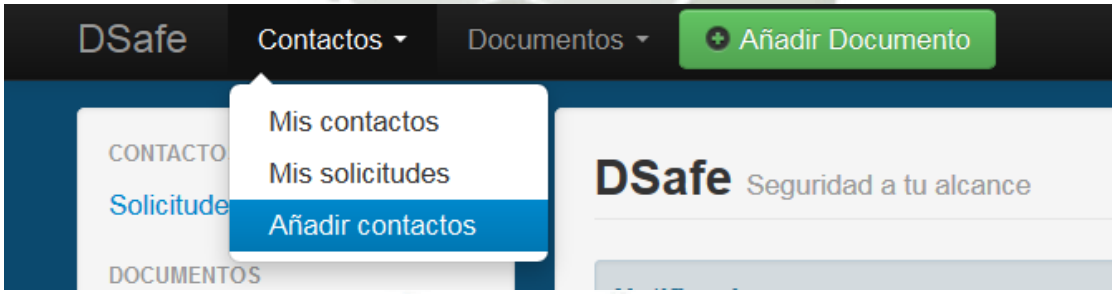


CONTACTOS

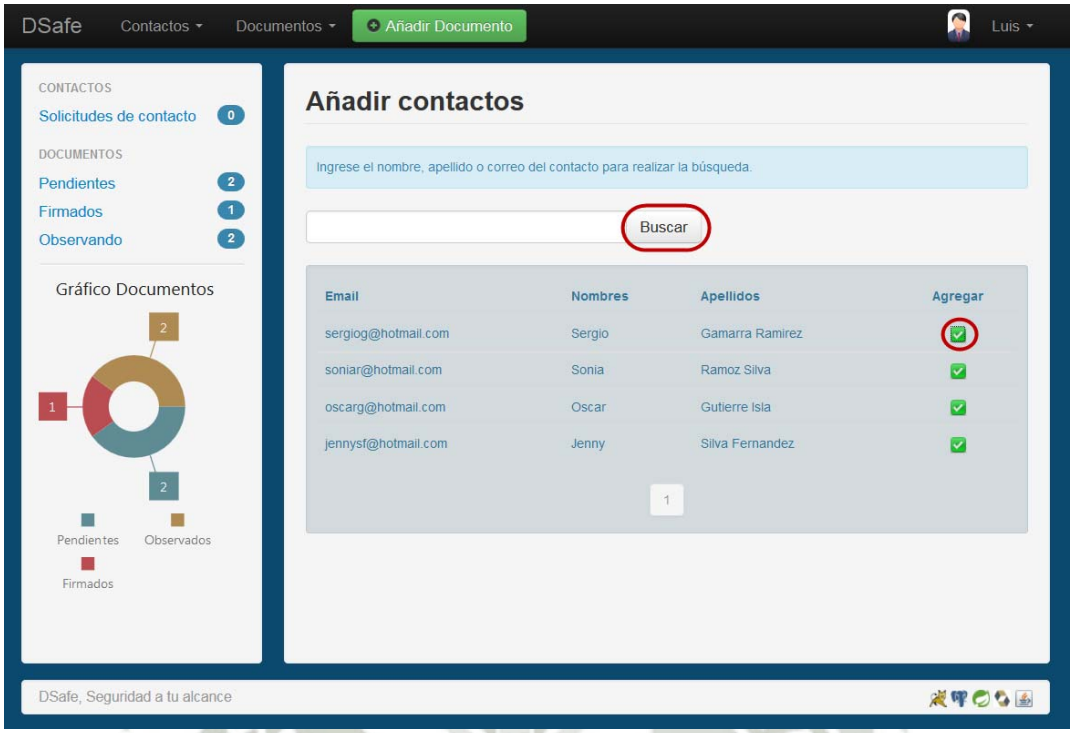
AÑADIR CONTACTOS

Para añadir un contacto:

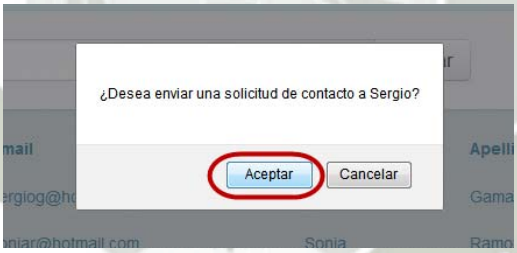
1. Ir al menú Contactos y seleccionar la opción “Añadir contactos”.



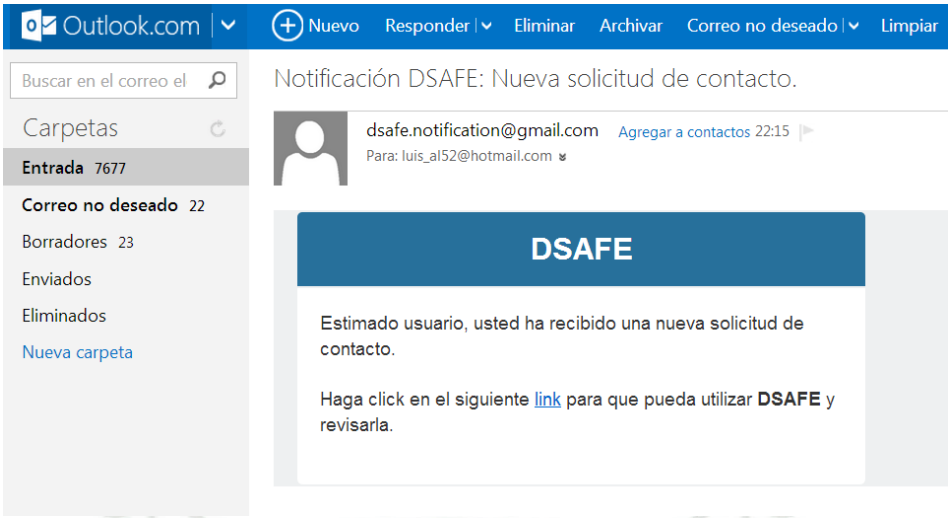
2. pantalla desde la cual podemos buscar los contactos que deseamos añadir, finalmente agregar el usuario presionando el respectivo ícono ubicado en la columna “Agregar”.



3. Aceptamos el mensaje de confirmación para añadir el contacto.



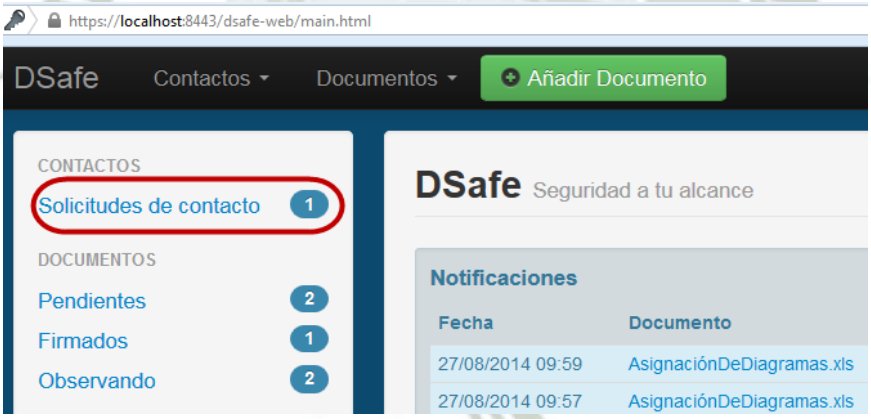
4. Adicionalmente se envía un mail a la persona que ha sido agregada para que esta pueda aceptar el contacto.



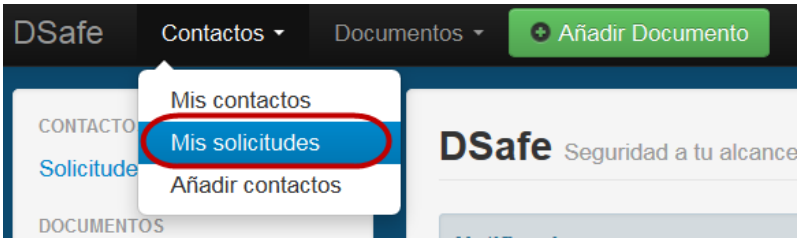
CONFIRMAR CONTACTO

Para aceptar un contacto:

1. Presionar el acceso rápido en la pantalla inicial del sistema que dice “Solicitudes de contacto”, el número a la derecha indica cuantas solicitudes tenemos pendientes de confirmación

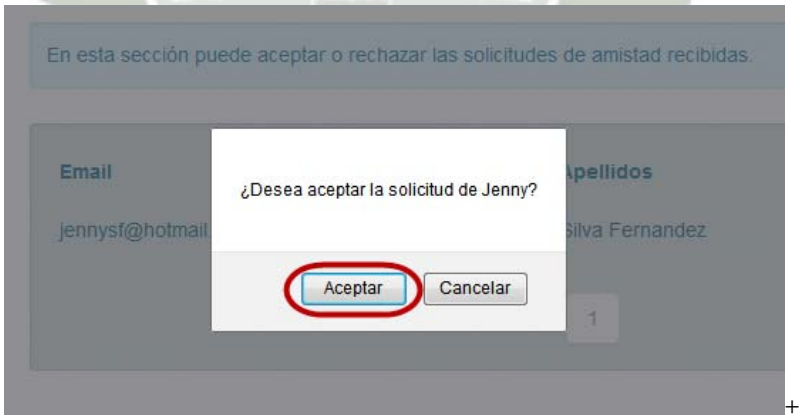


2. También se pueden ver las solicitudes usando el menú “Contactos” y seleccionando la opción “Mis solicitudes”.



3. Se muestra el listado de las solicitudes de contactos recibidas, las cuales podemos aceptar o rechazar, para este ejemplo optaremos por "Aceptar" presionando el ícono de color verde.

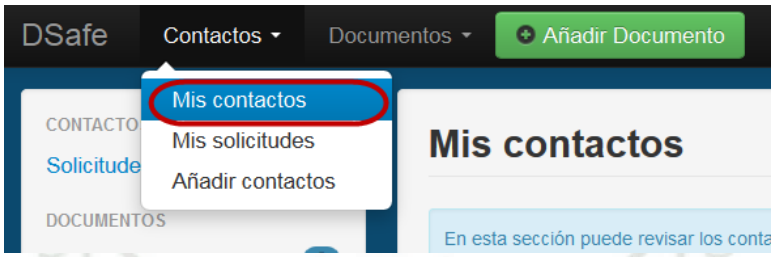
4. Aceptar el mensaje de confirmación para añadir la solicitud de contacto.



VER MIS CONTACTOS

Para ver los contactos que hemos añadido:

1. Seleccionar la opción “Mis contactos”, seguidamente se mostrará la lista de contactos.



ELIMINAR UN CONTACTO

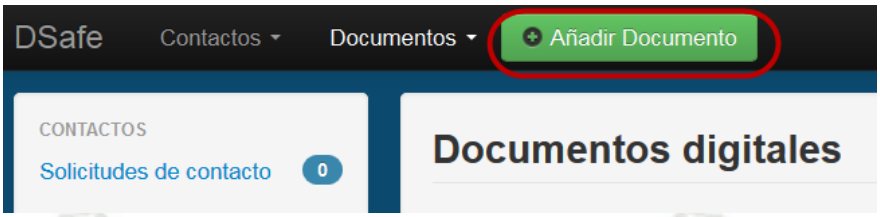
Para eliminar un contacto selecciona el ícono eliminar mostrado en el listado de “Mis Contactos”

DOCUMENTOS

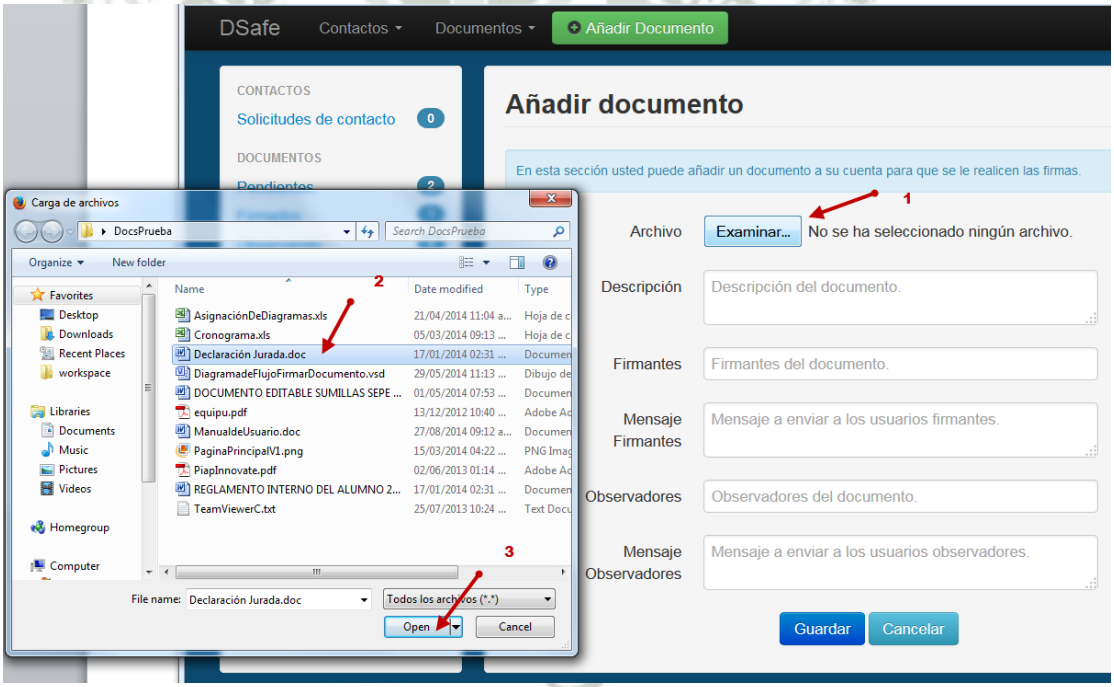
AÑADIR DOCUMENTO

Para añadir documentos:

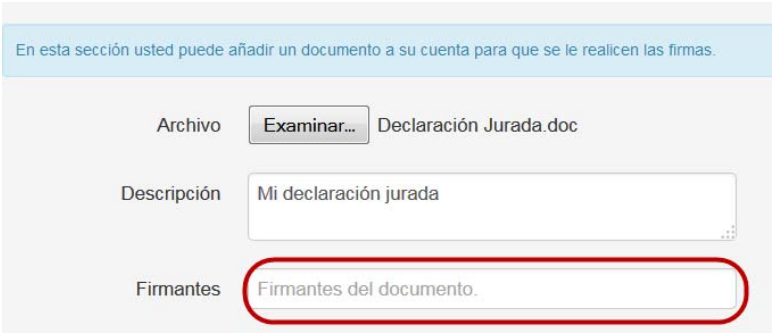
1. Presionar el botón “Añadir Documento”.

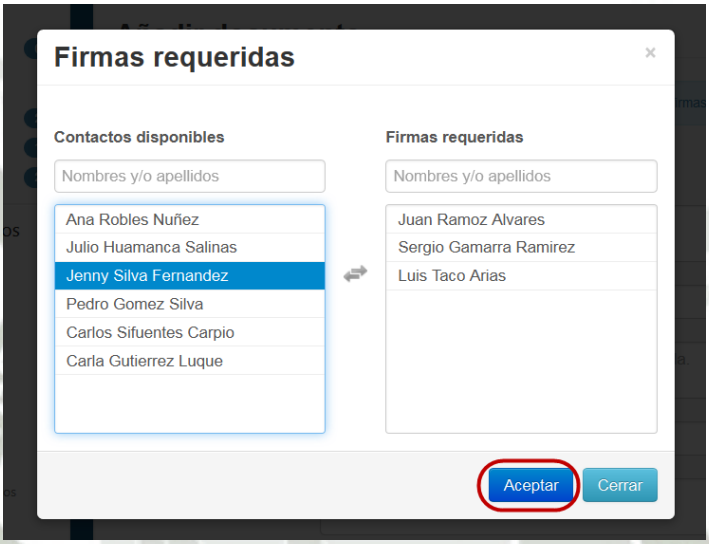


2. Seleccionar el archivo pulsando el botón “Examinar”, escoger el documento de la ventana emergente y presionar “Abrir (Open)”.

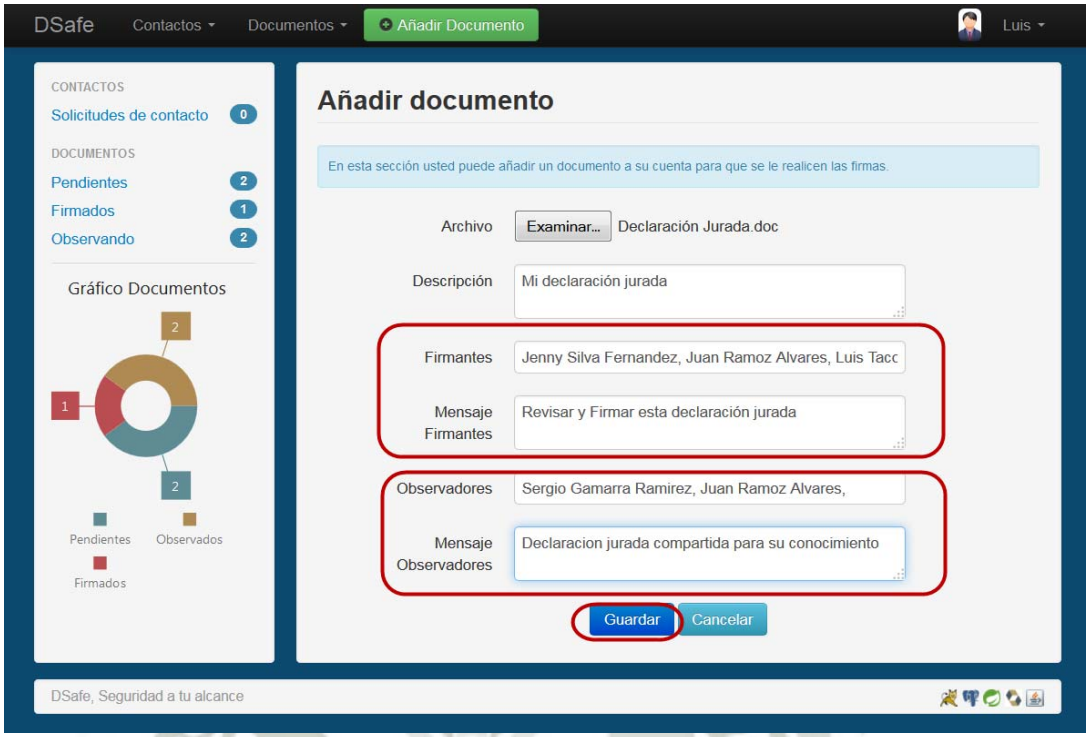


3. Ingresar la descripción y al momento de hacer click en el espacio de firmantes aparecerá una ventana para seleccionar las personas que firmarán los documentos, las personas que podremos seleccionar serán solamente aquellas que tengamos como contactos.

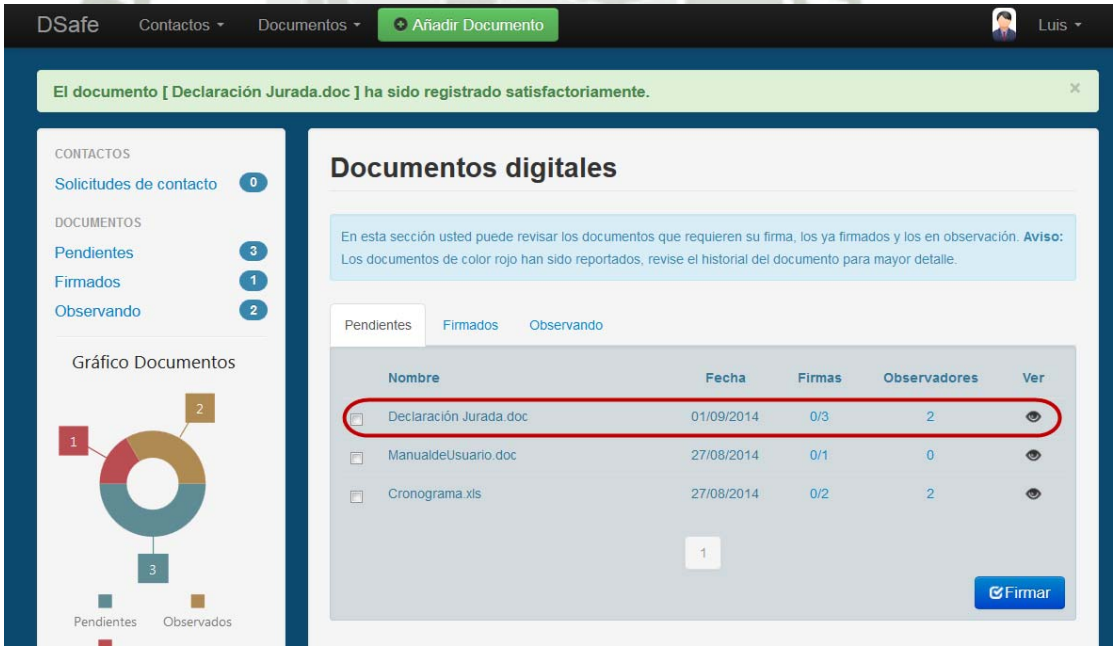




4. En la caja de texto “Mensaje Firmantes” ingresar el mensaje que será enviado a los correos de los firmantes del documento, seguir el mismo procedimiento de ingreso de firmantes para el ingreso de observadores, los observadores no tendrán que firmar el documento pero se les dará acceso para que puedan descargarlo y hacer la respectiva verificación de firmas.

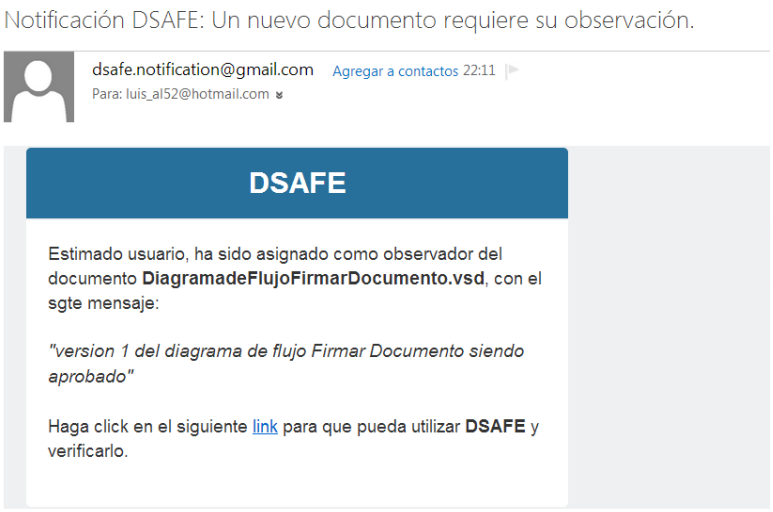


5. Finalmente presionar guardar y aparecerá el mensaje “El documento ** ha sido registrado satisfactoriamente”, este documento será añadido a las respectivas listas de los firmantes y observadores junto con los correos correspondientes.



6. Despúes de añadir un documento se envía un mail a los firmantes del documento.

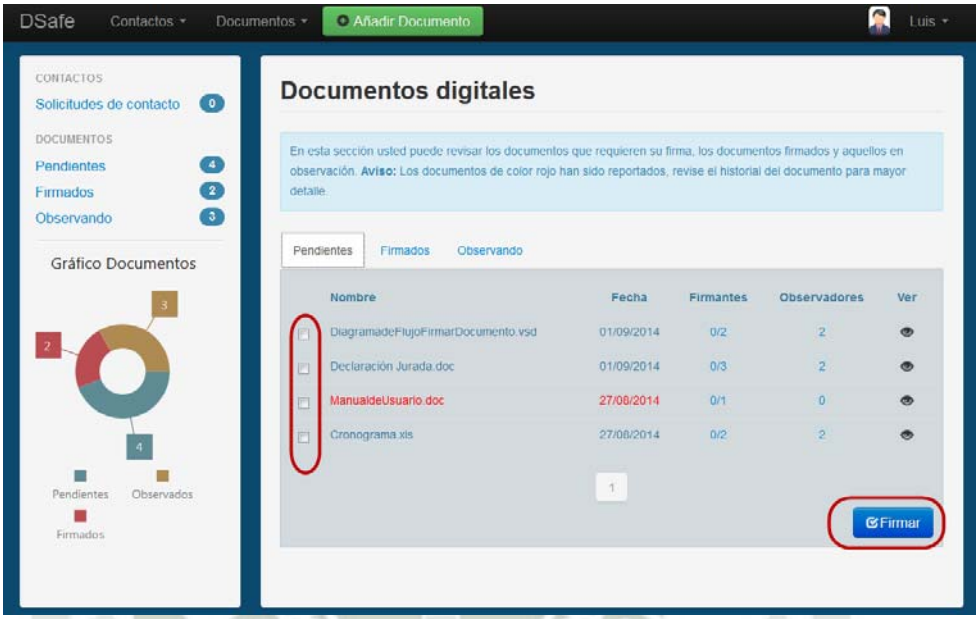
7. Despúes de añadir un documento se envía un mail a los Observadores del documento.



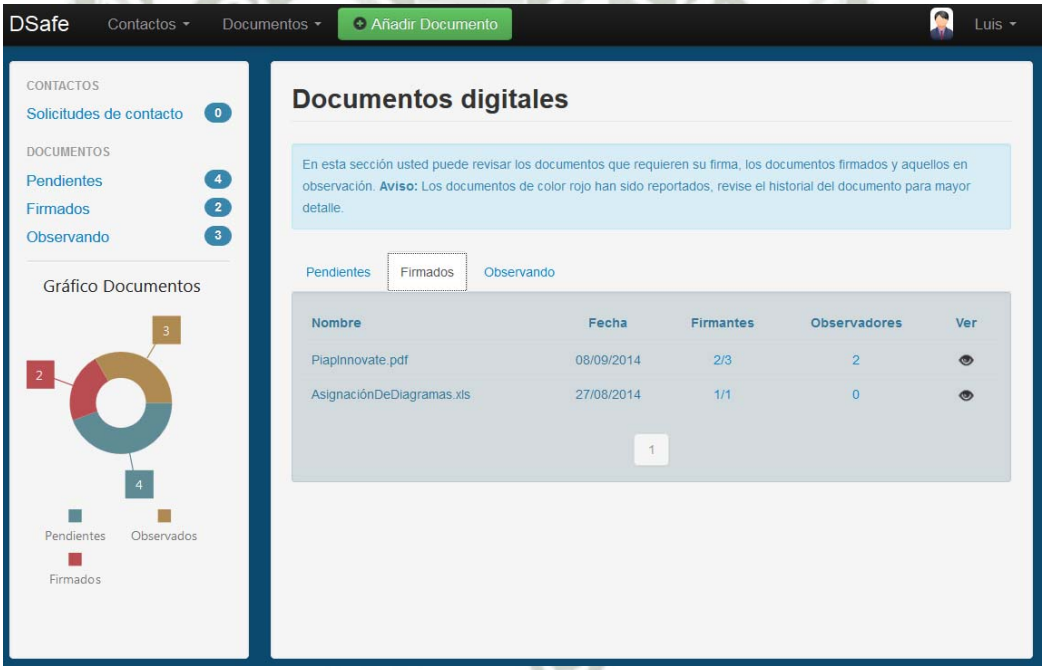
LISTADO DE DOCUMENTOS

El sistema se tiene 4 tipos de listados de documentos:

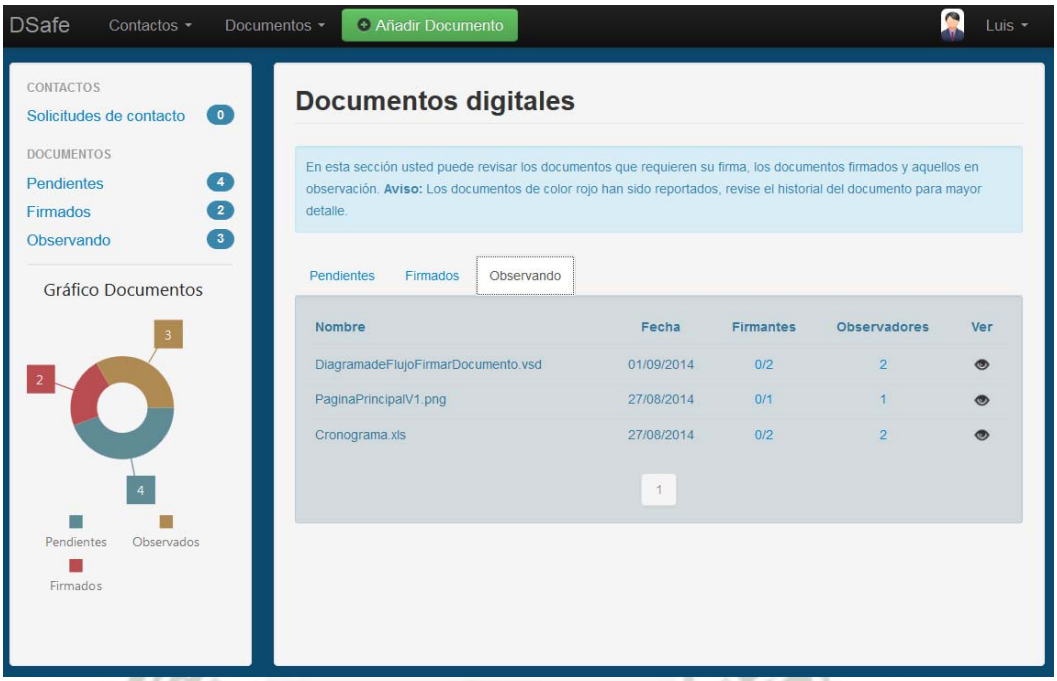
1. El primer tipo de listado son los documentos “Pendientes” de Firma, este listado contiene cajas para seleccionar los documentos y un botón para “Firmar” los ítems seleccionados.



2. El segundo tipo de listado son los documentos “Firmados” por el usuario.



3. El tercer tipo de listado son los documentos en los que el usario es asignado como observador.

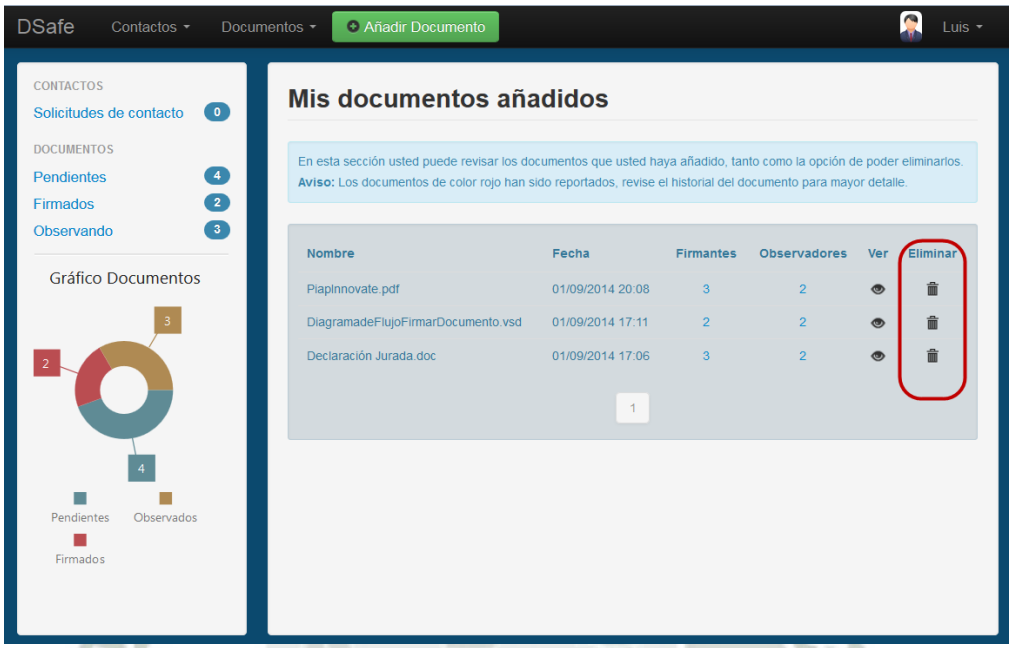


4. El cuarto y último tipo de listado son los documentos añadidos por el usuario, para acceder a este listado ir al menú “Documentos”, seleccionar la opción “Mis Documentos”.

ELIMINAR DOCUMENTOS

Para eliminar documentos:

1. Ir al listado de mis documentos
2. Presionar el ícono en la columna “Eliminar” del documento a efectuar esta acción.

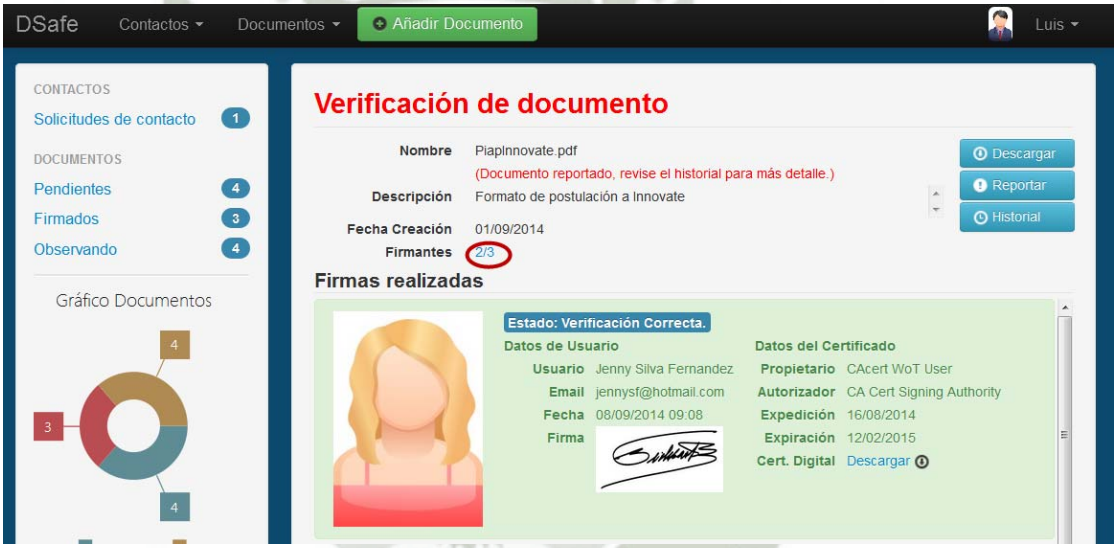


VER FIRMANTES DEL DOCUMENTO

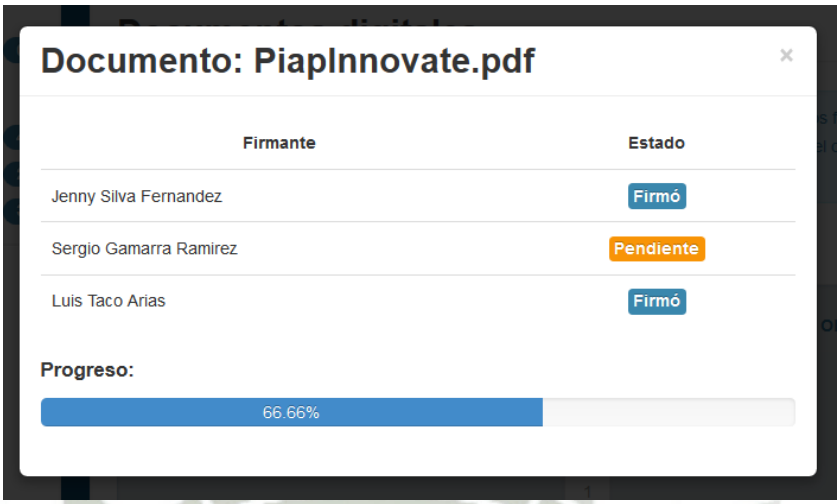
Para ver el listado de firmantes de un documento

1. Ir a un listado cualquiera de documentos y presionar el enlace ubicado en la columna “Firmantes”.

2. Otra opción es presionar el ícono ubicado en la columna “Ver” del listado de documentos, luego en la pantalla de verificación podremos presionar el link de firmantes.



3. En cualquiera de los 2 casos anteriores se muestra la ventana emergente con el listado de firmantes.



VER OBSERVADORES DEL DOCUMENTO

Para ver el listado de observadores de un documento.

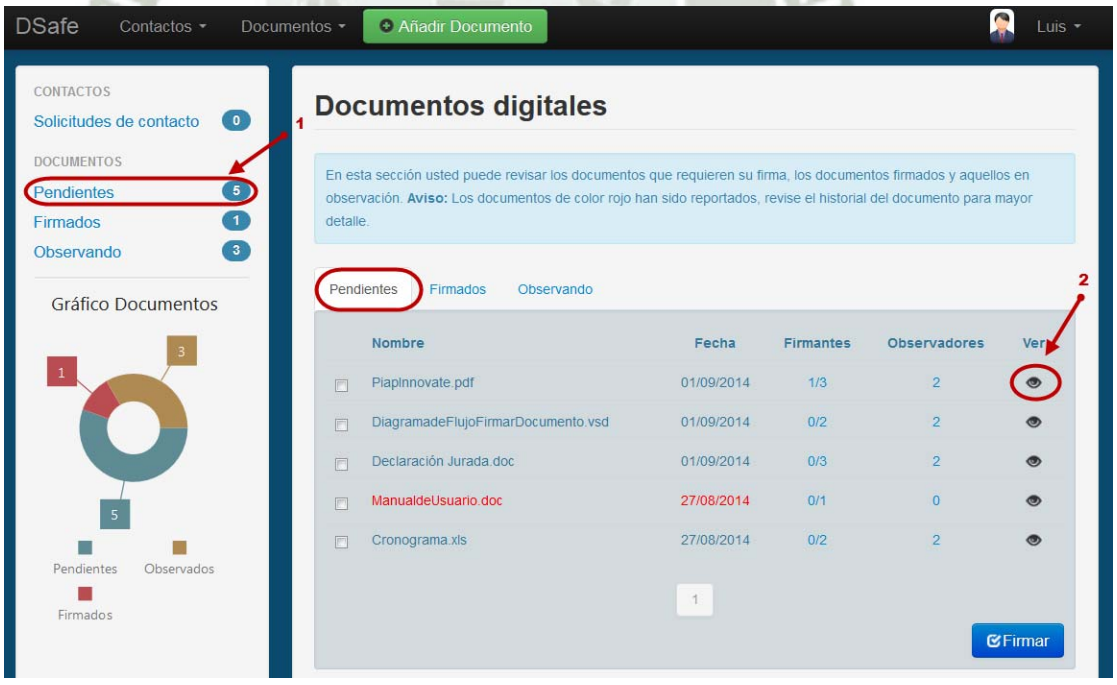
1. Ir a un listado cualquiera de documentos y presionar el enlace ubicado en la columna “Observadores”

2. Se muestra una ventana emergente con el listado de observadores del documento.

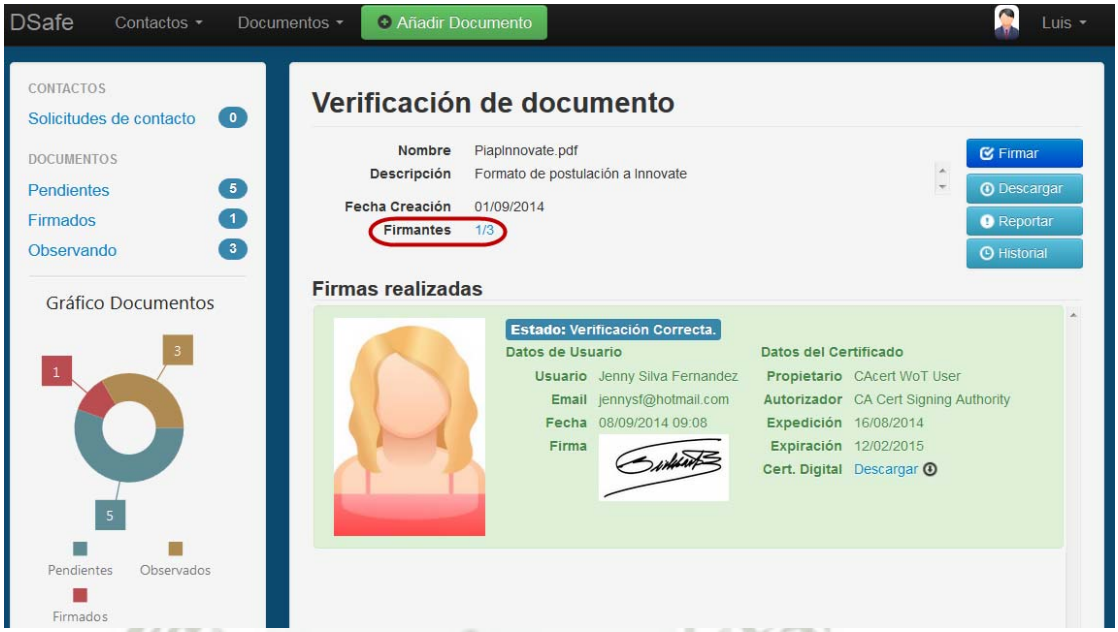
FIRMAR DOCUMENTO

Para firman un documento.

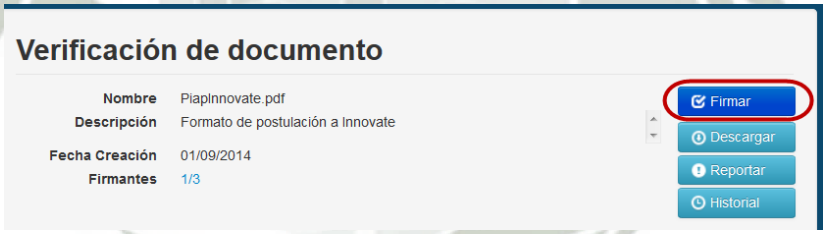
1. Para Firmar un Documento seleccionar la opción “Pendientes” en el menú izquierdo, esto mostrará un listado de los documentos pendientes de firma.
2. Luego hacer click en el ícono ubicado en la columna “Ver” del documento que se desea firmar.



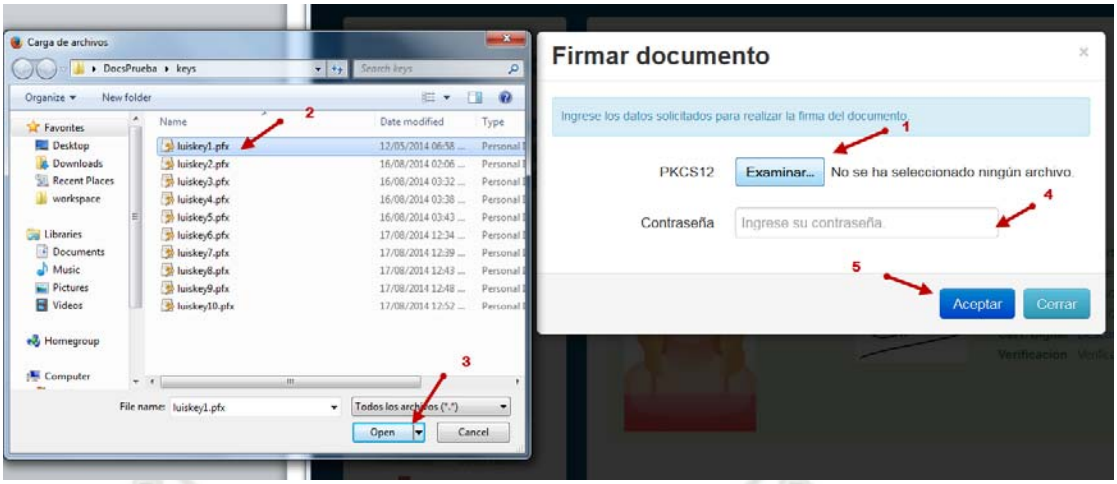
3. Se muestra la pantalla de verificación la cual da mayor detalle sobre el documento a Firmar, en la parte inferior se muestra los firmantes en caso los hubiera.



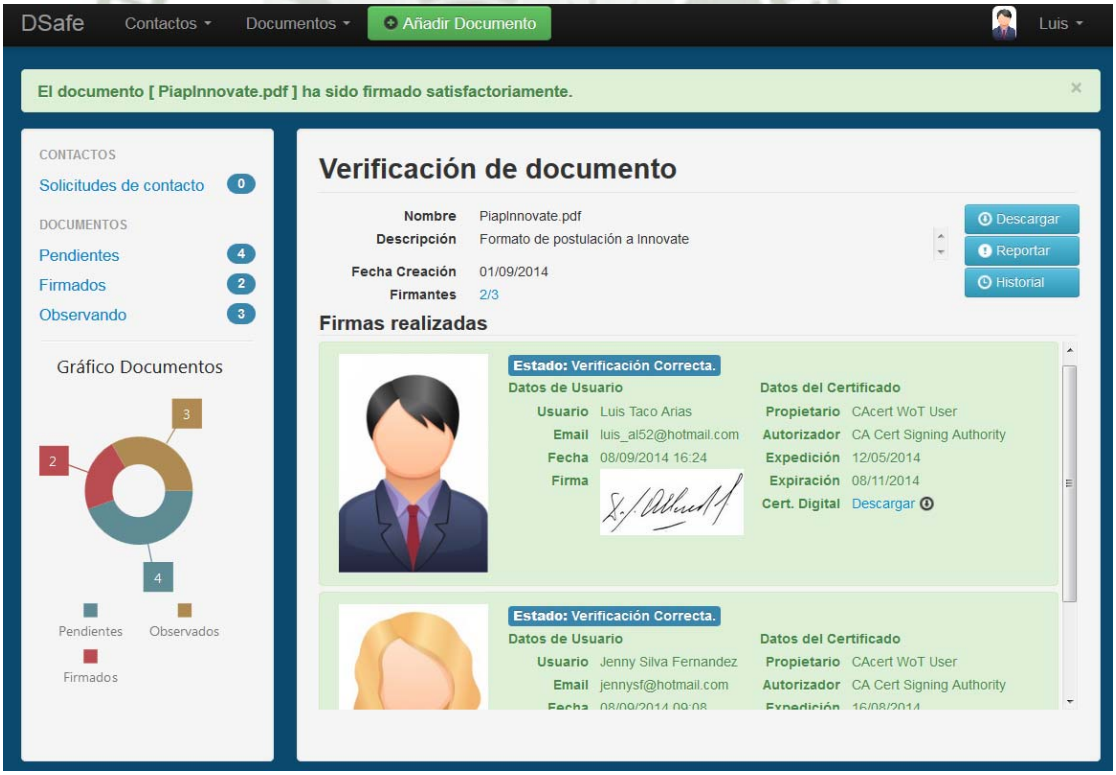
4. Presionar el botón “Firmar”.



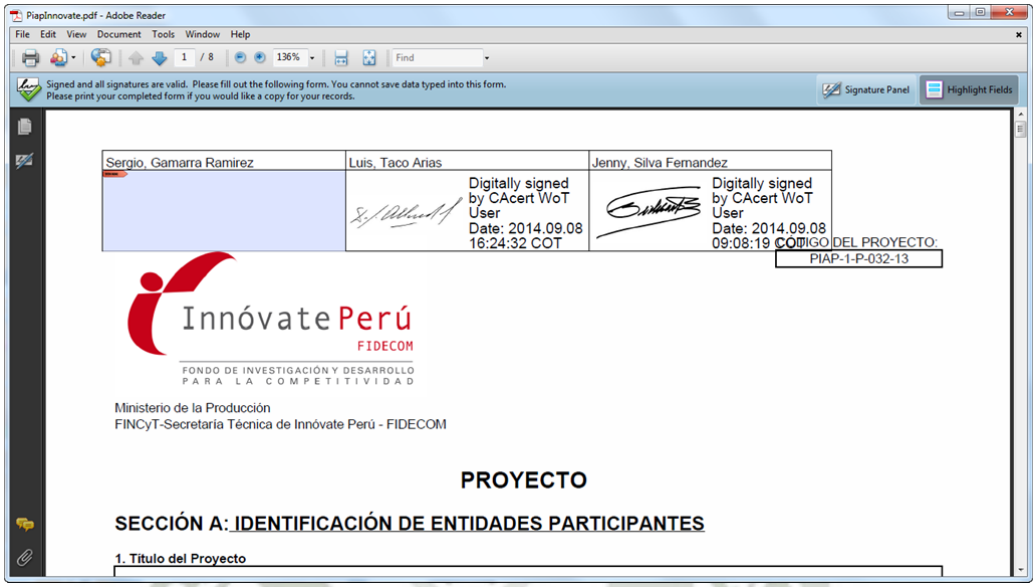
5. Se muestra una ventana emergente desde la cual podremos seleccionar nuestra llave privada e ingresar la contraseña.
6. Una vez ingresados los campos se presiona “Aceptar”.



7. Una vez realizados los pasos anteriores el sistema verifica la validez de la clave privada con el objetivo de descartar que el certificado haya sido revocado o sea caduco, si no hay problemas encontrados se procede con la Firma del documento, con lo cual se muestra un mensaje indicando que el documento ha sido firmado satisfactoriamente.



8. En el caso que el documento firmado sea un PDF, las firmas son añadidas en la parte superior del documento, y contarán con validaciones propias del estándar de firmas digitales de los documentos PDF

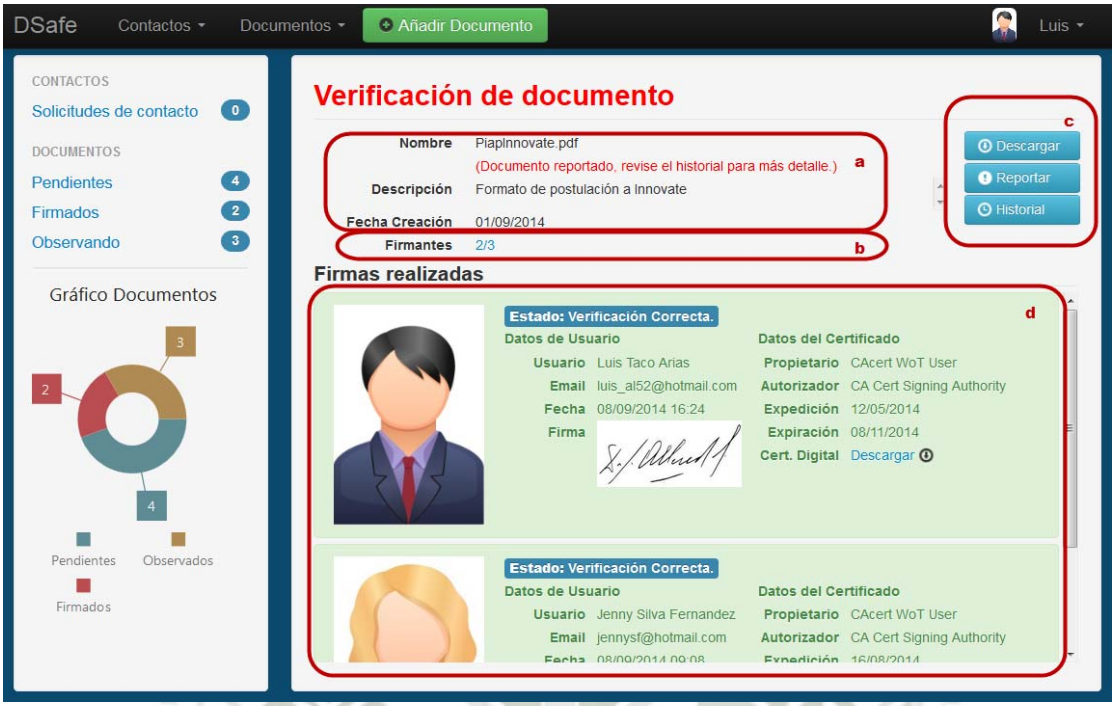


VERIFICAR DOCUMENTO

Para verificar un documento:

1. Ir a un listado de documentos.

2. Luego hacer click en el ícono ubicado en la columna “Ver” del documento que se desea firmar, si el documento a revisar contiene firmas el sistema realiza la verificación y muestra la siguiente información.
   a. Información del documento: (nombre, descripción, fecha de creación), también aparece el mensaje de documento reportado según sea el caso.
   b. Firmantes del documento: contiene un link para ver un listado de los firmantes del documento.
   c. Botones de función para: Descargar documento, reportar documento, ver historial, en caso de ser firmante aparecerá el botón para firmar el documento.
   d. Zona de firmas realizadas: aparecerán datos del firmantes, datos de su certificado (se podrá descargar el certificado del firmante presionando el link “Descargar” respectivo de cada firmante) e imágenes de sus retratos y rúbricas; la verificación se realiza por cada firmante, de ser correcto estos aparecerán en una zona de color verde, en caso de error en la verificación, la zona será de color rojo y se mostrará el mensaje indicando el problema en la verificación.

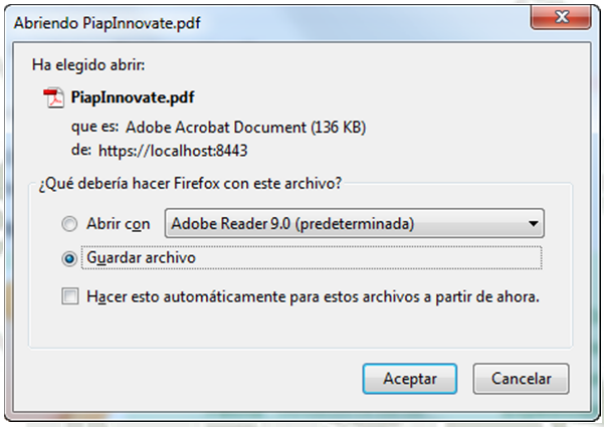


DESCARGAR DOCUMENTO

Para descargar un documento:

1. Ir a un listado de documentos y presionar el ícono ubicado en la columna “Ver” del documento a descargar.

2. En la pantalla de verificación presionar el botón “Descargar”.

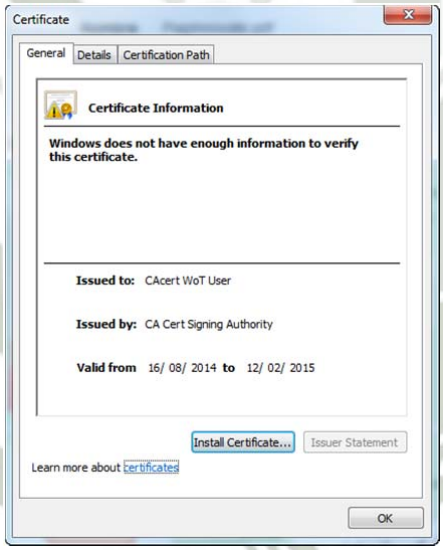


DESCARGAR CERTIFICADO DE UN FIRMANTE DEL DOCUMENTO.

Para descargar el certificado:

1. Ir a un listado de documentos y presionar el ícono ubicado en la columna “Ver” del documento revisar.

2. En la pantalla de verificación del documento ir a la sección de firmas realizadas y presionar el link “Descargar” ubicado en la columna “Datos del Certificado” del firmante correspondiente.



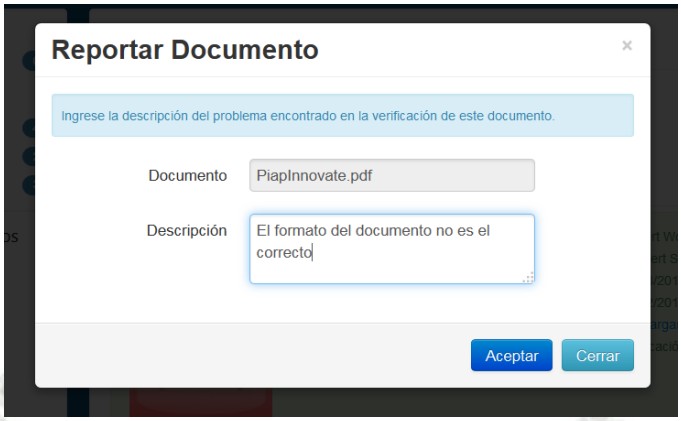
REPORTAR DOCUMENTO.

Para reportar un documento:

1. Ir a un listado de documentos y presionar el ícono ubicado en la columna “Ver” del documento revisar.

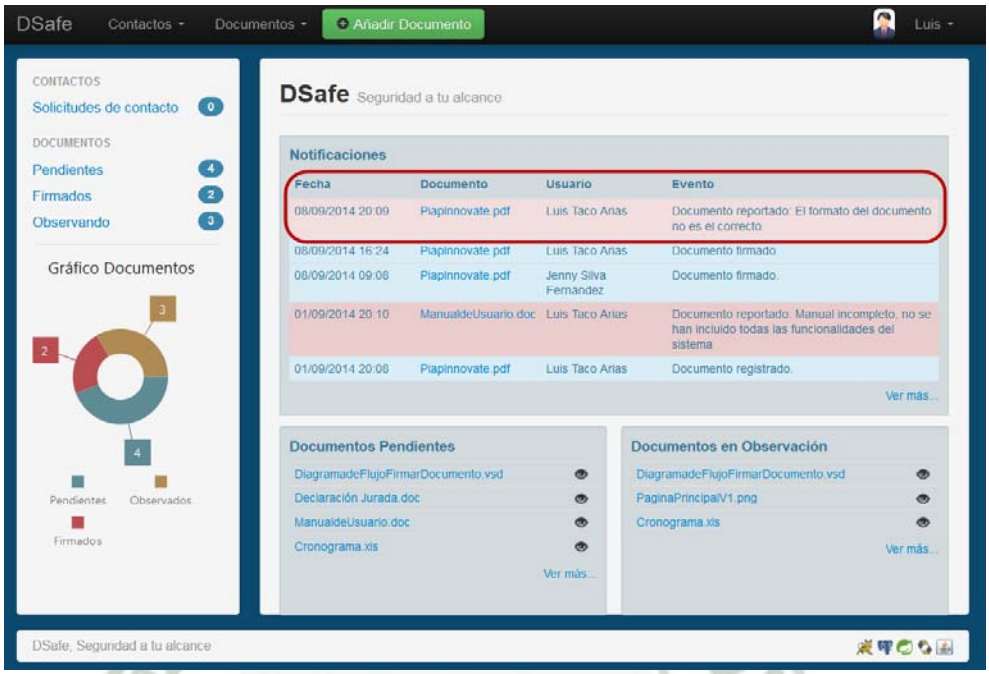
2. Presionar el botón “Reportar” ubicado en la pantalla de verificación del documento.

3. Ingresar la descripción de nuestra observación, y luego presionar “Aceptar”.



4. El título de la pagína cambia a el color rojo y se muestra un mensaje indicando que el documento ha sido reportado.

5. El reporte realizado se muestra en la pantalla de notificaciones e historial del documento.



6. Aquellos documentos reportados aparecen de color rojo en los listados de los documentos.

VER HISTORIAL DEL DOCUMENTO

Para ver el historial de un documento:

1. Ir a un listado de documentos y presionar el ícono ubicado en la columna “Ver” del documento revisar.

2. Presionar el botón “Historial” ubicado en la pantalla de verificación del documento.

3. Se muestra una ventana emergente que muestra el historial de cambios del documento ordenados por fecha en forma descendente.