



INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior de Cómputo

ESCOM

*Trabajo Terminal*

**“Protocolo criptográfico para el almacenamiento  
sin duplicados en la nube, resistente a ataques  
por fuerza bruta.”**

*2016-B045*

*Presentan*

**Eder Jonathan Aguirre Cruz**

**Diana Leslie González Olivier**

**Jhonatan Saulés Cortés**

*Directora*

**Dra. Sandra Díaz Santiago**

INSTITUTO POLITÉCNICO NACIONAL



ESCOM

**Mayo 2017**

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Contexto . . . . .	1
1.2. Problemática . . . . .	2
1.3. Estado del Arte . . . . .	4
1.4. Justificación . . . . .	5
1.5. Solución propuesta . . . . .	6
1.6. Objetivos . . . . .	7
1.6.1. Objetivo General . . . . .	7
1.6.2. Objetivos Específicos . . . . .	8
<b>2. Marco Teórico</b>	<b>9</b>
2.1. Definiciones. . . . .	9
2.1.1. Servicios criptográficos. . . . .	9
2.2. Ataques a servicios criptográficos. . . . .	10
2.2.1. Ataques por fuerza bruta . . . . .	11
2.3. Criptografía Simétrica. . . . .	11
2.4. Criptografía Asimétrica. . . . .	13
2.5. Cifrador por bloques. . . . .	14
2.6. Cómputo Nube. . . . .	15
<b>3. Protocolo Dupless</b>	<b>18</b>
3.1. Firma digital . . . . .	18
3.2. RSA . . . . .	19
3.3. Firmas a ciegas. . . . .	20
3.4. Firmas a ciegas con RSA. . . . .	21
3.5. Funciones Hash. . . . .	21
<b>4. Análisis</b>	<b>23</b>
4.1. Estudio de Factibilidad . . . . .	23
4.1.1. Factibilidad técnica . . . . .	23
4.1.2. Factibilidad operativa . . . . .	28
4.1.3. Facctibilidad económica . . . . .	28
4.1.4. Análisis de riesgos . . . . .	29
Escalas de impacto de un riesgo . . . . .	29
4.2. Arquitectura del sistema. . . . .	31
4.3. Descripcion de procesos . . . . .	35
4.3.1. Descripción del proceso subir archivo. . . . .	35
Participantes . . . . .	36
4.3.2. Descripción del proceso Descargar archivo. . . . .	37
Participantes . . . . .	37
4.3.3. Descripción del proceso eliminar archivo. . . . .	38

Participantes . . . . .	39
4.4. Modelo de entidades. . . . .	40
4.4.1. Diagrama de Entidad Relación. . . . .	40
4.4.2. Diagrama de clases. . . . .	41
4.5. Requerimientos Funcionales. . . . .	42
4.6. Requerimientos No Funcionales . . . . .	43
4.7. Reglas de Negocio . . . . .	45
<b>5. Diseño</b>	<b>47</b>
5.1. Especificación de Plataforma . . . . .	47
5.2. Casos de Uso . . . . .	48
5.2.1. CUSLL1 Generar las llaves del servidor de llaves . . . . .	49
Descripción completa . . . . .	49
Atributos importantes . . . . .	49
Trayectorias del Caso de Uso . . . . .	49
5.2.2. CUSLL2 Generar firma ciega (y). . . . .	51
Descripción completa . . . . .	51
Atributos importantes . . . . .	51
Trayectorias del Caso de Uso . . . . .	51
5.2.3. CUN3 Almacenar archivo cifrado . . . . .	52
Descripción completa . . . . .	52
Atributos importantes . . . . .	52
Trayectorias del Caso de Uso . . . . .	53
5.2.4. CUN4 Descargar archivo cifrado . . . . .	54
Descripción completa . . . . .	54
Atributos importantes . . . . .	54
Trayectorias del Caso de Uso . . . . .	54
5.2.5. CUN5 Eliminar archivo cifrado . . . . .	56
Descripción completa . . . . .	56
Atributos importantes . . . . .	56
Trayectorias del Caso de Uso . . . . .	56
5.2.6. CUCL1 Subir archivo . . . . .	58
Descripción completa . . . . .	58
Atributos importantes . . . . .	58
Trayectorias del Caso de Uso . . . . .	59
5.2.7. CUCL3 Descargar archivos descifrados. . . . .	61
Descripción completa . . . . .	61
Atributos importantes . . . . .	61
Trayectorias del Caso de Uso . . . . .	61
5.2.8. CUCL4 Eliminar archivos cifrado. . . . .	63
Descripción completa . . . . .	63
Atributos importantes . . . . .	63
Trayectorias del Caso de Uso . . . . .	63

5.2.9.	CUCL6 Iniciar Sesión. . . . .	65
	Descripción completa . . . . .	65
	Atributos importantes . . . . .	65
	Trayectorias del Caso de Uso . . . . .	65
5.2.10.	CUCL7 Registrar usuario. . . . .	68
	Descripción completa . . . . .	68
	Atributos importantes . . . . .	68
	Trayectorias del Caso de Uso . . . . .	68
5.3.	Diagramas de secuencia . . . . .	70
5.3.1.	Registrar Usuario . . . . .	70
5.3.2.	Iniciar Sesión . . . . .	74
5.3.3.	Subir Archivo . . . . .	78
5.3.4.	Firma a ciegas . . . . .	81
5.3.5.	Descargar Archivo . . . . .	82
5.3.6.	Eliminar Archivo . . . . .	84
5.4.	Mensajes del sistema . . . . .	85
5.4.1.	Mensajes . . . . .	86
<b>A.</b>	<b>Lista de acrónimos</b>	<b>90</b>
A.1.	Definiciones, acrónimos y abreviaturas . . . . .	90
<b>B.</b>	<b>Glosario de términos</b>	<b>92</b>
B.1.	Glosario de Términos . . . . .	92
	<b>Bibliografía</b>	<b>94</b>

# Índice de Figuras

1.1. Crecimiento global del tráfico en la nube . . . . .	3
1.2. Eliminación de Duplicados . . . . .	6
2.1. Diagrama de Criptografía Simétrica. . . . .	12
2.2. Diagrama de Criptografía Asimétrica. . . . .	13
2.3. Diagrama de Cifradores por Bloques . . . . .	14
4.1. Arquitectura general del sistema. . . . .	31
4.2. Arquitectura del sistema paso 1. . . . .	32
4.3. Arquitectura del sistema paso 2. . . . .	33
4.4. Arquitectura del sistema paso 3. . . . .	34
4.5. Arquitectura del sistema paso 4. . . . .	35
4.6. BPMN Subir archivo. . . . .	36
4.7. BPMN Descargar archivo. . . . .	38
4.8. BPMN Eliminar archivo. . . . .	39
4.9. Diagrama Entidad relación del sistema. . . . .	40
4.10. Diagrama de clases del sistema. . . . .	41
5.1. Diagrama de Casos de Uso del sistema. . . . .	48
5.2. Diagrama de secuencias de Registrar un usuario nuevo. . . . .	70
5.3. Diagrama de secuencias de Registrar un usuario nuevo con datos incorrectos. . . . .	71
5.4. Diagrama de secuencias de Registrar un usuario nuevo con datos incompletos. . . . .	72
5.5. Diagrama de secuencias de Registrar un usuario nuevo con usuario repetido. . . . .	73
5.6. Diagrama de secuencias de Iniciar sesion un usuario. . . . .	74
5.7. Diagrama de secuencias de Inicar sesion un usuario con datos incorrectos. . . . .	75
5.8. Diagrama de secuencias de Registrar un usuario nuevo con datos incompletos. . . . .	76
5.9. Diagrama de secuencias de Inicar sesion un usuario con campos vacios. . . . .	77
5.10. Diagrama de secuencias de subir un archivo nuevo. . . . .	78
5.11. Diagrama de secuencias de subir un archivo con carpeta vacia. . . . .	79
5.12. Diagrama de secuencias de subir un archivo incompatible. . . . .	79
5.13. Diagrama de secuencias de subir un archivo existente. . . . .	80
5.14. Diagrama de secuencias de la firma a ciegas del servidor de llaves. . . . .	81
5.15. Diagrama de secuencias de descargar un archivo de la nube. . . . .	82
5.16. Diagrama de secuencias de descargar un archivo de la nube no encontrado. . . . .	83

5.17. Diagrama de secuencias de eliminar un archivo de la nube. . . . .	84
---	----

# Índice de Tablas

4.1.	Componentes físicos . . . . .	28
4.2.	Requerimientos funcionales del servidor de llaves . . . . .	28
4.3.	Escalas de impacto de un riesgo . . . . .	29
4.4.	Análisis de riesgos del sistema . . . . .	30
4.5.	Requerimientos funcionales del servidor de llaves . . . . .	42
4.6.	Requerimientos funcionales del servidor de llaves . . . . .	42
4.7.	Requerimientos funcionales del cliente . . . . .	43
4.8.	Requerimientos funcionales del Servicio de almacenamiento (Nube) . . . . .	43
4.9.	Requerimientos no funcionales del sistema . . . . .	45

# Capítulo 1

## Introducción

### 1.1. Contexto

La criptografía es una ciencia que estudia técnicas matemáticas relacionadas con aspectos de seguridad de la información como confidencialidad, integridad de la información, entidades de autenticación y la autenticación de origen de datos entre otras. Esta ciencia tiene como principal objetivo el establecer la comunicación ya sea entre dos personas o dos entidades que requieren compartir información por un canal inseguro el cuál está propenso a un ataque para la manipulación o robo de esta información que viaja en el canal. Es por ello que la criptografía provee de protocolos, algoritmos y demás herramientas que ofrecen una solución para disminuir los ataques no deseados a la información que se requiera compartir de forma segura [7].

Estos protocolos, algoritmos y técnicas de la criptografía pueden ser utilizadas en distintas áreas de investigación o desarrollo, entre ellas esta el **Cómputo Nube**. La aplicación de la criptografía en los modelos computacionales basados en el cómputo nube tiene como objetivo proporcionar los algoritmos y protocolos criptográficos que den solución a los problemas en el crecimiento de los grandes volúmenes de información que hoy en día se tienen registro en el servicio de almacenamiento en línea (**Nube**), basándose en la eliminación de la información que se encuentre con una copia exacta en la nube. Por ejemplo:

El usuario registrado e identificado dentro de la nube como *Usuario A*, el día de hoy desea almacenar un archivo **Archivo F** que corresponde a la especificación de los requisitos para la obtención de una beca escolar. Este usuario para no extraviar o modificar el archivo lo almacena en la nube y ahí queda disponible para cuando el lo solicite. Ahora este usuario comparte este archivo mediante un dispositivo *USB* a su amigo ya que este también desea conocer los requisitos para solicitar una beca escolar, este amigo el cuál es otro usuario registrado e identificado dentro de la nube como *Usuario B* también quiere almacenar este archivo **Archivo F** en la nube, ya que requiere utilizar el espacio de memoria en su dispositivo *USB* para otras actividades y no desea perder los requisitos para la solicitud de su beca escolar. Ahora en la nube se encuentran almacenados 2 copias del **Archivo F** por dos diferentes usua-



rios identificados como *Usuario A* y *Usuario B*, ambos almacenaron el mismo archivo en el mismo lugar, sin darse cuenta que ahora este archivo se encuentra duplicado en la nube.

## 1.2. Problemática

Hoy en día el manejo de información en la sociedad juega un papel importante en el desarrollo de las actividades que la conforman. Millones de personas en el mundo tienen la facilidad de acceder a un dispositivo electrónico que les permite manipular esta información o almacenarla ya sea en un dispositivo físico o en algo más nuevo y eficiente como la nube, para posteriormente darle un uso específico. El cambio en las estrategias de negocio y la explosión de datos digitales se ha lanzado enormes demandas de alto volumen y almacenamiento de datos eficiente. Debido a los limitados recursos financieros y altos gastos de almacenamiento de datos electrónicos, los usuarios prefieren almacenar sus datos en los entornos de nube, el almacenamiento en la nube permite a sus usuarios transferir sus datos y aplicaciones en la web para que puedan operar esos programas sin ninguna infraestructura física necesaria [1].

Ahora bien, la información que circula en dispositivos electrónicos es mayor a la memoria disponible que ofrecen estos, a medida que el volumen de información aumenta, también lo hace la demanda para los servicios de almacenamiento en línea [2]. Un gran incremento en el uso de estos servicios implica tener más infraestructura y personal para que los sistemas de almacenamiento tengan más capacidad y puedan cubrir la demanda que se presenta en el mercado. Si bien el almacenamiento logró dar buenos resultados al cliente en sus primeras etapas, ahora la preocupación por el incremento de infraestructura para seguir dando esos resultados se ha incrementado considerablemente [1].

Para entender un poco más acerca de la problemática que se enfrenta el almacenamiento en la nube, mostramos el siguiente estudio realizado por EFE/Cisco:

EFE/Cisco realizó una estimación en el año 2014 en su cuarto informe anual Índice Global sobre la nube (2013-2018), donde prevé que en 2018 la mitad de la población mundial tendrá internet en sus hogares y más de la mitad almacenará contenidos en servicios personales de almacenamiento en la nube. El estudio adelanta que se triplicará el crecimiento de tráfico en los centros de datos en los próximos cinco años y la nube representará el 76 % de ese total, mientras que en 2013 éste solo representaba el 54 %, lo que supondría un aumento anual de 32 %. El tráfico en centros de datos, contando el saliente a usuarios finales, entre centros y dentro del propio sistema superará los 3.1 zettabytes de 2013 y llegará a los 8.6 que se esperan registrar en 2018, suponiendo una tasa anual del 23 % [5]. En 2018, el 53 % de los usuarios de internet con red doméstica usarán servicios de almacenamiento en la nube, aportando un tráfico medio por usuario de 811 mb mensuales, lejos de los 186 mb de 2013. Además se espera, según la nota que publicó cisco, que en 2018 el 69 % de la carga de trabajo en la nube se realizará en centros de datos con nubes privadas, un dato que fue del 78 % en 2013. El resto de cargas de trabajo, el 31 %, se realizarán en la nube pública, subiendo desde el 22 % del 2012 [5]. Crecimiento en el tráfico de información en la nube 1.1



Figura 1.1: Crecimiento global del tráfico en la nube

Una de las principales razones del incremento en el tamaño en la estructura de almacenamiento de servicios en línea es la duplicación de archivos por varios y diferentes usuarios, existen muchas copias en la nube de un mismo archivo que se encuentra presente en diferentes cuentas de usuarios. Un ejemplo sencillo y corto:  $n$  cantidad de usuarios pueden subir la misma canción a la nube, por lo tanto esta se encuentra almacenada en las  $n$  cantidad de cuentas que tiene registro la nube, esta misma canción que se encuentra almacenada está cubriendo un espacio en la memoria del servicio, si se tuviera una sola copia almacenada de esta canción se ahorraría mucho espacio en la nube que podría utilizarse para el almacenamiento de un archivo diferente.

Otro punto importante es la seguridad e integridad de la gran cantidad de información que se almacena en la nube. Este servicio de almacenamiento está sujeto a los ataques de adversarios que están interesados en el robo, manipulación o alteración de la información importante para los usuarios, el ataque que hasta ahora a sido intentado por estos adversarios es el **ataque por fuerza bruta**, dicho ataque funciona de la siguiente manera: En otros sistemas se propone una solución para evitar las duplicaciones en la nube, la solución fue el realizar una función al archivo que se quería almacenar y el resultado de esa función se convertía en una llave para poder cifrar el archivo y de esta manera quedaba seguro, sin embargo los adversarios siempre están en busca de corromper sistema de seguridad para intervenir el canal donde se comparte la información, si dicho adversario tenía la más mínima noción de cuál era el contenido del archivo que se intentó almacenar, este adversario podía estar intentando descifrar con lo poco que sabe de ese archivo almacenado ya que el mismo archivo se convertía en la llave que lo protegía, entonces el adversario realizaba ese proceso con todos los archivos registrados hasta que dicha llave coincidía con algún archivo almacenado y de esa manera logra corromper el esquema de seguridad que se había implementado para la protección de archivos sin duplicaciones.

### 1.3. Estado del Arte

APLICACIONES				
	DupLESS	TahoeFS	Flud Backup	ABS: The Apportioned Backup System
Evitar duplicación de archivos	Si	No	No	Si
Seguridad al cliente	Alta	Media	Media	Alta
Resistencia a ataques por fuerza bruta	Si	Media	No	No
Compromiso de resistencia ante fallos	Alto	Alto	Alto	Alto
Privacidad	Si	Si	No	Si
Servidor Seguro	Si	Si	Si	Si
Implementación	Pruebas	Actualmente Operacional	Actualmente Inactivo	Pruebas
Código abierto	Si	Si	Si	No
Gratuita o de paga	Sin información	Ambos	Gratuito	Sin información

#### ■ DupLESS

Este protocolo usa un servicio de almacenamiento en la nube, además implementa una interfaz sencilla con operaciones como guardar, recuperar o borrar un archivo. Es más adecuado para aplicaciones backup y busca proteger la confidencialidad de datos de los clientes, para ello usa seguridad semántica. Además promete capacidad de resistencia ante fallos, protección contra un servidor malintencionado, evitar duplicación de archivos y compatibilidad con diferentes sistemas operativos [2].

#### ■ TahoeFS

Este sistema utiliza diez diferentes servidores que se interconectan entre sí y consta de archivos mutables e inmutables. Se basa en la restricción a los usuarios de cierto comportamiento. A los archivos mutables les permite operaciones como leer y verificar y a los inmutables les permite leer, escribir y crear copia de solo lectura. Hace uso de cifrado convergente, el código Reed-Solomon para la tolerancia a fallos, el servicio AllMyData y control de acceso descentralizado. Es un servicio que promete almacenamiento seguro, integridad y confidencialidad a largo plazo y va enfocado a aplicaciones backup [18].

#### ■ Flud Backup

El proyecto Flud Backup está actualmente inactivo, sin embargo se crearon diversas

versiones para Ubuntu y Fedora donde usan paquetes distribuidos y un sistema de confianza. Prometen que los datos que se copian deben ser indestructibles y copias de seguridad descentralizadas [11].

- **ABS**

Este sistema se centra en el caso de uso de diez PC's conectadas a través de una LAN o a través de conexiones de Internet de Banda Ancha. Se basa en el almacenamiento de fragmentos y algo que denominan 'almacén de instancia única' donde si dos usuarios almacenan el mismo contenido del archivo, el sistema generará los fragmentos independientes de cada archivo y solamente almacenará una copia de cada fragmento en la red. También tiene un esquema de asignación de versiones basada en rsync (para generar una firma de diferencia sobre el archivo, la cual es una representación compacta, basada en el hash de un archivo que permita comparar entre dos versiones de archivos y verificar si están duplicadas. Promete el almacenamiento de datos seguro y eficiente, privacidad y seguridad, esto a través de tablas hash distribuidas, firmas de clave privadas, control de versiones y cifrado convergente. Además es tolerante a fallas catastróficas a nodos y permite unir nodos y restaurar operaciones sin pérdida de datos [3].

## 1.4. Justificación

En la actualidad millones de personas usan los servicios de almacenamiento que ofrece la nube, ya sean gratuitos o privados, este número de personas ha ido en un incremento exponencial lo cual hace que el espacio de almacenamiento disminuya, entonces ¿Cómo podría mitigar el problema de almacenamiento y tener privacidad de los datos al mismo tiempo.

Usando la criptografía clásica para poder cifrar un archivo se utiliza una clave privada la cuál es distinta para cada usuario, cada vez que se cifra un archivo el resultado de este es diferente para cada intento. Por tanto no se puede evitar la duplicación de archivos utilizando este mecanismo de la criptografía y se deben implementar soluciones más robustas.

Una solución para tener privacidad y evitar duplicación la proporcionó John R. Douceur, la cual dice que teniendo a  $M$  que será el contenido de un archivo de aquí en adelante denominado el mensaje, el cliente primero calcula una clave  $K \leftarrow H(M)$  mediante la aplicación de una función de hash criptográfica  $H$  al mensaje y luego calcula el texto cifrado  $C \leftarrow E(K, M)$  a través de un esquema de cifrado simétrico determinista. El derivado del mensaje  $K$  se almacena por separado cifrándolo con una llave por cliente. Un segundo cliente  $B$  cifra el mismo archivo  $M$  que producirá el mismo  $C$ , evitando la duplicación [10].

En el artículo publicado por Mihir Bellare, Sriram Keelveedhi, Thomas Ristenpart, nombrado "DupLESS: Server-Aided Encryption for Deduplicated Storage" [2], se observó que uno de los principales problemas al que nos enfrentamos es que el esquema de cifrado solo es seguro cuando el espacio de mensajes es demasiado grande, por lo tanto agentes externos pueden provocar agravios a la integridad de la información de los usuarios.

Si bien esta solución se ocupa de la duplicación de archivos deja muy vulnerable el aspecto

de la privacidad, ya que ante un espacio de mensajes pequeño las amenazas del adversario son demasiadas. Si se tuvieran como ejemplo 1000 mensajes, para el adversario sería muy fácil intentar encontrar la clave, probando las 1000 claves posibles generadas con la función hash, hasta descifrar el archivo, por lo tanto se comprueba que un espacio de 1000 mensajes sigue siendo pequeño.

Es por ello que este trabajo terminal tiene como principal meta atacar esta problemática de privacidad, proponiendo una arquitectura del sistema que a través de un servidor de claves se generaran claves de acuerdo al contenido del archivo, para con esta se pueda cifrar y luego almacenar en la nube donde se eludirá la duplicación de archivos.

## 1.5. Solución propuesta

La eliminación de duplicación de datos es una técnica que avanza favorablemente y da como resultado la disminución drástica de la cantidad de información duplicada en la nube, cuando esta se elimina del almacenamiento. En general, la eliminación de duplicados compara la información nueva que se requiere almacenar con la información que ya se tiene archivada y elimina las duplicaciones en la nube reduciendo la asignación de almacenamiento dentro de esta, esta disminución en la nube puede reducir las necesidades de almacenamiento en hasta un  $80\%$  para archivos y copias de seguridad que los usuarios resguardan en la nube como se ilustra en la figura 1.2. Las ventajas de no tener duplicados en la nube incluyen una mayor capacidad de almacenamiento y ahorro presupuestario, al igual que la minimización del ancho de banda para menos costosa y más rápida la repetición de la información fuera de la reserva simplificando y mejorando la gestión del almacenamiento de datos [14].

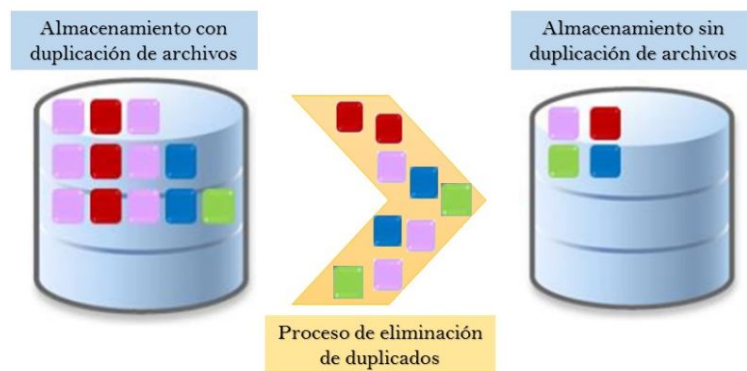


Figura 1.2: Eliminación de Duplicados

Una posible solución para la protección a los datos y eliminación de duplicaciones, es echar mano de la criptografía. Ciencia que se encarga del estudio de técnicas para transformar la información a una forma que no pueda entenderse a simple vista; sin embargo, el objetivo de la Criptografía no es sólo mantener los datos secretos, sino también protegerlos contra la modificación o manipulación y comprobar la fuente de los mismos [6].

Esta ciencia que mantiene la información segura se encuentra dividida en dos grandes tipos: **Criptografía Simétrica** y **Criptografía Asimétrica**.

*La criptografía simétrica* o también llamada criptografía de llave secreta, basa su seguridad en una sola llave que se comparte entre dos entidades que quieren compartir información, dicha llave es utilizada para cifrar un archivo al ser enviado a la otra entidad y este utilizará la misma llave para descifrarlo cuando lo reciba.

*La criptografía asimétrica* o criptografía de llave pública involucra el uso de un par de llaves para cada entidad que desea comunicarse, estas llaves llamadas pública y privada. Para que una entidad envíe un archivo a otra, necesita cifrar el archivo con la llave pública de esa entidad a la que se desea enviar, y cuando lo reciba esa entidad lo deberá descifrar con su llave privada o secreta. De esta manera se evita el compartir llaves para cifrar y descifrar como sucede en la criptografía simétrica y reduce los riesgos de un ataque de adversarios.

El objetivo de esta propuesta de solución es almacenar más datos en menos espacio mediante el uso de la criptografía. Esta ciencia nos proveerá con sus herramientas para la creación de un llavero criptográfico, dicho llavero realizará una firma la cuál dará paso a la creación de una llave correspondiente a un archivo  $F$  que se desee almacenar un usuario, si se llega a solicitar al llavero por un usuario diferente una nueva firma para la creación de una llave para el mismo archivo  $F$ , esta llave será la misma, ya que el mecanismo de funcionamiento entre un usuario y este servidor está diseñado para que sea capaz de identificar el mismo archivo sin comprometer el contenido e integridad de este. Esta llave va a lograr que cuando se cifre este archivo por  $n$  cantidad de usuarios diferentes que lo poseen, dicho cifrado será igual para la  $n$  cantidad de usuarios, permitiendo así que en la nube al subir estos cifrados se realice una comparación para que reconozca a quien pertenecen esos cifrados y sólo tenga almacenada una sola copia de este, ahorrando espacio de memoria y costos de infraestructura.

Puesto que ambas cuestiones, la eliminación de duplicados y la privacidad de la información, son importantes, se ha comenzado a proponer mecanismos que solucionen ambos problemas de manera conjunta, que son: Dupless [2], ABS: the apportioned backup system. [3], Flud Backup [11], SIGOPS Oper. Syst. [4], TahoeFS [18].

## 1.6. Objetivos

### 1.6.1. Objetivo General

Desarrollar un protocolo criptográfico para evitar la duplicación de archivos almacenados en la nube, garantizando la privacidad de los usuarios contra adversarios cuando el espacio de mensajes es pequeño, utilizando algoritmos criptográficos para su implementación.

### **1.6.2. Objetivos Específicos**

- Evitar la duplicación de archivos que sean almacenados por los usuarios de la nube
- Proteger ante los adversarios la información de los usuarios de la nube
- Establecer un esquema de autenticación de usuarios
- Reducir la pérdida y filtración de información de los usuarios de la nube
- Evitar los ataques por fuerza bruta al contenido de los archivos de usuarios en la nube.

# Capítulo 2

## Marco Teórico

El contenido de este capítulo abordará temas estrechamente relacionados con la criptografía, la seguridad de la información y las implicaciones que ésta podría traer si se encuentra corrompida por algun adversario. También, este capítulo contiene información acerca de los 2 tipos de criptografía que existen mencionando los diferentes esquemas de cifrado y los modos de operación que son utilizados por algunos de estos. De igual forma se describe con detalle los servicios que ofrece el cómputo nube, haciendo énfasis en un servicio en particular que es el de almacenamiento que se utilizará para la implementación de este protocolo criptográfico.

### 2.1. Definiciones.

- **Criptografía.** La Criptografía es la ciencia que se encarga del estudio de técnicas matemáticas relacionadas con aspectos de seguridad para transformar la información a una forma que no pueda entenderse a simple vista; sin embargo, el objetivo de la Criptografía no es sólo mantener los datos secretos, sino también protegerlos contra modificación y comprobar la fuente de los mismos [7].
- **Criptoanálisis.** Es la ciencia que se ocupa del análisis de un texto cifrado para obtener la información original sin conocimiento de la clave secreta, esto es, de forma ilícita rompiendo así los procedimientos de cifrado establecidos por la Criptografía, por lo que se dice que Criptoanálisis y Criptografía son ciencias complementarias pero contrarias. El criptoanálisis es el arte de descifrar comunicaciones cifradas sin conocer las llaves [19].

#### 2.1.1. Servicios criptográficos.

Los servicios criptográficos son aquellos que garantizan en un sistema de información la adquisición, almacenamiento, procesamiento y transmisión de la información y para lograrlo se valen de uno o más objetivos fundamentales.

- **Confidencialidad.** Es un servicio utilizado para mantener el contenido de la información de todos, excepto los autorizados a tenerla. El secreto es un término sinónimo



de confidencialidad y privacidad. Hay numerosos enfoques para proporcionar confidencialidad, que van desde la protección física a los algoritmos matemáticos que hacen que los datos sean ininteligibles.

- **Autenticación.** Es un servicio relacionado con la identificación. Esta función se aplica tanto a las entidades como a la propia información. Dos partes que participan en una comunicación deben identificarse entre sí. La información entregada a través de un canal debe ser autenticada en cuanto al origen, fecha de origen, contenido de los datos, tiempo enviado, etc. Por estas razones este aspecto de la criptografía suele subdividirse en dos clases principales: autenticación de entidad y autenticación de origen de datos. La autenticación de origen de datos proporciona implícitamente la integridad de los datos (si se modifica un mensaje, la fuente ha cambiado).
- **Integridad.** Es un servicio que se ocupa de la alteración no autorizada de los datos. Para asegurar la integridad de los datos, se debe tener la capacidad de detectar la manipulación de datos por parte de algún adversario. La manipulación de datos incluye cosas tales como inserción, supresión y sustitución.
- **No repudio.** Es un servicio que impide a una entidad negar compromisos o acciones anteriores. Cuando surgen disputas debido a que una entidad niega que se tomaron ciertas acciones, es necesario un medio para resolver la situación. Por ejemplo, una entidad puede autorizar la compra de una propiedad por otra entidad y posteriormente denegar que se concedió dicha autorización. Se necesita un procedimiento que involucre a un tercero de confianza para resolver la disputa [7].

## 2.2. Ataques a servicios criptográficos.

Un ataque es una violación a la seguridad de la información realizada por intrusos que tienen acceso físico al sistema sin ningún tipo de restricción, su objetivo es robar la información o hacer que ésta pierda valor relativo, o que disminuyan las posibilidades de su supervivencia a largo plazo.

- **Ataque sólo con texto cifrado.** Este caso es cuando el criptoanalista sólo conoce el criptograma y el algoritmo con que fue generado; con esta información pretende obtener el texto en claro.
- **Ataque con texto original conocido.** En esta situación el criptoanalista conoce mensajes en claro seleccionados por él mismo y sus correspondientes criptogramas, así como el algoritmo con que éstos fueron generados; aquí el objetivo es conocer la clave secreta y poder describir libremente cualquier texto.
- **Ataque con texto cifrado escogido.** El criptoanalista conoce el algoritmo de cifrado, así como un criptograma seleccionado por él mismo y su correspondiente texto en claro, su objetivo es obtener el mensaje en claro de todo criptograma que intercepte.

- **Ataque con texto escogido.** En este caso el criptoanalista además de conocer el algoritmo de cifrado y el criptograma que quiere descriptar, también conoce el criptograma de un texto en claro que él elija y el mensaje en claro de un criptograma también elegido por él [20].

### 2.2.1. Ataques por fuerza bruta

Los ataques de fuerza bruta se basan en un concepto simple: Oscar, el atacante, obtiene el texto cifrado escuchando en el canal y pasa a tener un pedazo corto del texto claro, por ejemplo, el encabezado de un archivo que fue cifrado. Oscar ahora simplemente descifra el primer pedazo del texto cifrado con todas las claves posibles. Si el texto claro resultante coincide con el pedazo corto del texto claro, sabe que ha encontrado la clave correcta.

#### **Definición 2.1** *Ataque de fuerza bruta*

Sea  $(x, y)$  el texto claro y el texto cifrado, y sea  $K = [K1, \dots, Kk]$  el espacio clave de todas las claves posibles  $k_i$ . Un ataque de fuerza bruta comprueba que para cada  $k_i \in K$  si

$$d_{k_i}(y) = x$$

*Si la igualdad se mantiene, se encuentra una posible clave correcta; Si no, se procede con la siguiente clave.*

En la práctica, un ataque de fuerza bruta puede ser más complicado porque las claves incorrectas pueden dar resultados positivos falsos.

Es importante señalar que un ataque de fuerza bruta contra cifrados simétricos es siempre posible en un principio. Si es factible en la práctica depende del espacio clave, es decir, en el número de posibles claves que existen para un cifrado dado. Si se están probando todas las claves en muchas computadoras modernas toma demasiado tiempo, es decir, varias décadas, el cifrado es computacionalmente seguro contra un ataque de fuerza bruta [8].

## 2.3. Criptografía Simétrica.

Los esquemas criptográficos simétricos también se conocen como esquemas o algoritmos de clave simétrica o clave secreta. Consideremos un esquema de cifrado que consiste en los conjuntos de transformaciones de cifrado y descifrado  $Ee: e \in \mathcal{K}$  y  $Dd: d \in \mathcal{K}$ , respectivamente, donde  $\mathcal{K}$  es el espacio de clave. El esquema de cifrado se dice que es de clave simétrica si para cada par asociado de cifrado/descifrado de claves  $(e, d)$ , es computacionalmente “fácil” para determinar  $d$  conociendo sólo  $e$ , y determinar  $e$  a partir de  $d$ . Puesto que  $e = d$  en los esquemas de cifrado de clave simétrica más prácticos, la clave simétrica término se convierte apropiado [7].

Cuando existen dos entidades que quieren comunicarse para compartir información a través de un canal inseguro que puede ser internet, teléfonos móviles o comunicación LAN inalámbrica, etc, se presenta un problema, ya que puede existir algún adversario que tiene acceso a ese canal de comunicación, a esto se le llama espionaje. En esta situación, la criptografía simétrica ofrece una solución: la entidad cifra su mensaje  $x$  usando un algoritmo simétrico, dando el texto cifrado  $y$ . La entidad destinatario recibe el texto cifrado y descifra el mensaje, si se cuenta con un algoritmo de cifrado fuerte, el texto cifrado se verá como bits aleatorios al adversario y no contendrá ninguna información que le resulte útil [8].

Los algoritmos criptográficos simétricos tienen dos versiones: cifrador en bloque y cifrador de flujo. El beneficio en cuanto al uso de un algoritmo simétrico se encuentra en el procesamiento rápido para cifrar y descifrar un alto volumen de datos. El cifrado simétrico es una práctica eficaz de almacenamiento de información sensible en una base de datos, un registro o archivo [12].

Así como la criptografía tiene grandes ventajas para el procesamiento rápido para la solución entre la comunicación de dos entidades a través de un canal inseguro, también cuenta con ciertas desventajas que son:

- La seguridad depende de un secreto compartido entre el emisor y el receptor.
- La administración de las claves no es escalable.
- La distribución manual de llaves es costosa, ocupa mucho tiempo y es propensa a errores.
- La distribución de claves debe hacerse a través de algún medio seguro como centros de distribución de llaves, implementación de algoritmos, etc [15].

El esquema de cifrado simétrico se puede representar a través de la siguiente figura 2.1.

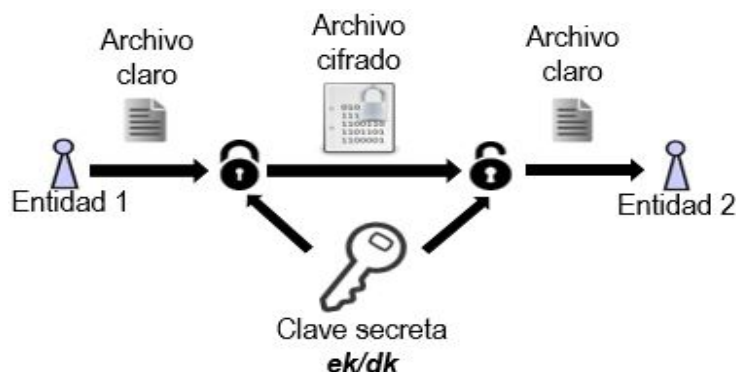


Figura 2.1: Diagrama de Criptografía Simétrica.

## 2.4. Criptografía Asimétrica.

EL objetivo detrás de un criptosistema de clave pública es que podría ser posible encontrar un criptosistema donde es computacionalmente imposible determinar  $dk$  dado  $ek$ . Si es así, entonces la regla de cifrado  $ek$  es una clave pública que podría ser publicada en un directorio, por ejemplo (de ahí el término sistema de clave pública). La ventaja de un sistema de clave pública es que una entidad puede enviar un mensaje cifrado a otra entidad (sin la comunicación previa de una clave secreta compartida) utilizando la regla de cifrado pública  $ek$ . La entidad que recibe la comunicación será la única que puede descifrar el texto cifrado, utilizando la regla de descifrado  $dk$ , que se llama la clave privada [17].

Sea  $Ee$ :  $e \in \mathcal{K}$  un conjunto de transformaciones de cifrado, y sea  $Dd$ :  $d \in \mathcal{K}$  el conjunto de transformaciones de descifrado correspondientes, donde  $\mathcal{K}$  es el espacio de clave. Consideremos cualquier par de transformaciones asociadas de cifrado/descifrado ( $Ee$ ,  $Dd$ ) y suponemos que cada par tiene la propiedad de saber que  $Ee$  es computacionalmente inviable, dado un texto cifrado  $c \in \mathcal{C}$ , para encontrar el mensaje  $m \in \mathcal{M}$  tal que  $Ee(m) = C$ . Esta propiedad implica que dada  $e$  es imposible determinar la clave de descifrado correspondiente  $d$ . ( $e$  y  $d$  son simplemente medios para describir las funciones de cifrado y descifrado, respectivamente) [7].

El cifrado asimétrico puede ser representado como aparece en la figura 2.2.

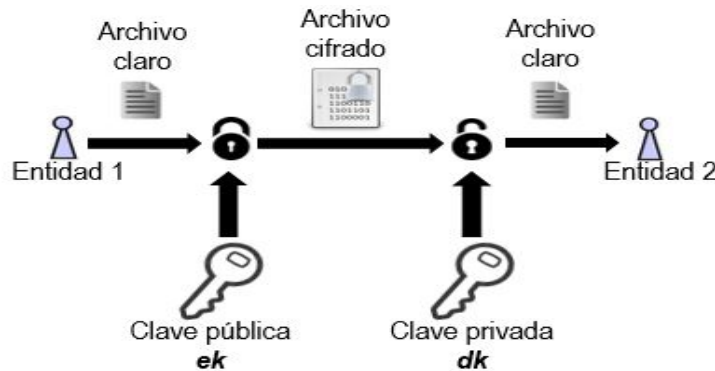


Figura 2.2: Diagrama de Criptografía Asimétrica.

Los beneficios de la criptografía asimétrica son la solución a los problemas de la criptografía simétrica, pues las claves públicas pueden ser distribuidas con toda tranquilidad, no valen de nada sin las claves privadas. El cifrado asimétrico se emplea frecuentemente para elaborar firmas digitales, un mecanismo que permite al receptor de un mensaje firmado digitalmente poder identificar a la entidad que origino ese mensaje y de esa manera confirmar que el mensaje no ha sido alterado. También el cifrado de clave pública es utilizado para pasar con seguridad una clave privada, que posteriormente, será la que se utilice para cifrar y/o descifrar otra información.

## 2.5. Cifrador por bloques.

Un cifrado de bloques es una función que convierte bloques de texto sin formato de  $n$  bits a bloques de texto cifrado de  $n$  bits; a  $n$  se denomina longitud de bloque. Dicho de otra manera, el cifrador por bloques actúa como una simple cifra de sustitución con un gran tamaño de caracteres. La función que convierte los bloques de texto simple está parametrizada por una clave  $K$  de  $k$ -bits, tomando valores de un subconjunto  $\mathcal{K}$  (el espacio de la llave) del conjunto de todos los vectores de  $k$  bits  $V_k$ . Generalmente la clave  $K$  se elige al azar. El uso de bloques de texto claro y texto cifrado de igual tamaño evita la expansión de datos [7]. Para permitir el descifrado único, la función de cifrado debe ser de uno por uno (es decir, invertible).

Para los bloques de texto de  $n$ -bit, texto cifrado de  $n$ -bit y una clave fija de  $n$ -bit, la función de cifrado es Una biyección, es decir que el texto cifrado debe ser siempre diferente pero cuando se descifre este debe corresponder al texto en claro, definiendo una permutación en vectores de  $n$ -bits. Cada clave potencial define una biyección diferente. El número de llaves es de longitud  $|\mathcal{K}|$ , y el tamaño efectivo de la clave es de longitud  $|\mathcal{K}|$ ; Esto es igual a la longitud de la clave si todos los vectores de  $k$ -bits son claves válidas ( $\mathcal{K} = V_k$ ). Si las llaves son equiprobables (Misma probabilidad) y cada una define una biyección diferente, la entropía (Medida de incertidumbre) del espacio clave es también de longitud  $|\mathcal{K}|$  [7].

Los cifradores por bloques mas usados son AES (Advanced Encryption Standard, por sus siglas en inglés) y DES (Data Encryption Standard, por sus siglas en inglés). [13]

Los cifradores por bloques pueden ser representados como se ve en la figura 2.3.

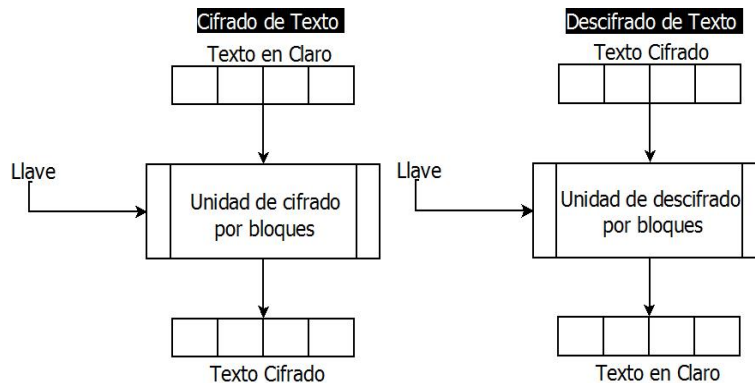


Figura 2.3: Diagrama de Cifradores por Bloques

## 2.6. Cómputo Nube.

El cómputo nube definido así por el NIST, es un modelo para permitir un acceso a la red ubicuo, es decir, que se encuentra presente en todas partes al mismo tiempo y conveniente a un conjunto de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se puede aprovisionar y liberar rápidamente con un esfuerzo mínimo de gestión o una interacción entre el proveedor de servicios. Este modelo de cómputo nube se compone de 5 características esenciales, 3 modelos de servicio y 4 modelos de despliegue.

### Características:

- **Auto-servicio bajo demanda.**

Un consumidor puede proporcionar unilateralmente capacidades del tiempo del servidor y el almacenamiento en red, según se necesite automáticamente sin interacción con cada proveedor de servicios.

- **Amplio acceso a la red.**

Las capacidades están disponibles a través de la red y se accede a través de mecanismos que promueven el uso por plataformas de cliente heterogéneas finas o gruesas (por ejemplo, teléfonos móviles, tablets, computadoras portátiles y estaciones de trabajo)

- **Agrupación de recursos.**

Los recursos informáticos del proveedor se agrupan para servir a múltiples consumidores utilizando un modelo de multi-usuario, con diferentes recursos físicos y virtuales asignados dinámicamente y reasignados de acuerdo con la demanda del consumidor. Hay una sensación de independencia de ubicación en que el cliente generalmente no tiene control o conocimiento sobre la ubicación exacta de los recursos proporcionados, pero puede especificar la ubicación en un nivel superior de abstracción (por ejemplo, país, estado o centro de datos). Ejemplos de recursos incluyen almacenamiento, procesamiento, memoria y ancho de banda de la red.

- **Elasticidad rápida.**

Las capacidades pueden ser suministradas elásticamente y liberadas, en algunos casos de forma automática, para escalar rápidamente hacia fuera y hacia adentro proporcional a la demanda. Para el consumidor, las capacidades disponibles para la provisión a menudo parecen ser ilimitadas y pueden ser apropiadas en cualquier cantidad en cualquier momento.

- **Servicio medido.**

Los sistemas de cómputo nube controlan y optimizan automáticamente el uso de recursos aprovechando una capacidad de medición en algún nivel de abstracción apropiado al tipo de servicio (por ejemplo, almacenamiento, procesamiento, ancho de banda y cuentas de usuario activas). El uso de recursos puede ser monitoreado, controlado y

reportado, proporcionando transparencia tanto para el proveedor como para el consumidor del servicio utilizado.

**Modelos de servicio.** Entre los diversos tipos de servicios de cómputo en la nube proporcionados internamente o por proveedores de servicios de terceros, los más habituales son:

- **Software como Servicio (SaaS).**

La capacidad proporcionada al consumidor es utilizar las aplicaciones del proveedor que se ejecutan en una infraestructura en la nube. Las aplicaciones son accesibles desde varios dispositivos cliente a través de una interfaz de cliente ligero, como un navegador web (por ejemplo, correo electrónico basado en web) o una interfaz de programa. El consumidor no gestiona ni controla la infraestructura oculta de la nube, incluyendo la red, los servidores, los sistemas operativos, el almacenamiento o incluso las capacidades de las aplicaciones individuales, con la posible excepción de las limitadas configuraciones específicas de la configuración de la aplicación.

- **Plataforma como Servicio (PaaS).**

- **Infraestructura como Servicio (IaaS).**

Explicado a grandes rasgos el software como servicio, ya que es donde se centrará todo el desarrollo de este protocolo.

**Modelos de despliegue.**

- **Nube privada.**

La infraestructura de la nube está preparada para el uso exclusivo de una sola organización que comprende varios consumidores (por ejemplo, unidades de negocio). Puede ser propiedad, administrado y operado por el órgano.

- **Nube de la comunidad.**

La infraestructura de la nube está preparada para uso exclusivo por una comunidad específica de consumidores de organizaciones que tienen preocupaciones compartidas (por ejemplo, misión, requisitos de seguridad, política y consideraciones de cumplimiento). Puede ser propiedad, administrado y operado por una o más de las organizaciones de la comunidad, un tercero, o una combinación de ellos, y puede existir dentro o fuera de las instalaciones.

- **Nube pública.**

La infraestructura de la nube está preparada para el uso abierto por el público en general. Puede ser propiedad, administrado y operado por una organización comercial, académica u gubernamental, o alguna combinación de ellos. Existe en las instalaciones del proveedor de la nube.

- **Nube híbrida.**

La infraestructura de la nube es una composición de dos o más infraestructuras de nube

distintas (privadas, comunitarias o públicas) que siguen siendo entidades únicas, pero están unidas por una tecnología estandarizada o propietaria que permite la portabilidad de datos y aplicaciones (por ejemplo, burbujas de nube para equilibrar la carga entre Nubes).

## **Problemas en Cómputo Nube.**

### ■ **Interfaces y API poco seguros.**

Generalmente los proveedores de servicios en la nube ofrecen una serie de interfaces y API (del inglés, Application Programming Interface) para controlar e interactuar con los recursos. De este modo, toda la organización, el control, la provisión y la monitorización de los servicios cloud se realiza a través de estos API o interfaces. Dado que todo (autenticación, acceso, cifrado de datos, etc.) se realiza a través de estas herramientas, se hace necesario que los interfaces estén diseñados de forma segura, evitando así los problemas de seguridad, tanto los que son intencionados como los que se producen de forma accidental.

### ■ **Pérdida o fuga de información.**

Existen muchas formas en las que los datos se pueden ver comprometidos. Por ejemplo, el borrado o modificación de datos sin tener una copia de seguridad de los originales, supone una pérdida de datos. En la nube, aumenta el riesgo de que los datos se vean comprometidos ya que el número de interacciones entre ellos se multiplica debido a la propia arquitectura de la misma. Esto deriva en pérdida de imagen de la compañía, daños económicos y, si se trata de fugas, problemas legales, infracciones de normas, etc.

### ■ **Secuestro de sesión o servicio.**

En un entorno en la nube, si un atacante obtiene las credenciales de un usuario del entorno puede acceder a actividades y transacciones, manipular datos, devolver información falsificada o redirigir a los clientes a sitios maliciosos.



# Capítulo 3

## Protocolo Dupless

### 3.1. Firma digital

Una firma digital es un mecanismo de autenticación que permite al creador de un mensaje fijar un código que actúa como una firma. La firma es formada tomando el hash del mensaje y cifrar el mensaje con clave privada del creador. La firma garantiza el origen y la integridad del mensaje [16].

El estándar de la firma digital (DSS) es un estándar NIST que utiliza el algoritmo de hash seguro (SHA). El desarrollo más importante del trabajo sobre criptografía de clave pública es la firma digital. La firma digital proporciona un conjunto de capacidades de seguridad que sería difícil de aplicar en cualquier otra forma [16].

Autenticación de mensajes protege dos partes que intercambian mensajes de terceros. Sin embargo, no protege a los dos partidos uno contra el otro. Varias formas de disputa entre los dos son posibles [16].

En situaciones donde no existe una completa confianza entre el emisor y el receptor, se necesita algo más que la autenticación. La solución más atractiva para este problema es la firma digital. La firma digital es análoga a la firma manuscrita. Debe tener las siguientes propiedades:

- Debe verificar el autor, la fecha y hora de la firma.
- Debe autenticar el contenido en el momento de la firma.
- Debe ser verificable por terceras personas, para resolver los conflictos [16].

Así, la función de firma digital incluye la función de autenticación. Sobre la base de estas propiedades, podemos formular los siguientes requisitos para una firma digital:

- La firma debe ser un patrón de bits que depende del mensaje firmado.
- La firma debe usar cierta información única del remitente, para evitar la falsificación y la negación.

- Debe ser relativamente fácil de realizar la firma digital.
- Debe ser relativamente fácil de reconocer y verificar la firma digital.
- Debe ser sea computacionalmente imposible falsificar una firma digital, mediante la construcción de un nuevo mensaje para una firma digital existente o mediante la construcción de una fraudulenta firma digital para un mensaje dado.
- Debe ser práctico conservar una copia de la firma digital en almacenamiento de información [16].

## 3.2. RSA

El esquema de criptografía RSA, a veces denominado algoritmo Rivest-Shamir-Adleman, es actualmente el esquema criptográfico asimétrico más utilizado, aunque las curvas elípticas y los esquemas de logaritmos discretos están ganando terreno. RSA fue patentado en los Estados Unidos (pero no en el resto del mundo) hasta el 2000. La función unidireccional subyacente de RSA es el problema de factorización de enteros: Multiplicar dos grandes primos es computacionalmente fácil (de hecho, se puede hacer con Papel y lápiz), pero factorizar el producto resultante es muy difícil, el teorema de Euler y la función  $\varphi$  de Euler desempeñan papeles importantes en RSA [8].

Hay muchas aplicaciones para RSA, pero en la práctica se usa con más frecuencia para:

- Cifrado de pequeñas piezas de datos, especialmente para el transporte de claves.
- Las firmas digitales, por ejemplo, para certificados digitales en Internet [8].

### Cifrado y descifrado

El cifrado y descifrado RSA se realiza en el campo de los números enteros  $Z_n$  y los cálculos modulares desempeñan un papel central. RSA cifra el texto en claro  $x$ , donde consideramos que la cadena de bits que representa  $x$  es un elemento en  $Z_n = 0, 1, \dots, n-1$ . Como consecuencia, el valor binario del texto en claro  $x$  debe ser menor que  $n$ . Lo mismo ocurre con el texto cifrado. El cifrado con la clave pública y el descifrado con la clave privada son los siguientes:

#### **Cifrado RSA**

Dada la clave pública  $(n, e) = k_{pub}$  y el texto en claro  $x$ , la función de cifrado es:

$$y = ek_{pub}(x) \equiv x^e \mod n$$

Donde:  $x, y \in Z_n$

#### **Descifrado RSA**

Dada la clave privada  $d = K_{pr}$  y el texto cifrado  $y$ , la función de descifrado es:

$$x = dk_{pr}(y) \equiv y^d \mod n$$

Donde:  $x, y \in Z_n$  [8].

### Generación de llaves

Estos son los pasos involucrados en el cálculo de la clave pública y privada para un criptosistema RSA.

- Elegir 2 números primos grandes  $p$  y  $q$ .
- Calcular  $n = p \cdot q$ .
- Calcular  $\varphi(n) = (p - 1)(q - 1)$ .
- Seleccionar la clave pública  $e \in 1, 2, \dots, \varphi(n) - 1$  tal que  $\gcd(e, \varphi(N)) = 1$ .
- Calcular la clave privada  $d$  tal que,  $d \cdot e \equiv \text{mod } \varphi(n)$  [8].

### Requisitos para el criptosistema RSA:

- Dado que un atacante tiene acceso a la clave pública, debe ser computacionalmente imposible determinar la clave privada  $d$  dados los valores de clave pública  $e$  y  $n$ .
- Como  $x$  es único hasta el tamaño del módulo  $n$ , no podemos cifrar más de  $l$  bits con un cifrado RSA, donde  $l$  es la longitud de bits de  $n$ .
- Debe ser relativamente fácil calcular  $x \cdot e \text{ mod } n$ , es decir, cifrar y  $y \cdot d \text{ mod } n$ , es decir, descifrar. Esto significa que necesitamos un método para una rápida exponenciación con números grandes.
- Para un  $n$  dado, debe haber muchos pares de clave privada / clave pública, de lo contrario un atacante podría ser capaz de realizar un ataque de fuerza bruta. (Resulta que esta exigencia es fácil de satisfacer.) [8].

## 3.3. Firmas a ciegas.

Las firmas a ciegas son un tipo especial de firmas digitales en las que se firma algo que no se conoce. Para hacer firmas a ciegas se utilizan factores de opacidad, para ocultar el mensaje original que se requiere que esté firmado, y así la autoridad no pueda conocer lo que está firmando. Por lo tanto, el propósito de una firma a ciegas es evitar que el firmante B conozca el mensaje que firma; y así posteriormente, sea incapaz de asociar el mensaje que firmó con el remitente A. Entonces, las firmas a ciegas tienen aplicación en varias situaciones. A continuación se mencionan dos de ellas:

- Cuando se utiliza dinero electrónico. En este caso,  $m$  representa un valor monetario que A (el cliente) tiene derecho a gastar. Y así, cuando  $m$  y  $s(m)$  se presentan a B (el banco) para efectuar el pago, B es incapaz de identificar al cliente que originalmente le dio ese dinero electrónico a firmar, pues le fue enviado de manera oculta. Lo anterior permite que la identidad de A permanezca anónima, y sus movimientos financieros no puedan ser monitoreados.

- En las elecciones electrónicas también pueden utilizarse las firmas a ciegas, ya que se requiere que  $B$  (una autoridad electoral) no conozca la identidad de  $A$  (el votante) debido a que el voto debe efectuarse de manera anónima. Sin embargo, es necesario que  $A$  demuestre que su voto  $m$  es válido. Lo cual se logra cuando  $A$  presenta ante  $B$  la firma  $s(m)$ . Y se sabe de antemano que  $B$  no puede asociar  $s(m)$  a  $A$ , debido a que el votante previamente le envió a  $B$  su voto  $m$  pero de forma oculta para que se lo firmara. [9]

### 3.4. Firmas a ciegas con RSA.

Un esquema de firmas a ciegas es un protocolo que involucra un remitente  $A$  y un firmante  $B$ . La idea básica en un esquema basado en RSA es la siguiente:  $A$  le envía cierta información  $z$  a  $B$ , donde  $z$  está compuesto por el mensaje que se desea que firme  $B$  y por un factor de ocultamiento cifrado con la llave pública de  $B$ , es decir,  $z = (m * b^e) \bmod n$ .  $B$  firma dicha información  $s(z)$  y se la regresa a  $A$ . De la firma  $s(z)$ ,  $A$  puede obtener la firma de  $B$  para el mensaje  $m$ , quitando el factor de ocultamiento  $b$  a  $s(z)$ . Pues:

$$s(z) = (m * b^e) \bmod n = (m^d * b^{ed}) \bmod n = (m^d \bmod n) * b$$

Ahora bien, al dividir  $s(z)$  entre  $b$ , obtendremos  $s(m)$ :

$$(m) = s(z)/b = ((m^d \bmod n) * b)/b = m^d \bmod n$$

Al finalizar el protocolo,  $B$  no conoce el mensaje  $m$  ni la firma asociada a él  $s(m)$  que ahora posee  $A$  [9]

### 3.5. Funciones Hash.

A continuación se describirán las características de las *funciones hash*, también conocidas como *funciones de resumen*. Las funciones hash basan su definición en funciones de un solo sentido (*one-way functions*, en inglés). Una función de un solo sentido es aquella que para un valor  $x$ , es muy fácil calcular  $f(x)$ , pero es muy difícil hallar  $f^{-1}(x)$ . Es complicado en general, hallar funciones de este tipo y probar que lo son.

**Definición 3.1** *Una función hash, es una función de un solo sentido cuya entrada  $m$  es un mensaje de longitud arbitraria y la salida es una cadena binaria de longitud fija. Al resumen o hash de un mensaje  $m$ , se le denotará como  $h(m)$ . Una función hash debe tener las siguientes propiedades:*

- Para cualquier mensaje  $m$ , debe ser posible calcular  $h(m)$  eficientemente.
- Dado  $h(m)$ , debe ser computacionalmente difícil, hallar un mensaje  $m'$ , tal que  $h(m) = h(m')$ .

- *Debe ser computacionalmente difícil, hallar dos mensajes  $m$  y  $m'$  tales que  $h(m) = h(m')$ .*

Entre las funciones hash que se usan para criptografía están: MD2, MD4, MD5, donde MD significa *Message Digest*, y el algoritmo estándar al momento de escribir éstas notas es el *Secure Hash Algorithm* por sus siglas en inglés SHA. La MD5 fue diseñada por Ron Rivest, toma como entrada un mensaje de longitud arbitraria y proporciona como salida una cadena binaria de 128 bits. El mensaje de entrada se procesa por bloques de 512 bits. La SHA 256 fue diseñada por en NIST (*National Institute of Standards and Technology*) y se estableció como estándar en 1993. Recibe como entrada un mensaje con longitud menor a  $2^{64}$  bits y como salida se obtiene una cadena binaria de 160 bits. Al igual que el MD5, se procesa en bloques de 512 bits [16].

# Capítulo 4

## Análisis

### 4.1. Estudio de Factibilidad

Como se expuso anteriormente, el protocolo criptográfico nos arroja excelentes beneficios en cuanto a la optimización de espacio de memoria en la nube y garantiza la seguridad e integridad de los datos de usuarios registrados en la nube. Hasta ahora se tiene muy claro cuál será el impacto de la implementación de este protocolo, pero aún queda por establecer cuán viable es la realización de este protocolo en cuanto a facilidades técnicas y operativas.

#### 4.1.1. Factibilidad técnica

Para la implementación del protocolo criptográfico, se cuentan con las herramientas de software necesarias ya que estas existen y están disponibles para su uso y disponibilidad inmediata al desarrollo del protocolo. Cabe mencionar que las herramientas que involucran todo el desarrollo del protocolo, se encuentran completas en tanto a funcionalidades, soportes, documentación, seguridad, etc. En cuanto a los conocimientos necesarios al protocolo, se cuentan con los suficientes que este protocolo necesita para su aplicación. Dichos conocimientos se han adquirido a lo largo de la estadía en la ingeniería mediante las clases recibidas en las instalaciones de la escuela o por la adquisición por medios externos. Ahora bien, un pequeño inconveniente para la aplicación del protocolo, es el desarrollo del software, ya que existen partes fundamentales del protocolo que requieren de nuevo conocimiento que aun no se adquiere y es necesario aprender, por tanto existe una pequeña posibilidad de que esto signifique un retraso en la producción del protocolo.

**Lenguaje de programación:** para el desarrollo del sistema se estará trabajando con varios lenguajes de programación, la base de este desarrollo será Python, ya que es un lenguaje de fácil comprensión y con una lógica bastante familiar a lenguajes que anteriormente se habían trabajado, además de que Python nos permite trabajar primitivas criptográficas con una mayor facilidad.

**Sistema gestor de base de datos:** Se implementará MySQL como un manejador de base de datos, ya que es software libre y además se cuenta con una mayor experiencia de manejo y entendimiento.

De acuerdo con los requerimientos del sistema, los componentes necesarios para la implementación de este protocolo son:

**Hardware** Se requiere de equipo de cómputo para poder llevar a cabo la codificación del protocolo y la configuración de los servidores que le darán soporte a las actividades realizadas en la interacción del usuario con el sistema. El equipo de trabajo cuenta con 3 computadoras personales (LAPTOP) que son las siguientes:

Componentes Físicos (Hardware)	
Componente	Características
Laptop HP Pavilion g4	<ul style="list-style-type: none"> <li>■ Procesador: AMD A6-4400M APU 2.70Hz</li> <li>■ Memoria RAM: 8.00GB</li> <li>■ Disco Duro: 750GB</li> <li>■ Tipo Sistema: 64bits x64</li> </ul>
Sigue en la página siguiente.	

Componente	Características
Acer Aspire V5	<ul style="list-style-type: none"> <li>■ Procesador: Intel(R) Celeron(R) 1.50GHz</li> <li>■ Memoria RAM: 2.00GB</li> <li>■ Disco Duro: 250GB</li> <li>■ Tipo Sistema: 64bits x64</li> </ul>
Sigue en la página siguiente.	



Componente	Características
HP probook	<ul style="list-style-type: none"> <li>■ Procesador: AMD Phenom(tm) II X2 545 3.00GHz</li> <li>■ Memoria RAM: 4.00GB</li> <li>■ Disco Duro: 350GB</li> <li>■ Tipo Sistema: 64bits x64</li> </ul>
Sigue en la página siguiente.	

Componente	Características
ownCloud	<ul style="list-style-type: none"> <li>■ Servicio de alojamiento de archivos con almacenamiento en la nube</li> <li>■ Servidores de instalación: PHP, SQLite, MySQL, PostgreSQL</li> <li>■ Servidor de archivos: WebDAV</li> <li>■ Calendario de sincronización: CardDAV</li> <li>■ Sistema operativo: MMultipataforma</li> </ul>
Sigue en la página siguiente.	

Componente	Características
------------	-----------------

Tabla 4.1: Componentes físicos

**Software** Con respecto al software, no se pretende usar un sistema operativo en específico ya que el servicio de la nube es compatible con cualquier sistema operativo en que se esté desarrollando. Sin embargo se cuenta con los siguientes sistemas operativos:

Sistemas operativos	
Tipo	Cantidad
Windows 8.1 Pro	2
Windows 10 Pro	1

Tabla 4.2: Requerimientos funcionales del servidor de llaves

#### 4.1.2. Factibilidad operativa

El uso e implantación de este protocolo criptográfico, tiene como principal objetivo el eliminar las duplicaciones de archivos que se almacenan en la nube y de igual manera el evitar los ataques de adversarios por fuerza bruta a la información de los usuarios en la nube, por lo cual resulta factible la operación del proyecto, ya que se está proponiendo una solución de bajo costo con grandes beneficios. Hoy en día va en aumento el número de usuarios de la nube, ya sean empresas, organizaciones, personas, etc. todos estos usuarios buscan su información rápido y de fácil acceso en cualquier lugar con una conexión a internet, y esto hace que también aumente la cantidad de información almacenada que en ocasiones está replicada en uno o más usuarios. Este protocolo pretende atender al problema del crecimiento en la infraestructura de la nube ofreciendo una solución para contrarrestarlo y de esa manera más usuarios podrán tener acceso a la nube sin necesidad de aumentar a la infraestructura de esta.

#### 4.1.3. Factibilidad económica

El estudio de factibilidad económica nos permite analizar los costos - beneficios monetarios que se obtendrán con el desarrollo del proyecto, ya que dicho protocolo en un largo plazo podría comenzar en producción industrial para su comercialización y distribución en el mercado de las tecnologías de la información, ya que proveerá de servicios en el cómputo en la nube que hoy en día las empresas están optando por invertir en él y optimizar sus recursos y procesos del negocio.

En cuanto al análisis de el gasto económico que se requiere para la implementación y la realización de pruebas del protocolo, existe un beneficio ya que todos los componentes para el desarrollo del sistema son libres, es decir que no se realizará la adquisición de licencias de

desarrollo ni programas para la implementación.

El servidor que nos proveerá del almacenamiento en la nube de nombre **ownCloud.org** es de libre acceso y no existe restricción alguna la utilización de este. Nos permite una libre manipulación y configuración de tal modo que podamos cumplir nuestros objetivos.

#### 4.1.4. Análisis de riesgos

EL análisis de riesgos consiste en identificar los riesgos que este sistema puede tener en su futura implementación y desarrollo. Este análisis presenta los riesgos identificados en el sistema y 3 indicadores (Prioridad, Probabilidad e Impacto) fundamentales para llevar a cabo la gestión de riesgos y así tener presente cuales son los riesgos más destacados en el sistema para dimensionar las consecuencias que estos podrían traer consigo si llegaran a ocurrir. Para elaborar este análisis de riesgos, se evaluaron los impactos que estos riesgos pueden traer, mediante 5 escalas de impacto de un riesgo, dichas escalas son:

##### Escalas de impacto de un riesgo

Escalas de impacto de un riesgo					
<i>Objetivo del proyecto</i>	Muy bajo/ 0.05	Bajo/ 0.10	Moderado/ 0.20	Alto/ 0.40	Muy alto/ 0.80
<b>Costo</b>	Aumento insignificante de costo	Aumento del costo <10 %	Aumento del costo 10 - 20 %	Aumento del costo 20 - 40 %	Aumento del costo >40 %
<b>Tiempo</b>	Aumento insignificante de tiempo	Aumento del tiempo <5 %	Aumento del tiempo 5 - 10 %	Aumento del tiempo 10 - 20 %	Aumento del tiempo >20 %
<b>Alcance</b>	Disminución del alcance apenas perceptible	Áreas de alcance secundarias afectadas	Áreas de alcance principales afectadas	Reducción del alcance inaceptable para los objetivos	El producto terminado es inservible
<b>Calidad</b>	Degradación de calidad apenas perceptible	Sólo las aplicaciones muy exigentes se ven afectadas	La reducción de la calidad requiere aprobación externa	Reducción de calidad inaceptable	El producto terminado es inservible

Tabla 4.3: Escalas de impacto de un riesgo

Análisis de riesgos					
ID	Riesgo	Prioridad	Probabilidad	Impacto	Causa
R1	Modificar requerimientos	ALTA	MEDIA	ALTO	Nuevas características del protocolo
R2	Cambios en tecnología	BAJA	MEDIA	BAJO	La tecnología usada es menos eficiente y causa conflictos
R3	Falta de usuarios y peticiones	ALTA	BAJA	BAJO	No se localizaron todos los involucrados en el protocolo
R4	Equipo de cómputo	ALTA	BAJA	MUY BAJO	El equipo de cómputo empleado falla o no se encuentra funcionando
R5	Incumplimiento de acuerdos	ALTA	ALTA	MODERADO	Retrazo en actividades programadas
R6	Problemas de planeación	MEDIA	ALTA	ALTO	Falta de comunicación por parte del equipo
R7	Falta de recursos	ALTA	MEDIA	MUY ALTO	No se cuenta con el poder adquisitivo de recursos e insumos
R8	Falta de conocimientos	MEDIA	MEDIA	MUY ALTO	El equipo no se encuentra capacitado para el desarrollo del sistema.
R9	Sobre estimación de insumos	MEDIA	MEDIA	MODERADO	Los insumos para la implementación del sistema no son los adecuados para el desarrollo.

Tabla 4.4: Análisis de riesgos del sistema

## 4.2. Arquitectura del sistema.

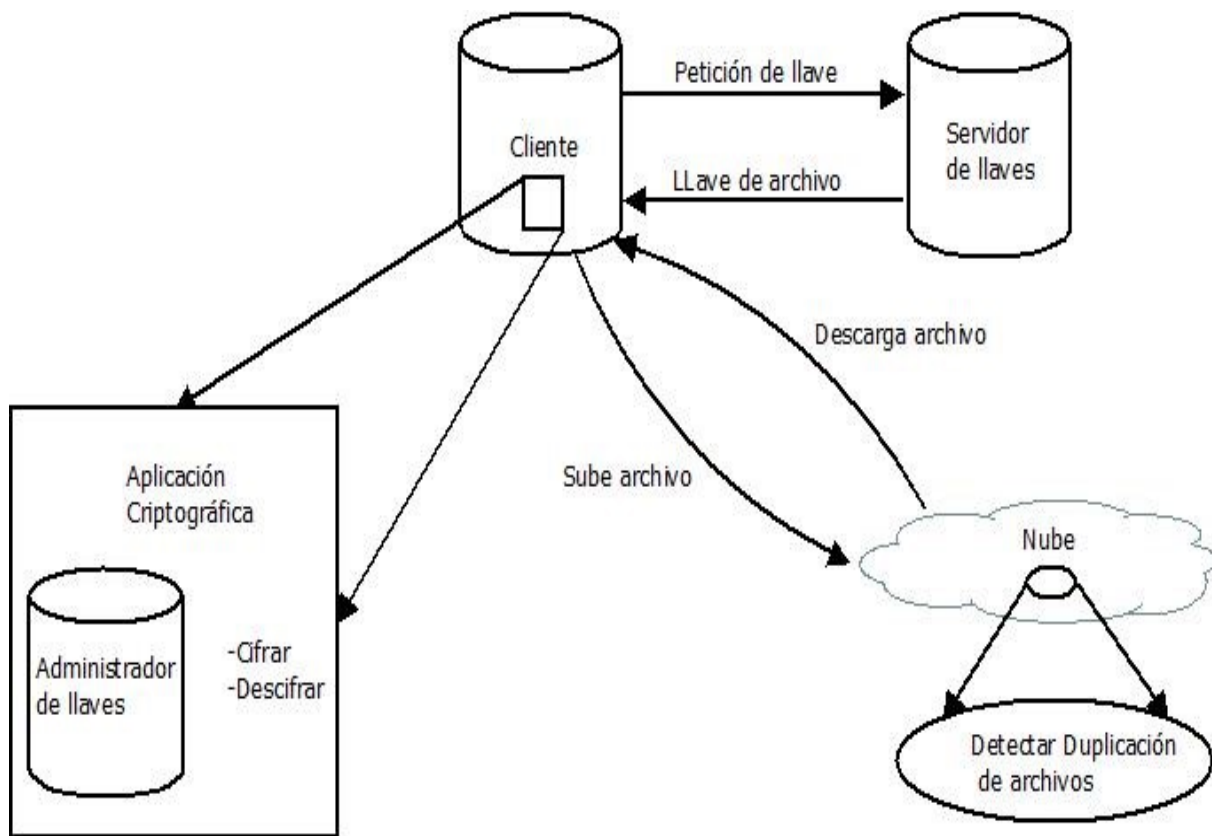


Figura 4.1: Arquitectura general del sistema.

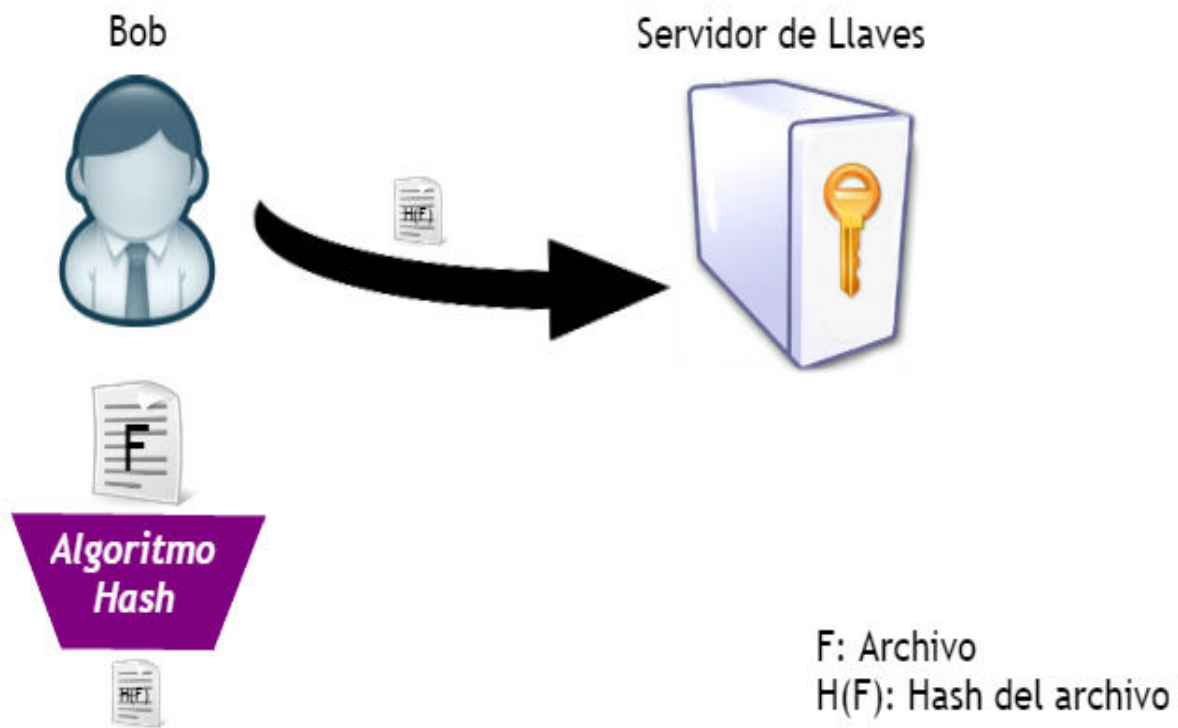
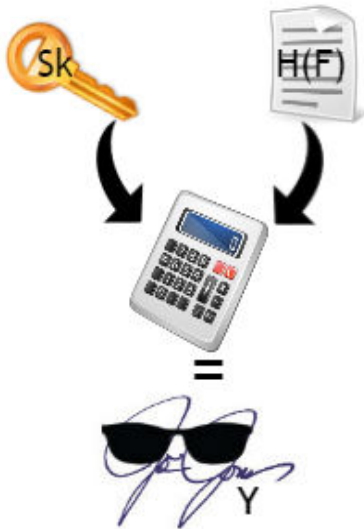


Figura 4.2: Arquitectura del sistema paso 1.

Servidor de Llaves



Bob



Sk: Llave privada del Servidor de Llaves  
H(F): Hash del archivo  
Y: Firma a ciegas

Figura 4.3: Arquitectura del sistema paso 2.



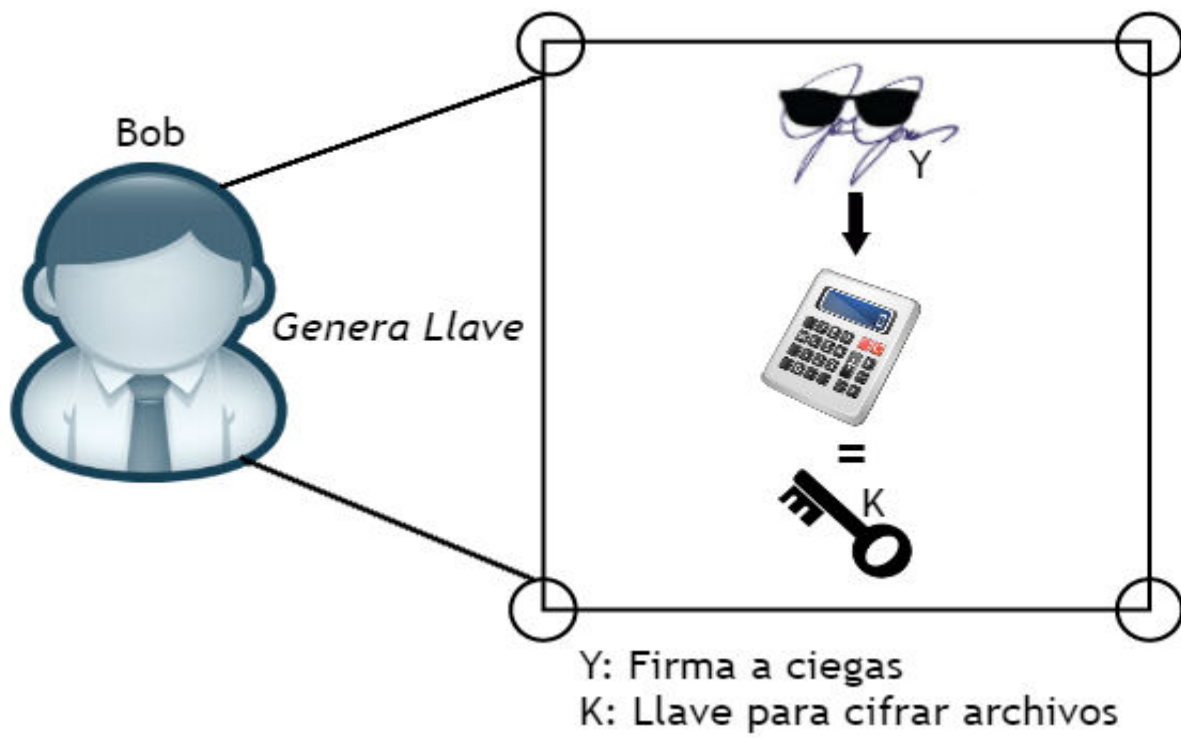


Figura 4.4: Arquitectura del sistema paso 3.

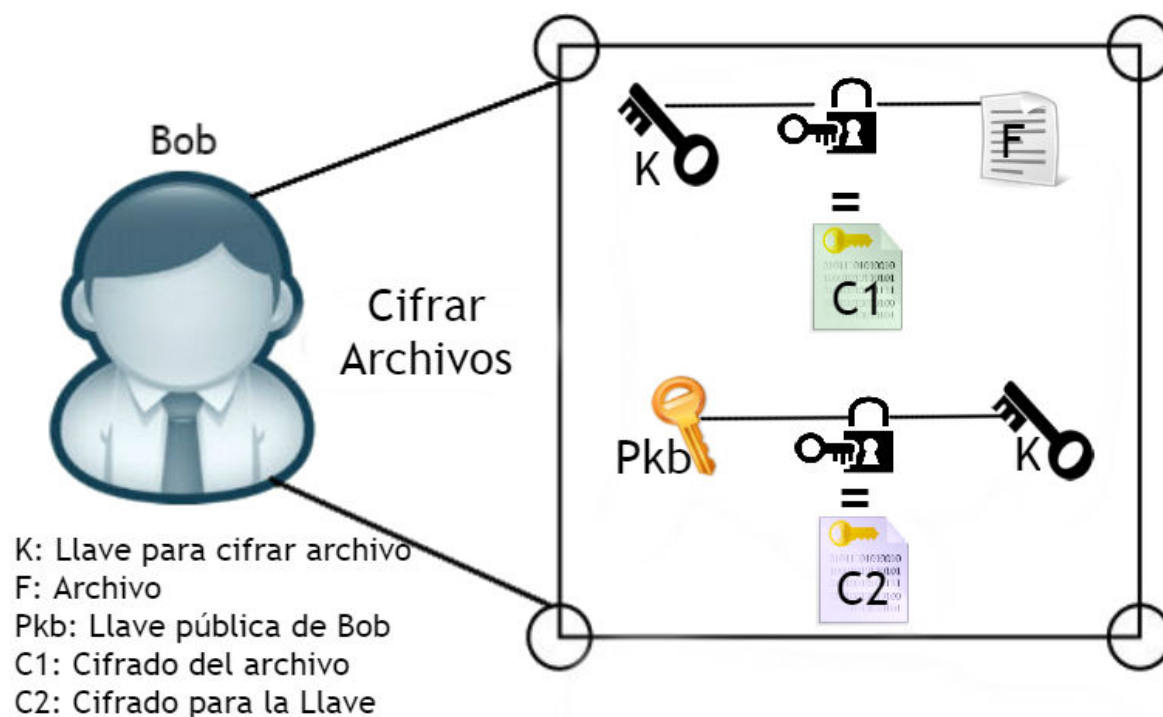


Figura 4.5: Arquitectura del sistema paso 4.

### 4.3. Descripción de procesos

#### 4.3.1. Descripción del proceso subir archivo.

El proceso inicia cuando el cliente desea subir un archivo nuevo, el cliente debe dar clic en la opción de subir archivo y seleccionar el archivo que desee subir, el sistema va a calcular el hash del archivo elegido, después hará unas operaciones aritméticas con el has para generar una x que se enviara al servidor para que realice una firma a ciegas, y con esta firma que se le regresara al cliente, se va a generar del lado del cliente su llave "k" que será la llave con la cual cifrara el archivo, y así si otro archivo que se quiera subir es igual a este tendrá la misma k y podrá el sistema detectar que son duplicados, también el sistema va a cifrar la llave k por si se le llega a perder al cliente, para poder almacenarlo en la nube el sistema mandara el hash del archivo cifrado para ver si ya está registrado en la base de datos, si es así solo guarda la llave y actualiza la lista de los usuario que comparten el archivo, de lo contrario solicita la llave cifrada y el archivo cifrado para almacenarlos y actualiza su lista de usuario añadiendo un archivo en ella, y por último se le notificará al cliente que su archivo ha sido almacenado.

## Participantes

Participantes		
Nombre	Descripción	Responsabilidades
Servidor	Actor que realiza la firma a ciegas del archivo.	<ul style="list-style-type: none"> <li>■ Firma a ciegas.</li> </ul>
Cliente	Actor que sube archivos a la Nube.	<ul style="list-style-type: none"> <li>■ Selecciona archivo a subir.</li> <li>■ Genera hash del archivo.</li> <li>■ Calcula la llave k.</li> <li>■ Cifra los archivos a subir.</li> </ul>
Nube	Actor que almacena los archivos.	<ul style="list-style-type: none"> <li>■ Almacena los archivos seleccionados.</li> <li>■ Genera lista de usuarios relacionados.</li> </ul>

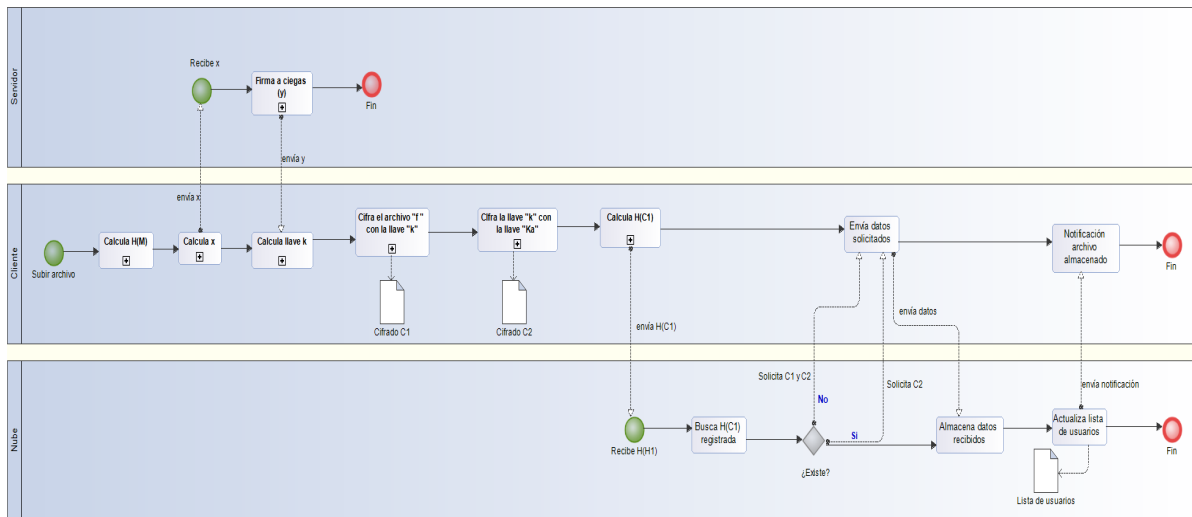


Figura 4.6: BPMN Subir archivo.

### 4.3.2. Descripción del proceso Descargar archivo.

El proceso inicia cuando el cliente desea descargar un archivo, el cliente debe dar clic en la opción de descargar archivo y seleccionar el archivo que desee descargar, el sistema va a mandar el nombre del archivo a la nube para que busque en su base de datos los archivos correspondientes al usuario y nombre del archivo, se le regresaran al cliente y el sistema en el lado del cliente deberá descifrar el archivo C2 que contiene la llave k para poder descifrar el otro archivo C1 donde se encuentra el archivo original, el sistema notificara al cliente que su archivo se ha descargado con éxito y este podrá abrirlo.

#### Participantes

Participantes		
Nombre	Descripción	Responsabilidades
Cliente	Actor que descarga archivos de la Nube.	<ul style="list-style-type: none"><li>■ Selecciona archivo a descargar.</li><li>■ Descifra los archivos descargados.</li></ul>
Nube	Actor que almacena los archivos.	<ul style="list-style-type: none"><li>■ Almacena los archivos seleccionados.</li><li>■ Genera lista de usuarios relacionados.</li><li>■ Enviar los archivos a descargar.</li></ul>

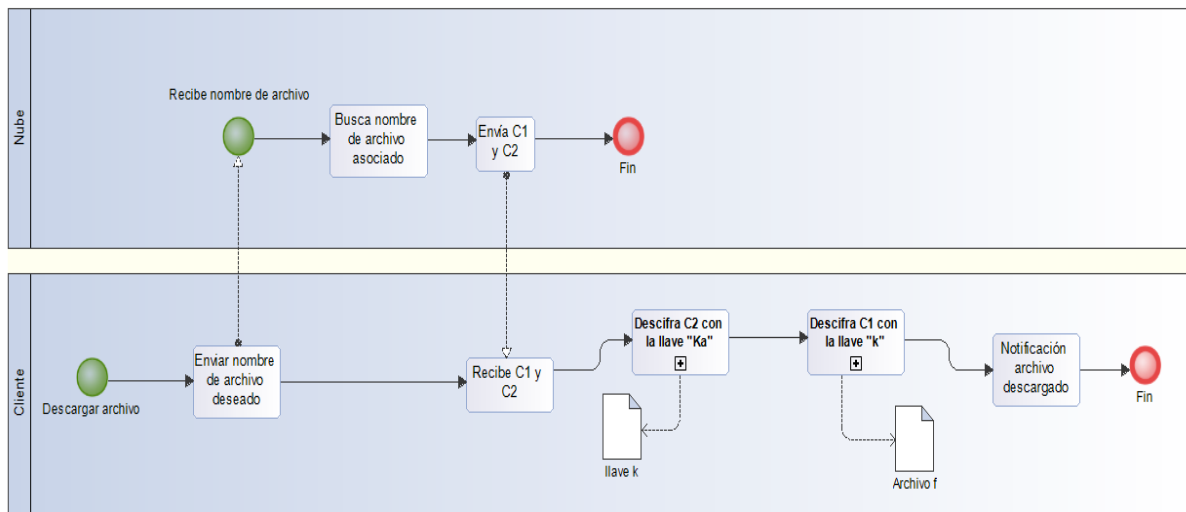


Figura 4.7: BPMN Descargar archivo.

### 4.3.3. Descripción del proceso eliminar archivo.

El proceso inicia cuando el cliente desea eliminar un archivo nuevo, el cliente debe dar clic en la opción de eliminar archivo y seleccionar el archivo que desee eliminar, el sistema va a enviar el nombre del archivo a la nube donde este buscara en su base de datos los archivos que corresponden al usuario y nombre del archivo, los va a eliminar de su base de datos y actualizara su lista de usuarios eliminado de ella los datos del archivo y usuario que coinciden con el archivo eliminado, se le enviara una notificación al cliente que su archivo ha sido eliminado con éxito de la nube.

## Participantes

Participantes		
Nombre	Descripción	Responsabilidades
Cliente	Actor que elimina archivos de la Nube.	<ul style="list-style-type: none"> <li>■ Selecciona archivo a eliminar.</li> </ul>
Nube	Actor que almacena los archivos.	<ul style="list-style-type: none"> <li>■ Elimina los archivos seleccionados.</li> <li>■ Genera lista de usuarios relacionados.</li> </ul>

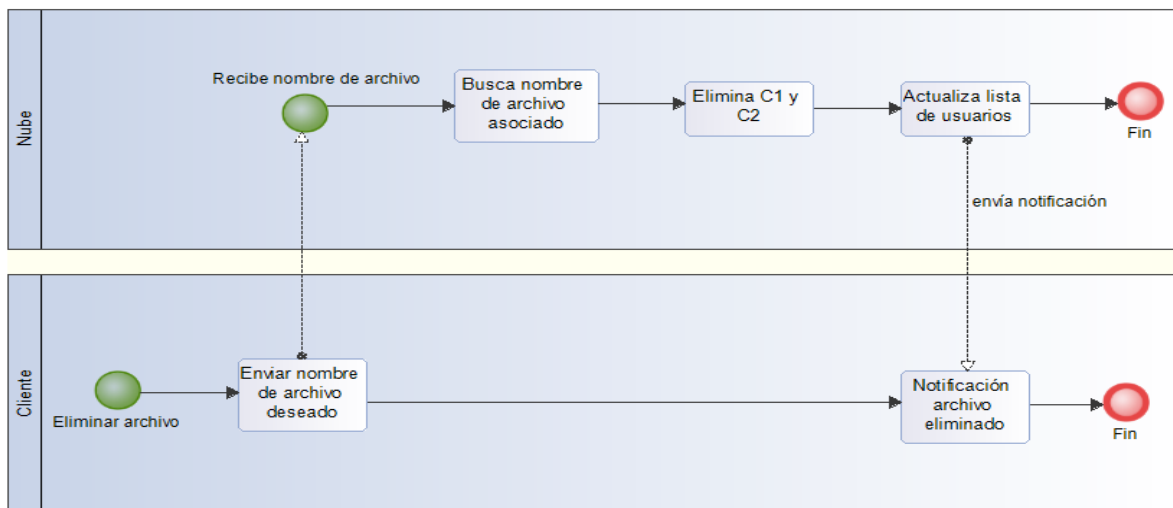


Figura 4.8: BPMN Eliminar archivo.

## 4.4. Modelo de entidades.

### 4.4.1. Diagrama de Entidad Relación.

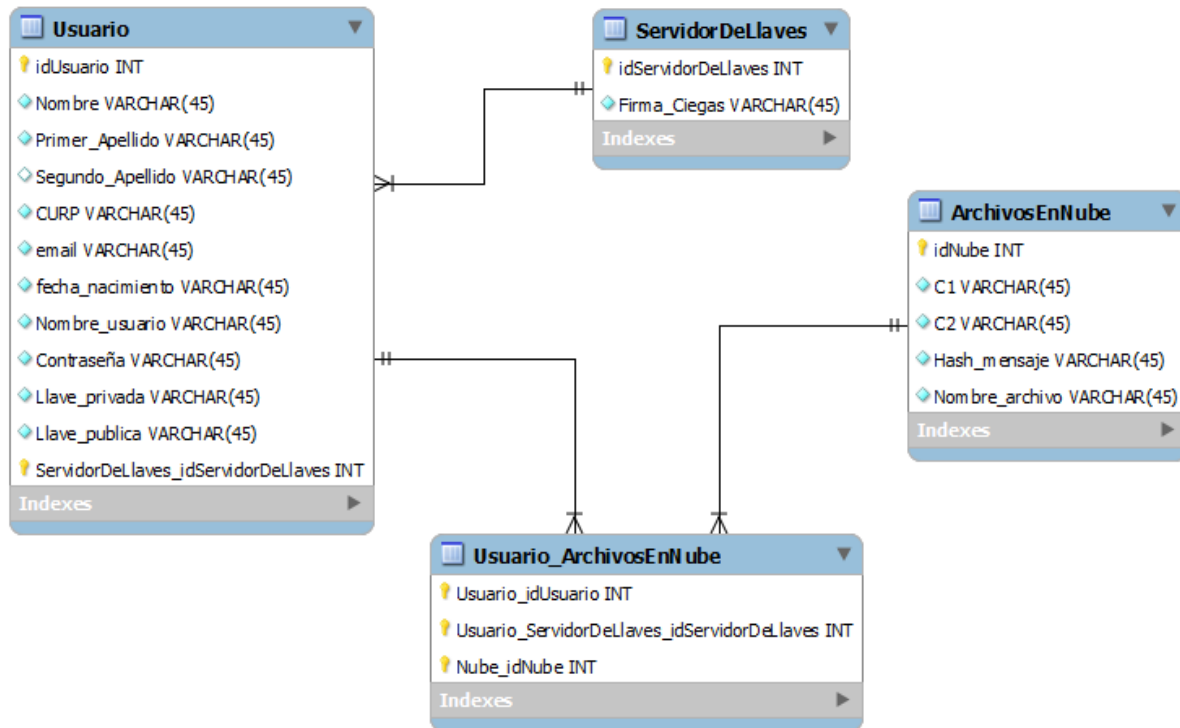


Figura 4.9: Diagrama Entidad relación del sistema.

#### 4.4.2. Diagrama de clases.

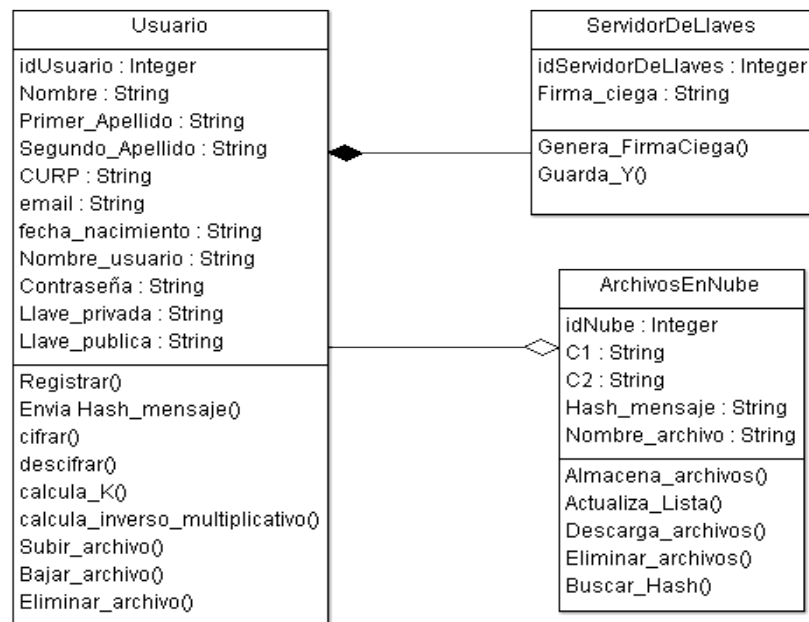


Figura 4.10: Diagrama de clases del sistema.



## 4.5. Requerimientos Funcionales.

Requerimientos funcionales del protocolo	
ID	Descripción
RF1	El sistema permitirá el registro de un nuevo usuario
RF2	El sistema permitirá al usuario iniciar sesión para comenzar a manipular su información
RF3	El sistema permitirá al usuario gestionar su perfil para la visualización, actualización y configuración de su información
RF4	El sistema permitirá al usuario gestionar los archivos que dicho usuario tiene registrado en su perfil

Tabla 4.5: Requerimientos funcionales del servidor de llaves

Servidor de Llaves	
ID	Descripción
RF – SLL1	El sistema permitirá la generación de llaves de usuario a través de las llaves pública y privada del servidor de llaves
RF – SLL2	El sistema permitirá la firma a ciegas (y) de cualquier archivo que se desee almacenar

Tabla 4.6: Requerimientos funcionales del servidor de llaves

Cliente	
ID	Descripción
RF – CL1	El sistema permitirá al usuario gestionar archivos: Subir, Descargar, Eliminar
RF – CL2	El sistema permitirá al usuario subir un archivo (F) cifrado al servicio de almacenamiento
RF – CL3	El sistema permitirá al usuario descargar un archivo (F) descifrado elegido de su lista de archivos en el servicio de almacenamiento.
RF – CL4	El sistema permitirá al usuario eliminar un archivo (F) cuando el usuario elige alguno de su lista de archivos cargados en el servicio de almacenamiento.

Tabla 4.7: Requerimientos funcionales del cliente

Servicio de almacenamiento (Nube)	
ID	Descripción
RF – N1	El sistema permitirá al servicio de almacenamiento llevar a cabo un proceso para la detección de archivos duplicados.

Tabla 4.8: Requerimientos funcionales del Servicio de almacenamiento (Nube)

## 4.6. Requerimientos No Funcionales

Requerimientos No Funcionales		
ID	Atributo	Descripción
RNF1	Eficiencia	<ul style="list-style-type: none"> <li>■ El servidor de llaves tendrá la capacidad de realizar 1000 peticiones de gestión de almacenamiento de archivos por segundo.</li> <li>■ El sistema podrá funcionar de forma correcta con usuarios conectados de manera concurrente.</li> <li>■ Los archivos que sean gestionados dentro del servidor de almacenamiento, deben ser actualizados en la base datos y la visualización de cada cliente de manera casi inmediata.</li> </ul>
Sigue en la página siguiente.		

ID	Atributo	Descripción
RNF2	Fiabilidad	<ul style="list-style-type: none"> <li>■ La pérdida de consultas en el servidor de llaves es menor a 3 veces el máximo de consultas realizadas.</li> <li>■ Los archivos almacenados en el servidor de almacenamiento deben ser recuperados por el usuario al instante en que este lo solicite.</li> <li>■ El tiempo de latencia que existe entre el servidor de llaves y el cliente será de máximo 118ms.</li> </ul>
RNF3	Seguridad	<ul style="list-style-type: none"> <li>■ El sistema almacenará los datos de los usuarios y sus contraseñas en una base de datos MySQL, dichos datos serán modificados mínimo 2 veces al año.</li> <li>■ Se autenticarán los clientes antes de comenzar el proceso de generación de llaves de archivo.</li> <li>■ El servidor de llaves firmará claves para un sólo mensaje a la vez sin saber el contenido de éste.</li> <li>■ El inicio de sesión de usuarios estará protegido en un canal seguro utilizando algoritmos criptográficos.</li> <li>■ Las funciones hash de archivos a almacenar utilizarán la función criptográfica SHA-(256)</li> <li>■ Los formularios para ingresar datos al sistema estarán validados por tipo de dato, longitud e internamente se utilizará un ORM (Object Relational Mapping) para evitar inyecciones SQL.</li> </ul>
Sigue en la página siguiente.		

ID	Atributo	Descripción
RNF4	Mantenibilidad	<ul style="list-style-type: none"> <li>■ Cuaquier nuevo requerimiento funcional o no funcional tendrá que ser analizado y diseñado para poder cuantificar las implicaciones que este tendrá sobre el funcionamiento del sistema.</li> <li>■ El sistema contará con un plan de pruebas que facilitará la identificación de posibles fallas existentes en el funcionamiento de este.</li> </ul>
RNF5	Usabilidad	<ul style="list-style-type: none"> <li>■ El tiempo de aprendizaje del sistema por un usuario deberá ser menor a 15 días.</li> <li>■ El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final.</li> <li>■ El sistema debe poseer interfaces gráficas bien formadas.</li> </ul>
RNF6	Extensibilidad	<ul style="list-style-type: none"> <li>■ El sistema podrá tener un crecimiento a futuro ya que este será programado por módulos lo cuál hará más fácil su crecimiento.</li> <li>■ El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final.</li> <li>■ El sistema debe poseer interfaces gráficas bien formadas.</li> </ul>

Tabla 4.9: Requerimientos no funcionales del sistema

## 4.7. Reglas de Negocio

### Regla de Negocio: RN1 Datos requeridos

**Descripción:** El usuario debe ingresar toda la información marcada como requerida en el modelo conceptual del negocio.

**Tipo:** Restricción de operación.

### Regla de Negocio: RN2 Datos correctos

**Descripción:** La información que el usuario proporcione, debe ser del tipo y longitud definida en el modelo conceptual del negocio.

**Tipo:** Restricción de operación.

**Regla de Negocio:** RN3 Unicidad de elementos

**Descripción:** Hay ciertos elementos que no pueden repetirse, ya sea por ser idénticos o por coincidir en uno o más datos. Esto se define como dato único en la tabla de atributos del modelo conceptual del negocio para cada entidad.

**Tipo:** Restricción de operación.

**Regla de Negocio:** RN4 Usuario registrado

**Descripción:** El usuarios debe tener una cuenta activa en el sistema.

**Tipo:** Hecho

# Capítulo 5

## Diseño

### 5.1. Especificación de Plataforma

#### Estación de trabajo y computadores personales

##### 1. Hardware

- Procesador: Intel Celeron N2840 o superior
- RAM: 2 GB o superior

##### 2. Software

- Navegador Web: Chrome 24, Firefox 24, Internet Explorer 10, Safari 7 o superior.
  - Soporte para cookies

##### 3. Red

- Conexión a internet de 2 Mb/s

## 5.2. Casos de Uso

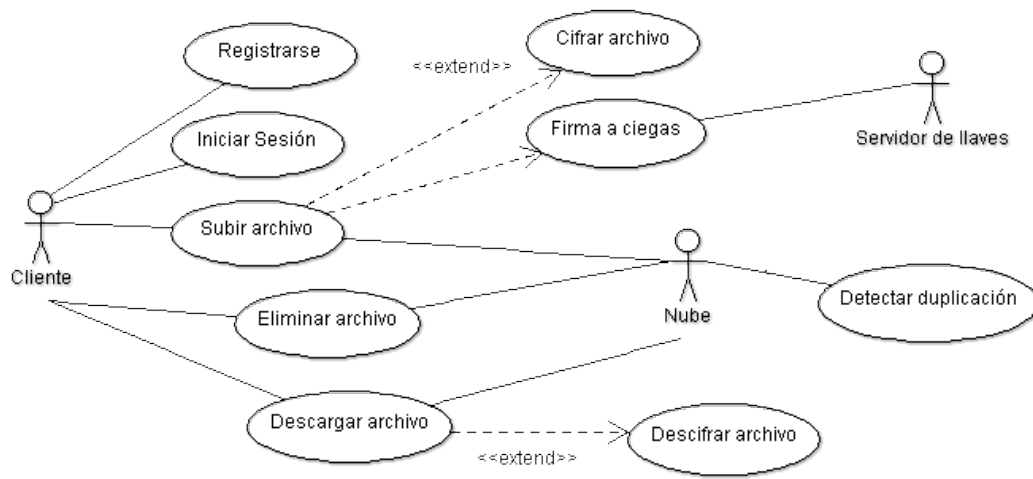


Figura 5.1: Diagrama de Casos de Uso del sistema.

### 5.2.1. CUSLL1 Generar las llaves del servidor de llaves

#### Descripción completa




El servidor de llaves realizará un proceso el cuál involucra la implementación de algoritmos criptográficos de clave pública, dichos algoritmos crearán la llave pública  $e$  y la llave privada  $d$ , la cuál servirá para la firma a ciegas de archivos que se almacenarán en la nube.

#### Atributos importantes




<b>Caso de Uso:</b>	CUSLL1 Generar las llaves del servidor de llaves
<b>Versión:</b>	1.0 - 15/04/17
<b>Autor:</b>	Eder Jonathan Aguirre Cruz
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Servidor de Llaves
<b>Actor:</b>	Servidor
<b>Propósito:</b>	Tener las llaves del servidor para poder comenzar el proceso de firma a ciegas de un archivo
<b>Entradas:</b>	
<b>Salidas:</b>	<ul style="list-style-type: none"><li>▪ Llave pública <math>e</math></li><li>▪ Llave privada <math>d</math></li></ul>
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	El servidor de llaves esta listo para realizar formas a ciegas de archivos a almacenar
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>▪ MSG-SLL1 Generación de llaves</li></ul>

#### Trayectorias del Caso de Uso

##### Trayectoria: Principal




- 1  Seleccionar dos números primos aleatorios. [Trayectoria A]
- 2  Encontrar el producto de esos números primos denominado  $N$ .
- 3  Calcular la función de euler  $\varphi(N)$ .



- 4  Elegir un número aleatorio  $e$  menor a  $\varphi(N)$ , tal que ese número sea  **$\gcd(e, \varphi(N))=1$** .  
[Trayectoria B]
  - 5  Elegir un número aleatorio  $d$ , tal que cumpla con la congruencia  $e \cdot d \equiv 1 \pmod{\varphi(N)}$ . [Trayectoria C]
  - 6  Se generan las llaves pública  $e$  y  $d$  y muestra un mensaje MSG-SLL1 Generación de llaves
- - - - *Fin del caso de uso.*




### Trayectoria alternativa A:

**Condición:** Numeros aleatorios iguales

- A1  Seleccionar números aleatorios iguales o no primos.
  - A2  Muestra el Mensaje MSG-SLL2 Números Iguales.
  - A3  Continúa en el paso 1 del CUSLL1.
- - - - *Fin de la trayectoria.*




### Trayectoria alternativa B:

**Condición:** Número aleatorio menor

- B1  Elegir un número aleatorio menor al tamaño establecido de  $\varphi(N)$ .
  - B2  Muestra el Mensaje MSG-SLL3 Número incorrecto
  - B3  Continúa en el paso 4 del CUSLL1.
- - - - *Fin de la trayectoria.*

### Trayectoria alternativa C:

**Condición:** Número aleatorio incorrecto

- C1  Elegir un número aleatorio incongruente con  $e \cdot d \equiv 1 \pmod{\varphi(N)}$
  - C2  Muestra el Mensaje MSG-SLL3 Número incorrecto
  - C3  Continúa en el paso 5 del CUSLL1.
- - - - *Fin de la trayectoria.*

### 5.2.2. CUSLL2 Generar firma ciega (y).

#### Descripción completa





El servidor realizara una firma a ciegas de un archivo solicitado, este archivo ha sido ocultado para que el servidor no sepa de donde proviene o que contiene, esta firma ayudara para la generación de una llave para cifrar el archivo solicitado.

#### Atributos importantes

Caso de Uso: CUSLL2 Generar firma ciega (y).	
<b>Versión:</b>	1.0 - 16/04/17
<b>Autor:</b>	Diana Leslie González Olivier
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Servidor de Llaves
<b>Actor:</b>	Servidor
<b>Propósito:</b>	Que el servidor firme el archivo solicitado sin saber a que cliente corresponde.
<b>Entradas:</b>	Archivo oculto (x)
<b>Salidas:</b>	Firma a ciegas (y)
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	

#### Trayectorias del Caso de Uso

##### Trayectoria: Principal

- 1  Recibe el archivo oculto (x) .
- 2  Firma el archivo generando un nuevo archivo (y).
- 3  Guarda en la base de datos el archivo (y).
- 4  Envía al cliente el archivo (y).

- - - Fin del caso de uso.

### 5.2.3. CUN3 Almacenar archivo cifrado

#### Descripción completa

Guardar un archivo cifrado en el servicio de almacenamiento (Nube) junto con la llave secreta del usuario que solicita almacenar el archivo cuando este sea cargado por primera vez al almacenamiento










#### Atributos importantes

Caso de Uso: CUN3 Almacenar archivo cifrado	
<b>Versión:</b>	1.0 - 15/04/17
<b>Autor:</b>	Eder Jonathan Aguirre Cruz
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Servidor de Llaves
<b>Actor:</b>	Usuario
<b>Propósito:</b>	Almacenar una sólo copia del archivo y reconocer cuando ya existe una copia de este guardada para evitar su almacenamiento.
<b>Entradas:</b>	<ul style="list-style-type: none"><li>■ Archivo cifrado <i>C1</i>.</li><li>■ Llave secreta cifrada <i>C2</i>.</li><li>■ Función hash del archivo cifrado.</li></ul>
<b>Salidas:</b>	Lista de archivos del usuario actualizada.
<b>Precondiciones:</b>	<ul style="list-style-type: none"><li>■ El servicio de almacenamiento debe estar disponible.</li><li>■ El archivo a almacenar debe estar cifrado bajo un algoritmo criptográfico</li></ul>
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>■ El archivo quedará almacenado en el servicio de almacenamiento.</li><li>■ El usuario tendrá actualizada su lista de archivos en la nube.</li></ul>
<b>Reglas del negocio:</b>	

Caso de Uso: CUN3 Almacenar archivo cifrado	
Mensajes:	<ul style="list-style-type: none"> <li>■ MSG1 Operación exitosa.</li> </ul>







## Trayectorias del Caso de Uso

### Trayectoria: Principal

- 1  Envía la función hash del archivo  $H(C1)$ .
  - 2  Recibe la función hash del archivo  $H(C1)$  a almacenar y corrobora la inexistencia de esta. [Trayectoria A]
  - 3  Solicita los archivos cifrados a almacenar  $C1$  y  $C2$ .
  - 4  Selecciona de su carpeta personal los archivos  $C1$  y  $C2$  y da clic en el botón .
  - 5  Almacena los archivos  $C1$  y  $C2$  en la nube.
  - 6  Asocia la función hash  $H(C1)$  al usuario con el archivo  $C1$  y  $C2$ .
  - 7  Actualiza la lista de usuarios y archivos almacenados en la nube.
  - 8  Muestra el mensaje MSG1 Operación exitosa.
- - - Fin del caso de uso.

### Trayectoria alternativa A:

**Condición:** Archivo existente

- A1  Detecta la existencia de la función hash  $H(C1)$  almacenada en la nube.
  - A2  Solicita el archivo cifrado a almacenar  $C2$ .
  - A3  Selecciona de su carpeta personal el archivos  $C2$  que va almacenar en la nube.
  - A4  Almacena el archivo  $C2$  en la nube.
  - A5  Asocia la función hash  $H(C1)$  al usuario con el archivo  $C2$ .
  - A6  Continúa en el paso 7 del CUN3.
- - - Fin de la trayectoria.

#### 5.2.4. CUN4 Descargar archivo cifrado

##### Descripción completa


Descargar un archivo cifrado del servicio de almacenamiento (Nube) junto con la llave secreta del usuario que este tiene almacenada en la nube






##### Atributos importantes

<b>Caso de Uso:</b>	CUN4 Descargar archivo cifrado
<b>Versión:</b>	1.0 - 15/04/17
<b>Autor:</b>	Eder Jonathan Aguirre Cruz
<b>Prioridad:</b>	Media
<b>Módulo:</b>	Servidor de Llaves
<b>Actor:</b>	Usuario
<b>Propósito:</b>	Entregar al usuario archivos que desea obtener para un uso posterior.
<b>Entradas:</b>	Nombre archivo a descargar
<b>Salidas:</b>	<ul style="list-style-type: none"><li>▪ Archivo cifrado <i>C1</i></li><li>▪ Archivo cifrado <i>C2</i></li></ul>
<b>Precondiciones:</b>	<ul style="list-style-type: none"><li>▪ El usuario debe tener el nombre del archivo que desea descargar</li><li>▪ El archivo a descargar debe estar almacenado en la nube</li></ul>
<b>Postcondiciones:</b>	El archivo estará descargado para que el usuario pueda descifrarlo
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>▪ MSG1 Operación exitosa</li><li>▪ MSG - N1 Archivo no encontrado</li></ul>

##### Trayectorias del Caso de Uso






##### Trayectoria: Principal

- 1  Selecciona de su lista de archivos en la nube el nombre del archivo que desea descargar.

- 2  Recibe el nombre del archivo a descargar.
  - 3  Realiza la búsqueda del archivo asociado al nombre que recibió. [Trayectoria A]
  - 4  Envía los archivos encontrados *C1* y *C2*.
  - 5  Recibe los archivos *C1* y *C2* asociados al nombre que selecciono.
  - 6  Muestra el mensaje MSG1 Operación exitosa.
- - - *Fin del caso de uso.*

## Trayectoria alternativa A:

**Condición:** Archivo inexistente

- A1  Detecta la inexistencia del nombre recibido por el usuario.
  - A2  Muestra el mensaje MSG - N1 Archivo no encontrado.
  - A3  Da clic en el botón 
  - A4  Termina el caso de uso.
- - - *Fin de la trayectoria.*

### 5.2.5. CUN5 Eliminar archivo cifrado

#### Descripción completa






Eliminar un archivo cifrado del servicio de almacenamiento (Nube)


#### Atributos importantes

<b>Caso de Uso:</b> CUN5 Eliminar archivo cifrado	
<b>Versión:</b>	1.0 - 15/04/17
<b>Autor:</b>	Diana Leslie González Olivier
<b>Prioridad:</b>	Media
<b>Módulo:</b>	Servidor de Llaves
<b>Actor:</b>	Usuario
<b>Propósito:</b>	Eliminar archivos que el usuario ya no desea almacenar en la Nube para un uso posterior.
<b>Entradas:</b>	Nombre de archivo a eliminar
<b>Salidas:</b>	
<b>Precondiciones:</b>	<ul style="list-style-type: none"><li>■ El usuario debe tener el nombre del archivo que desea eliminar</li><li>■ El archivo a eliminar debe estar almacenado en la nube</li></ul>
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG1 Operación exitosa</li><li>■ MSG - N1 Archivo no encontrado</li></ul>

#### Trayectorias del Caso de Uso






##### Trayectoria: Principal

- 1  Selecciona de su lista de archivos en la nube el nombre del archivo que desea eliminar.
- 2  Recibe el nombre del archivo a eliminar
- 3  Realiza la búsqueda del archivo asociado al nombre que recibió. [Trayectoria A]
- 4  Elimina los archivos encontrados *C1* y *C2*.
- 5  Actualiza la lista de usuarios de ese archivo.

- 6  Muestra el mensaje MSG1 Operación exitosa.  
- - - *Fin del caso de uso.*

### **Trayectoria alternativa A:**

**Condición:** Archivo inexistente

- A1  Detecta la inexistencia del nombre recibido por el usuario.  
A2  Muestra el mensaje MSG - N1 Archivo no encontrado.  
A3  Da clic en el botón   
A4  Termina el caso de uso.  
- - - *Fin de la trayectoria.*



### 5.2.6. CUCL1 Subir archivo

#### Descripción completa























El usuario seleccionara el archivo que desea que sea almacenado en la nube, este archivo será cifrado y enviado de manera transparente para el usuario, y dependiendo si el archivo se detecta duplicado se enviaran distintos archivos.

#### Atributos importantes

Caso de Uso: CUCL1 Subir archivo	
<b>Versión:</b>	1.0 - 19/04/17
<b>Autor:</b>	Jhonatan Saulés Cortés
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Cliente
<b>Actor:</b>	Usuario
<b>Propósito:</b>	Almacenar un archivo en la nube, el cual debe estar cifrado para que no lo pueda entender el servicio de almacenamiento.
<b>Entradas:</b>	<ul style="list-style-type: none"><li>■ Archivo a almacenar <math>M</math></li><li>■ Llave pública del servidor <math>d</math></li></ul>
<b>Salidas:</b>	Archivo X a firmar por el servidor de llaves
<b>Precondiciones:</b>	El servidor de llaves debe tener asignada tanto su llave pública como llave privada.
<b>Postcondiciones:</b>	El archivo del usuario estará listo para ser firmado por el servidor de llaves.
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG-CL1 Carpeta vacía</li><li>■ MSG-CL2 Archivo incompatible</li><li>■ MSG-CL3 Número incorrecto</li><li>■ MSG-CL4 Error al generar la llave.</li><li>■ MSG-CL5 Archivo almacenado.</li></ul>




## Trayectorias del Caso de Uso

### Trayectoria: Principal

- 1  Da un clic en la opción "Subir Archivo.<sup>en</sup> la pantalla .
  - 2  Despliega una ventana con la carpeta personal que muestra los archivos del usuario. [Trayectoria A]
  - 3  Selecciona el archivo ( $M$ ) que va a subir y da un clic en el botón  en la pantalla . [Trayectoria B]
  - 4  Elige un número aleatorio  $r$  dentro del campo de números primos de igual o menor tamaño a las llaves del servidor de llaves. [Trayectoria C]
  - 5  Calcula una función hash del archivo seleccionado  $H(M)$ .
  - 6  Eleva el número aleatorio  $r$  a la potencia llave pública del servidor de llaves,  $r^e$ .
  - 7  Multiplica  $H(M)$  por  $r^e$ ,  $H(M) \cdot r^e$ .
  - 8  Obtiene Archivo X a firmar y lo envía al servidor de llaves.
  - 9  Recibe la firma a ciegas  $Y$  del servidor.
  - 10  Calcula el inverso multiplicativo del numero aleatorio  $r$ .
  - 11  Multiplica  $Y$  por  $r^{-1}$ ,  $Y \cdot r^{-1}$ .
  - 12  Verifica que  $k^e$  sea igual a  $H(M)$ . [Trayectoria D]
  - 13  Obtiene llave  $k$  y la almacena, junto con el nombre del achivo al que le corresponde.
  - 14  Cifra el archivo ( $M$ ) con su llave  $k$ .
  - 15  Obtiene el archivo  $C1$ .
  - 16  Cifra el archivo  $k$  con su llave publica del usuario  $ka$ .
  - 17  Obtiene el archivo  $C2$ .
  - 18  Envía a la nube el hash del archivo  $C1$ ,  $H(C1)$ .
  - 19  Recibe solicitud de archivos. [Trayectoria E]
  - 20  Envía los archivos  $C1$  y  $C2$ .
  - 21  Muestra el Mensaje MSG-CL5 Archivo almacenado.
- - - Fin del caso de uso.




### Trayectoria alternativa A:

Condición: Archivos inexistentes

- A1  Despliega una ventana con la carpeta personal del usuario sin archivos existentes.
  - A2  Muestra el Mensaje MSG-CL1 Carpeta vacía.
  - A3  Termina el caso de uso.
- - - Fin de la trayectoria.




### Trayectoria alternativa B:

**Condición:** Archivo incompatible

- B1**  Selecciona el archivo  $(M)$  en un formato incompatible para el protocolo y su almacenamiento
- B2**  Muestra el Mensaje MSG-CL2 Archivo incompatible.
- B3**  Termina el caso de uso.
- - - *Fin de la trayectoria.*




### Trayectoria alternativa C:

**Condición:** Número aleatorio incorrecto

- C1**  Elegir un número aleatorio no primo o mayor al tamaño de las llaves del servidor de llaves.
- C2**  Muestra el Mensaje MSG-CL3 Número incorrecto.
- C3**  Continúa en el paso 4 del CUCL2.
- - - *Fin de la trayectoria.*




### Trayectoria alternativa D:

**Condición:** Comparacion es diferente

- D1**  Detecta que  $k^e$  y  $H(M)$  son diferentes.
- D2**  Muestra el Mensaje MSG-CL4 Error al generar la llave.
- D3**  Continúa en el paso 4 del CUCL2.
- - - *Fin de la trayectoria.*

### Trayectoria alternativa E:

**Condición:** Archivo duplicado

- E1**  Detecta que el archivo  $H(C1)$  ya ha sido almacenado.
- E2**  Envía el archivo  $C2$ .
- E3**  Continúa en el paso 21 del CUCL2.
- - - *Fin de la trayectoria.*

### 5.2.7. CUCL3 Descargar archivos descifrados.

#### Descripción completa









El cliente podrá descargar su archivo y descifrarlo.

#### Atributos importantes



<b>Caso de Uso:</b> CUCL3 Descargar archivos descifrados.	
<b>Versión:</b>	1.0 - 16/04/17
<b>Autor:</b>	Diana Leslie González Olivier
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Cliente
<b>Actor:</b>	Cliente
<b>Propósito:</b>	Que el cliente pueda obtener su archivo con texto en claro
<b>Entradas:</b>	C1 y C2
<b>Salidas:</b>	Archivo descargado
<b>Precondiciones:</b>	El archivo debe existir en la Nube
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG1 Operación exitosa</li><li>■ MSG-CL6 Archivo inexistente</li></ul>

#### Trayectorias del Caso de Uso

##### Trayectoria: Principal

- 1  Selecciona el archivo a descargar y da clic en la opción de descargar archivo.
  - 2  Envía a la nube una petición con el nombre del archivo que desea descargar.
  - 3  Recibe los archivos *C1* y *C2* asociados al nombre que envió.
  - 4  Descifra *C2* con la llave *Ka* del cliente.
  - 5  Obtiene un archivo con la llave *K*.
  - 6  Descifra *C1* con la llave *K*.
  - 7  Obtiene su archivo *M* con su información visible.
  - 8  Muestra el mensaje MSG1 Operación exitosa.
- - - Fin del caso de uso.

## Trayectoria: Trayectoria Alternativa

- 1  Envía a la nube una petición con el nombre del archivo que desea descargar.
  - 2  Muestra el mensaje MSG-CL4 Archivo inexistente.
- - - - *Fin del caso de uso.*

### 5.2.8. CUCL4 Eliminar archivos cifrado.

#### Descripción completa









El cliente podrá elegir la opción de eliminar un archivo cifrado del servicio de almacenamiento en la nube.

#### Atributos importantes

<b>Caso de Uso:</b> CUCL4 Eliminar archivos cifrado.	
<b>Versión:</b>	1.0 - 16/04/17
<b>Autor:</b>	Diana Leslie González Olivier
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Cliente
<b>Actor:</b>	Cliente
<b>Propósito:</b>	Que el cliente pueda eliminar un archivo
<b>Entradas:</b>	
<b>Salidas:</b>	Archivo eliminado
<b>Precondiciones:</b>	El archivo debe existir en la Nube
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG1 Operación exitosa</li></ul>

#### Trayectorias del Caso de Uso

##### Trayectoria: Principal

- 1  El cliente da clic en el botón eliminar archivo.
  - 2  El sistema despliega la pantalla para seleccionar el archivo que se desea eliminar.
  - 3  El cliente selecciona el archivo que desea eliminar.
  - 4  El sistema recibe petición para eliminar archivo.
  - 5  El sistema busca el nombre de archivo asociado en su base de datos.[Trayectoria A]
  - 6  El sistema elimina C1 y C2.
  - 7  El sistema despliega la lista de usuarios en la base de datos.
  - 8  Muestra el mensaje MSG1 Operación exitosa.
- - - Fin del caso de uso.

## Trayectoria alternativa A:

**Condición:** Archivo inexistente

**A1**  El cliente da clic en el botón .

**A2**  El sistema despliega la pantalla principal.

- - - - *Fin de la trayectoria.*

### 5.2.9. CUCL6 Iniciar Sesión.

#### Descripción completa



Permitir el acceso al sistema con su usuario y contraseña correspondientes, el cual es autenticado y autorizado para la utilización del sistema.

#### Atributos importantes









<b>Caso de Uso:</b> CUCL6 Iniciar Sesión.	
<b>Versión:</b>	1.0 - 19/04/17
<b>Autor:</b>	Jhonatan Saulés Cortés.
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Cliente
<b>Actor:</b>	Cliente
<b>Propósito:</b>	Dar acceso al usuario al sistema para poder realizar sus actividades.
<b>Entradas:</b>	Nombre de usuario, Contraseña.
<b>Salidas:</b>	Página principal del usuario que inicio sesión
<b>Precondiciones:</b>	Estar registrado en el sistema.
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	<ul style="list-style-type: none"><li>■ RN4 Usuario registrado</li></ul>
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG1 Operación exitosa.</li><li>■ MSG5 Dato incorrecto.</li><li>■ MSG6 Longitud inválida.</li><li>■ MSG9 Dato requerido.</li><li>■ MSG10 No existe información.</li><li>■ MSG11 Contraseña incorrecta</li></ul>

#### Trayectorias del Caso de Uso

#### Trayectoria: Principal




- 1  Da clic en la opción *Iniciar sesión*.
- 2  Despliega los campos para introducir nombre de usuario y contraseña.



- 3  Ingresa su nombre de usuario y contraseña en los campos mostrados.
  - 4  Da clic en el botón *Ingresar*.
  - 5  Autentica y autoriza el nombre usuario y contraseña con base en la regla de negocio RN4 Usuario registrado. [Trayectoria A] [Trayectoria B] [Trayectoria C] [Trayectoria D] [Trayectoria E]
  - 6  Solicita las llaves privada y pública del usuario.
  - 7  Almacena las llaves en el dispositivo actual.
  - 8  Muestra el mensaje MSG1 Operación exitosa.
  - 9  Muestra el menú principal del usuario.
  - 10  Fin del caso de uso.
- - - *Fin del caso de uso.*




### Trayectoria alternativa A:

Condición: Datos incorrectos

- A1  Muestra el mensaje MSG5 Dato incorrecto.
  - A2  Da clic en el botón *Cerrar*.
  - A3  Continúa en el paso 3 del CUCL6
- - - *Fin de la trayectoria.*




### Trayectoria alternativa B:

Condición: Longitud inválida

- B1  Muestra el mensaje MSG6 Longitud inválida.
  - B2  Da clic en el botón *Cerrar*.
  - B3  Continúa en el paso 3 del CUCL6
- - - *Fin de la trayectoria.*

### Trayectoria alternativa C:



Condición: Datos requeridos

- C1  Muestra el mensaje MSG9 Dato requerido.
  - C2  Da clic en el botón *Cerrar*.
  - C3  Continúa en el paso 3 del CUCL6
- - - *Fin de la trayectoria.*

### Trayectoria alternativa D:




Condición: No existe información

- D1  Muestra el mensaje MSG10 No existe información.

- D2**  Da clic en el botón *Cerrar*.
- D3**  Continúa en el paso 3 del CUCL6
- - - *Fin de la trayectoria.*

### **Trayectoria alternativa E:**

**Condición:** Contraseña incorrecta

- E1**  Muestra el mensaje MSG11 Contraseña incorrecta
- E2**  Da clic en el botón *Cerrar*.
- E3**  Continúa en el paso 3 del CUCL6
- - - *Fin de la trayectoria.*

### 5.2.10. CUCL7 Registrar usuario.

#### Descripción completa


Solicitar los datos importantes de un usuario nuevo, crearle sus llaves pública y privada para darlo de alta en el sistema.










#### Atributos importantes

Caso de Uso: CUCL7 Registrar usuario.	
<b>Versión:</b>	1.0 - 19/04/17
<b>Autor:</b>	Jhonatan Saulés Cortés.
<b>Prioridad:</b>	Alta
<b>Módulo:</b>	Cliente
<b>Actor:</b>	Cliente
<b>Propósito:</b>	Habilitar un nuevo usuario generandole sus llaves privada y pública.
<b>Entradas:</b>	Datos del usuario.
<b>Salidas:</b>	Llaves del usuario
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	
<b>Reglas del negocio:</b>	<ul style="list-style-type: none"><li>■ RN1 Datos requeridos</li><li>■ RN2 Datos correctos</li><li>■ RN3 Unicidad de elementos</li></ul>
<b>Mensajes:</b>	<ul style="list-style-type: none"><li>■ MSG1 Operación exitosa.</li><li>■ MSG4 Registro repetido</li><li>■ MSG5 Dato incorrecto.</li><li>■ MSG6 Longitud inválida.</li><li>■ MSG9 Dato requerido.</li></ul>

#### Trayectorias del Caso de Uso




#### Trayectoria: Principal

- 1  Da clic en la opción *Registrarse*.

- 2  Despliega los campos para introducir nombre, primer apellido, segundo apellido, nombre de usuario, contraseña, fecha de nacimiento, CURP.
  - 3  Ingresa sus datos que han sido solicitados.
  - 4  Da clic en el botón *Registrar*.
  - 5  Verifica los datos proporcionados por el usuario con base en las reglas de negocios RN1 Datos requeridos, RN2 Datos correctos, RN3 Unicidad de elementos . [Trayectoria A] [Trayectoria B] [Trayectoria C] [Trayectoria D]
  - 6  Genera su llave privada y pública del usuario con RSA.
  - 7  Almacena sus llaves en el servidor y en el dispositivo actual.
  - 8  Muestra el mensaje MSG1 Operación exitosa.
  - 9  Muestra el menú principal del usuario.
  - 10  Fin del caso de uso.
- - - - *Fin del caso de uso.*




### Trayectoria alternativa A:

**Condición:** Datos incorrectos

- A1  Muestra el mensaje MSG5 Dato incorrecto.
  - A2  Da clic en el botón *Cerrar*.
  - A3  Continúa en el paso 3 del CUCL7
- - - - *Fin de la trayectoria.*




### Trayectoria alternativa B:

**Condición:** Longitud inválida

- B1  Muestra el mensaje MSG6 Longitud inválida.
  - B2  Da clic en el botón *Cerrar*.
  - B3  Continúa en el paso 3 del CUCL7
- - - - *Fin de la trayectoria.*




### Trayectoria alternativa C:

**Condición:** Datos requeridos

- C1  Muestra el mensaje MSG9 Dato requerido.
  - C2  Da clic en el botón *Cerrar*.
  - C3  Continúa en el paso 3 del CUCL7
- - - - *Fin de la trayectoria.*

### Trayectoria alternativa D:

**Condición:** Registro repetido

- D1  Muestra el mensaje MSG4 Registro repetido
  - D2  Da clic en el botón *Cerrar*.
  - D3  Continúa en el paso 3 del CUCL7
- - - - *Fin de la trayectoria.*

## 5.3. Diagramas de secuencia

### 5.3.1. Registrar Usuario

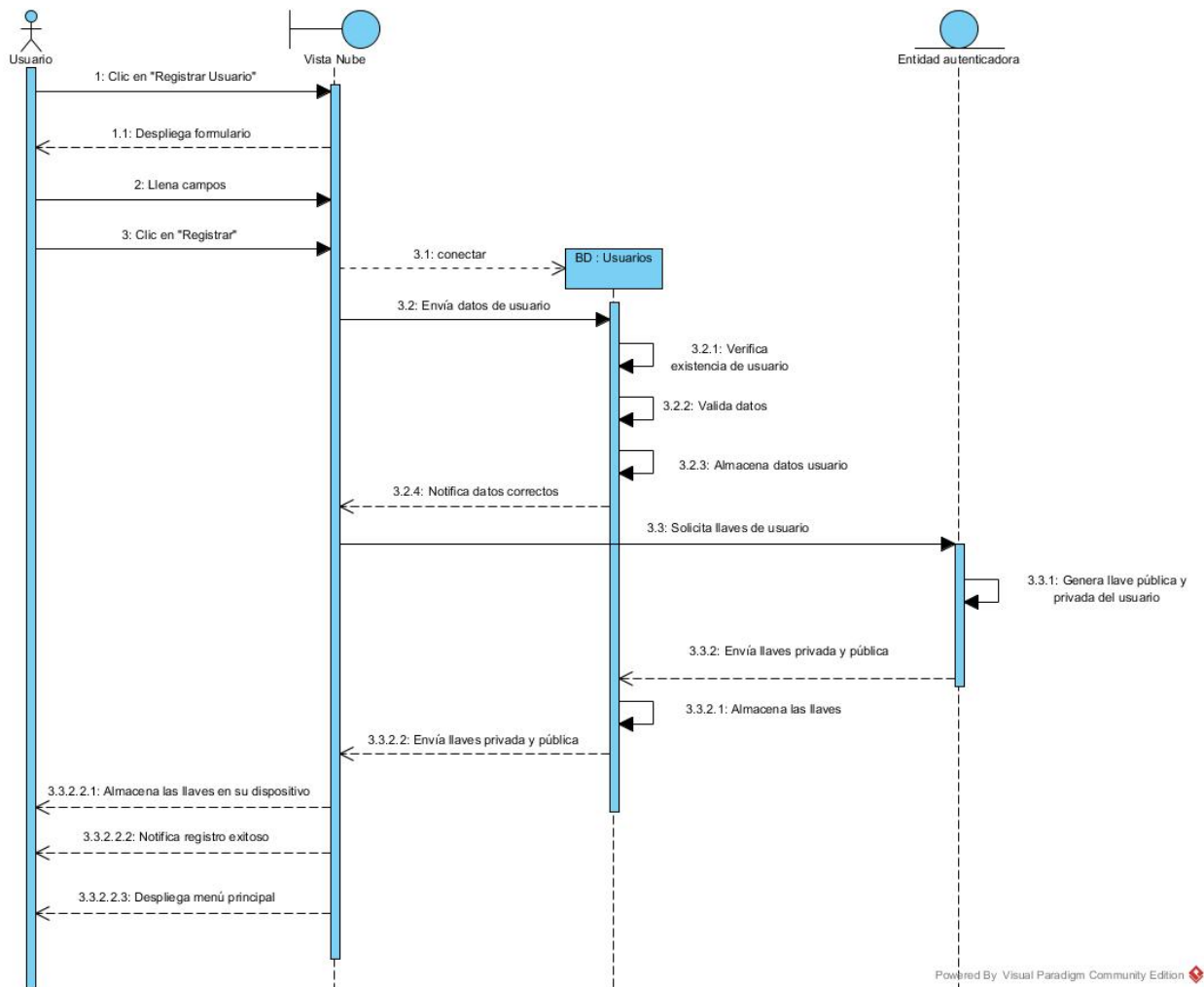


Figura 5.2: Diagrama de secuencias de Registrar un usuario nuevo.

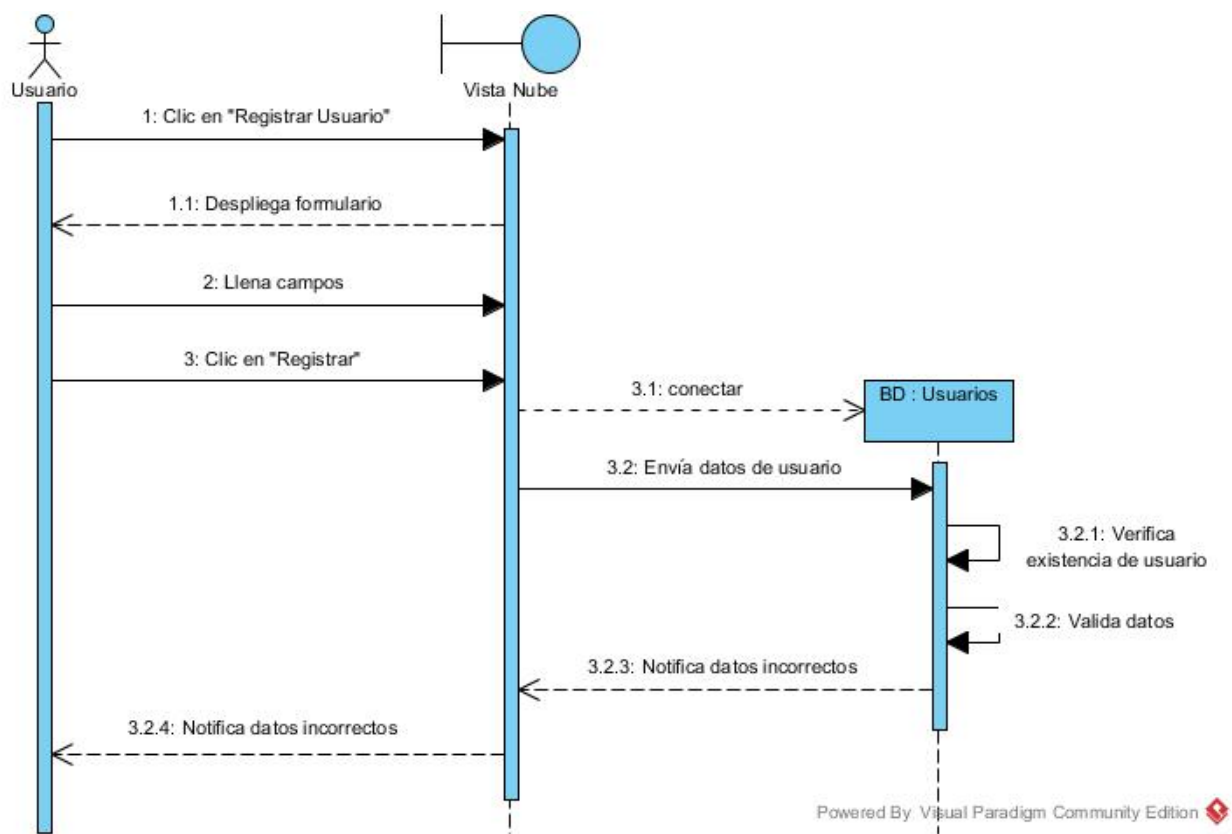


Figura 5.3: Diagrama de secuencias de Registrar un usuario nuevo con datos incorrectos.

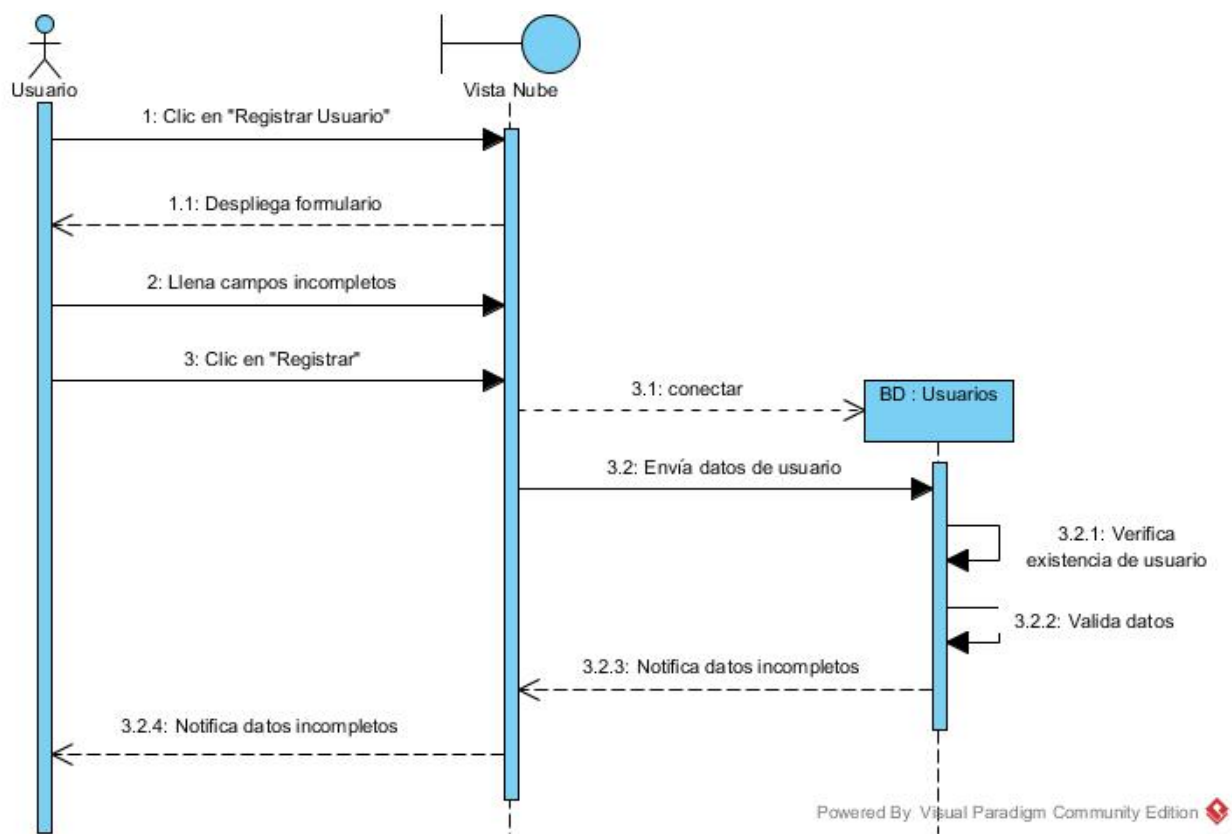


Figura 5.4: Diagrama de secuencias de Registrar un usuario nuevo con datos incompletos.

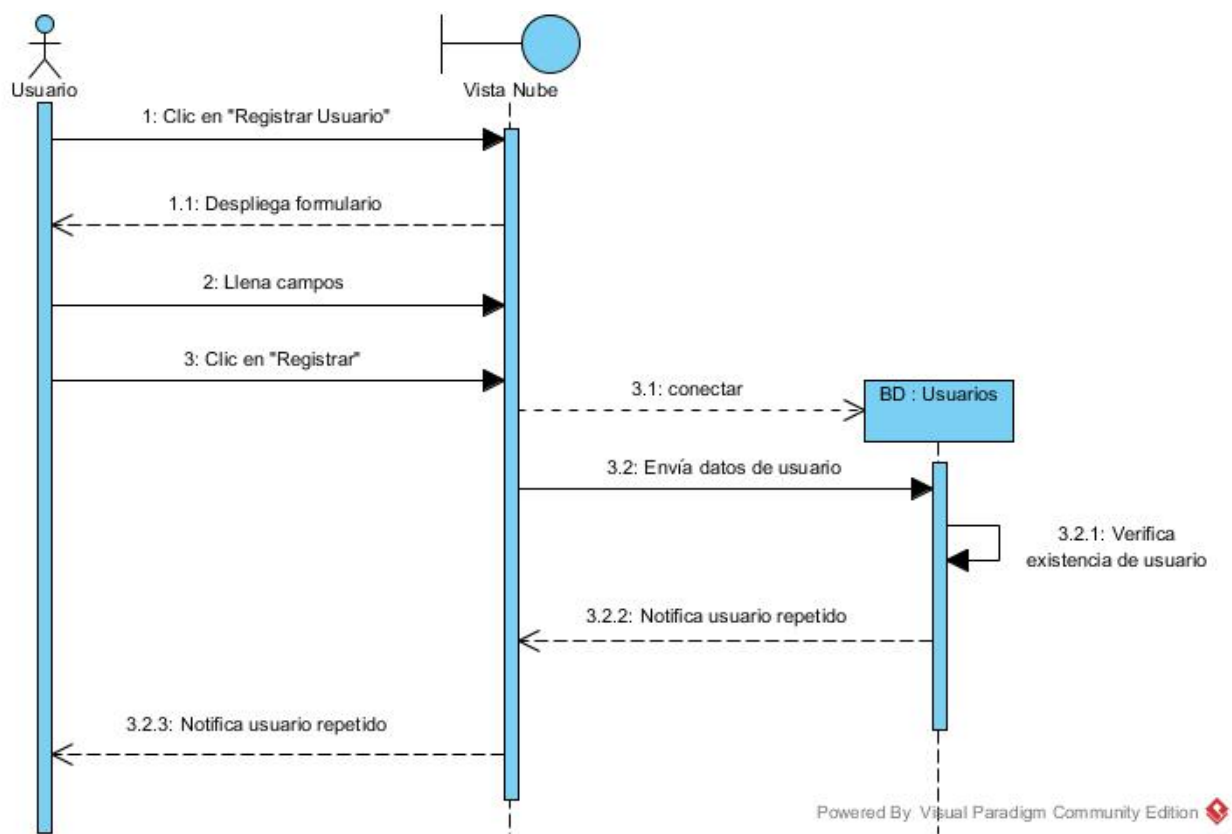


Figura 5.5: Diagrama de secuencias de Registrar un usuario nuevo con usuario repetido.



### 5.3.2. Iniciar Sesión

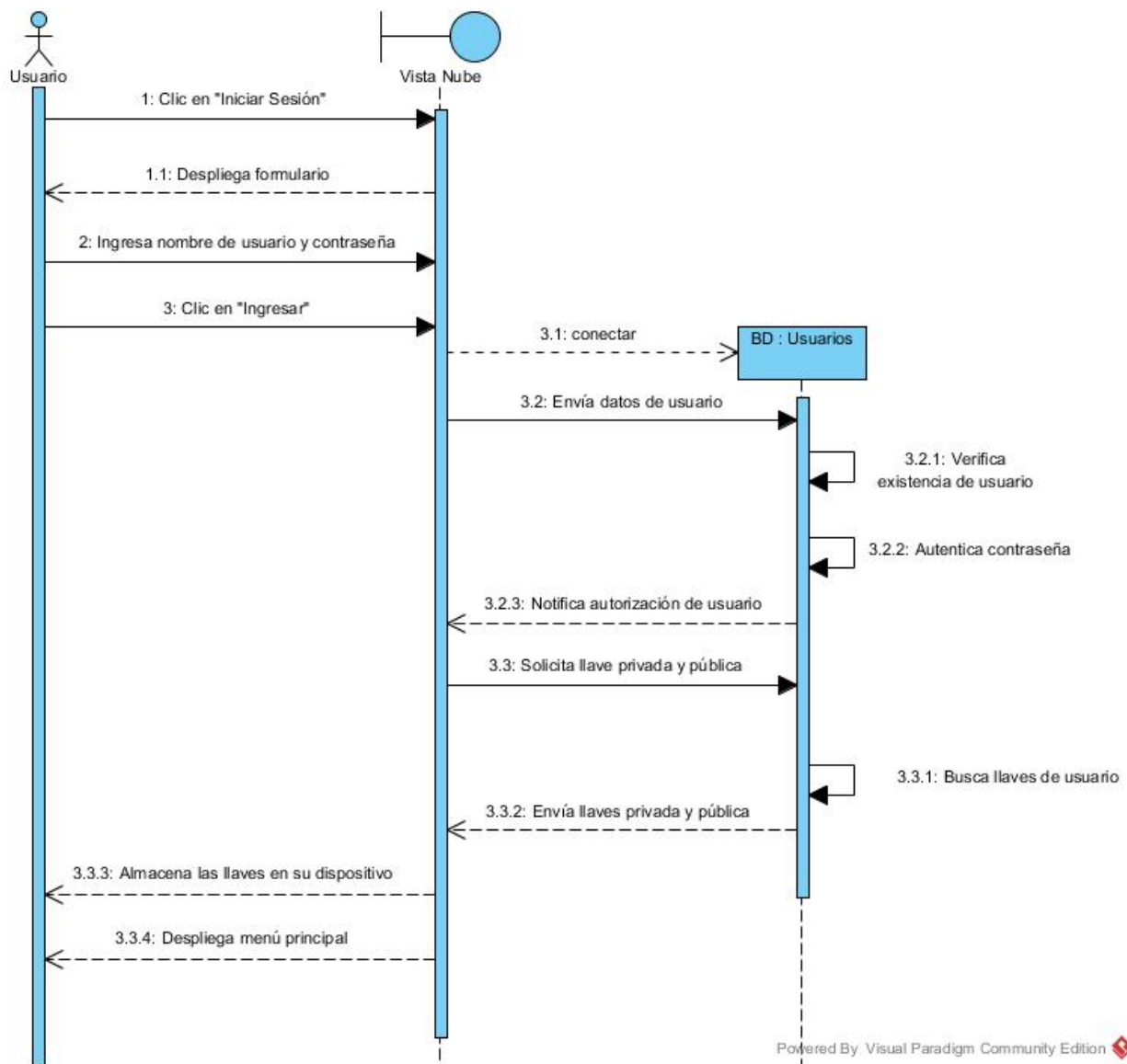


Figura 5.6: Diagrama de secuencias de Iniciar sesion un usuario.

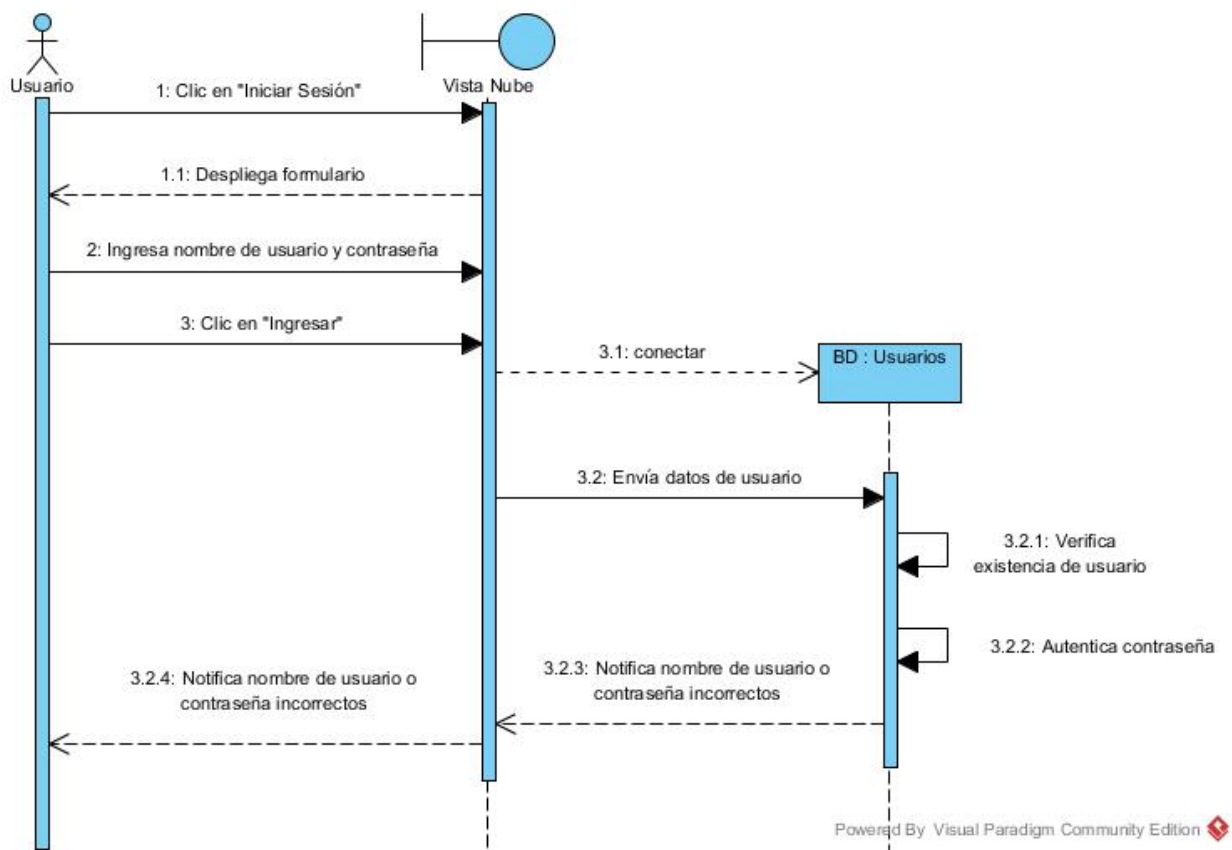


Figura 5.7: Diagrama de secuencias de Inicar sesion un usuario con datos incorrectos.

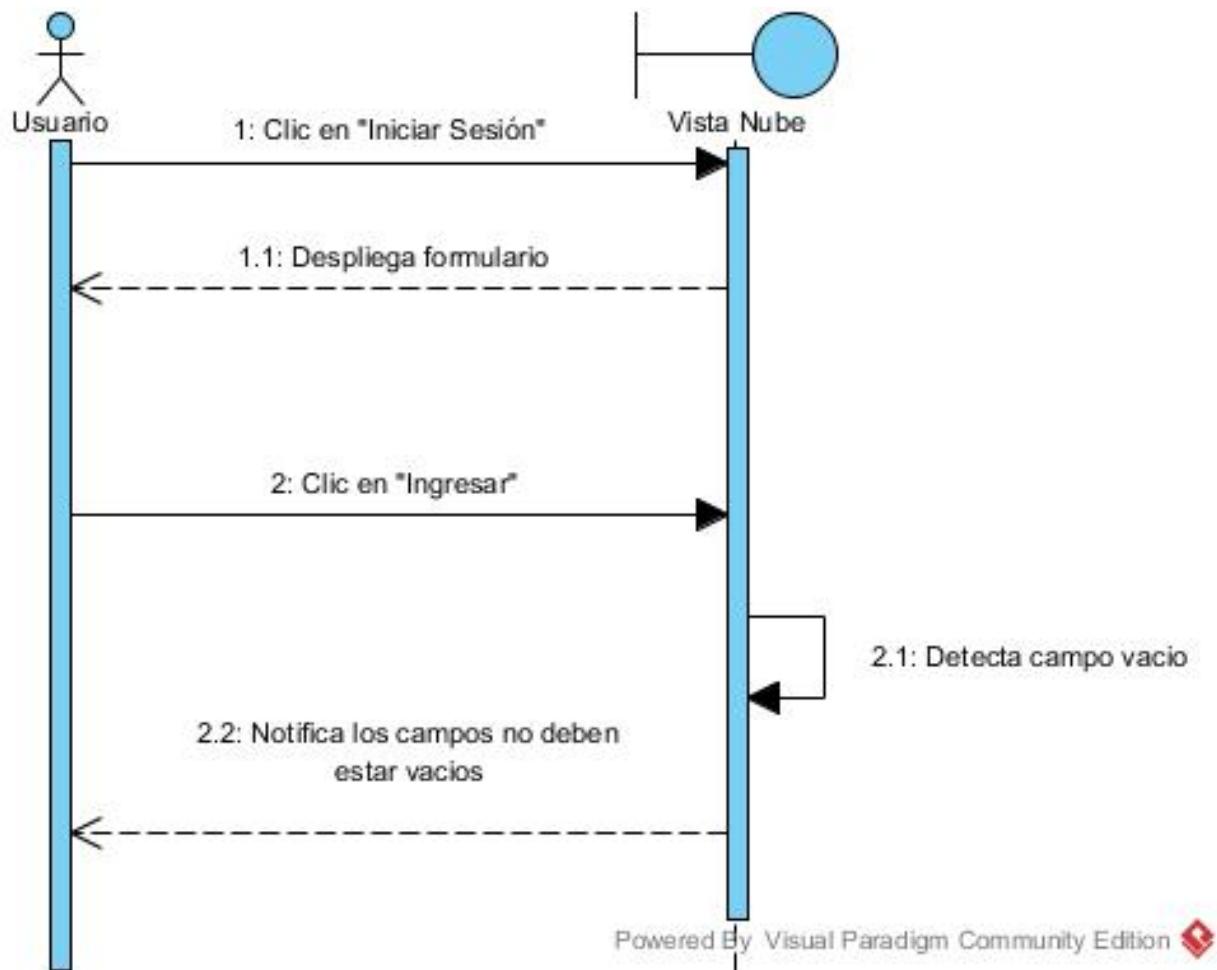


Figura 5.8: Diagrama de secuencias de Registrar un usuario nuevo con datos incompletos.

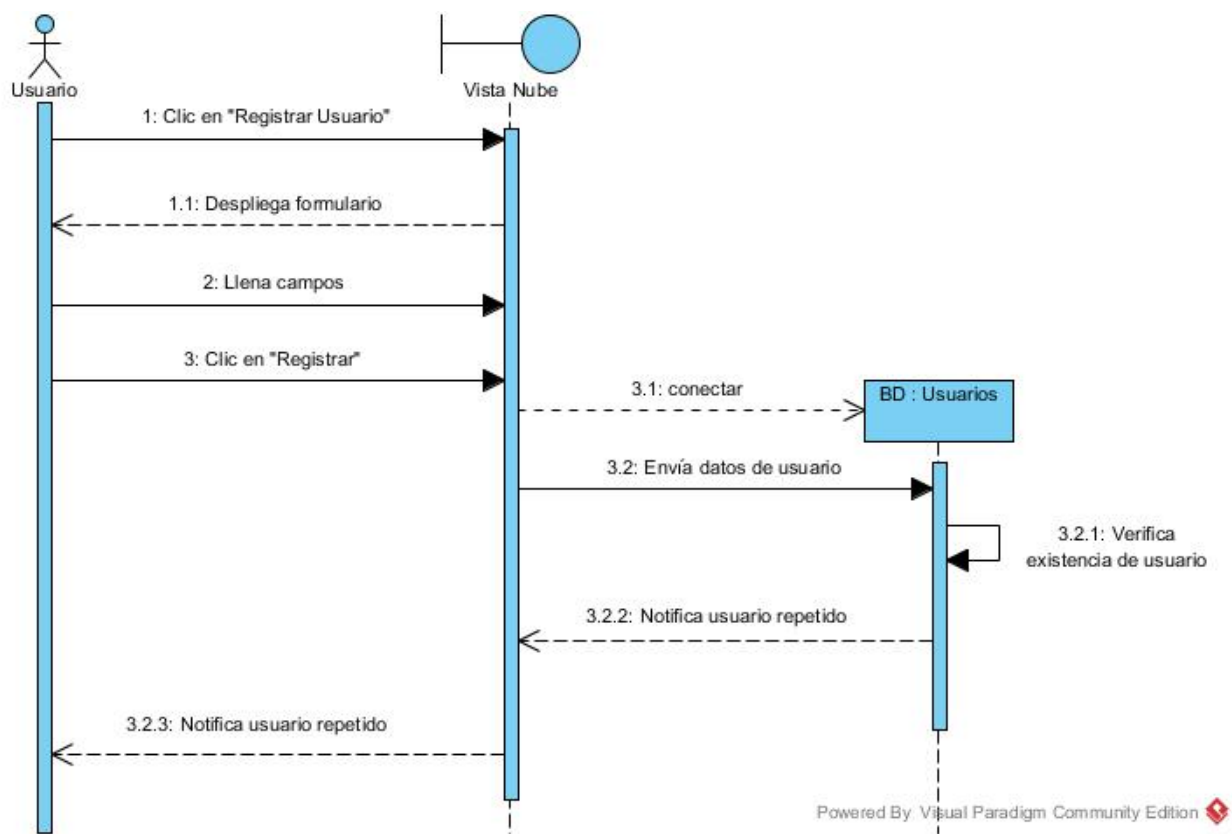


Figura 5.9: Diagrama de secuencias de Inicar sesion un usuario con campos vacios.

### 5.3.3. Subir Archivo

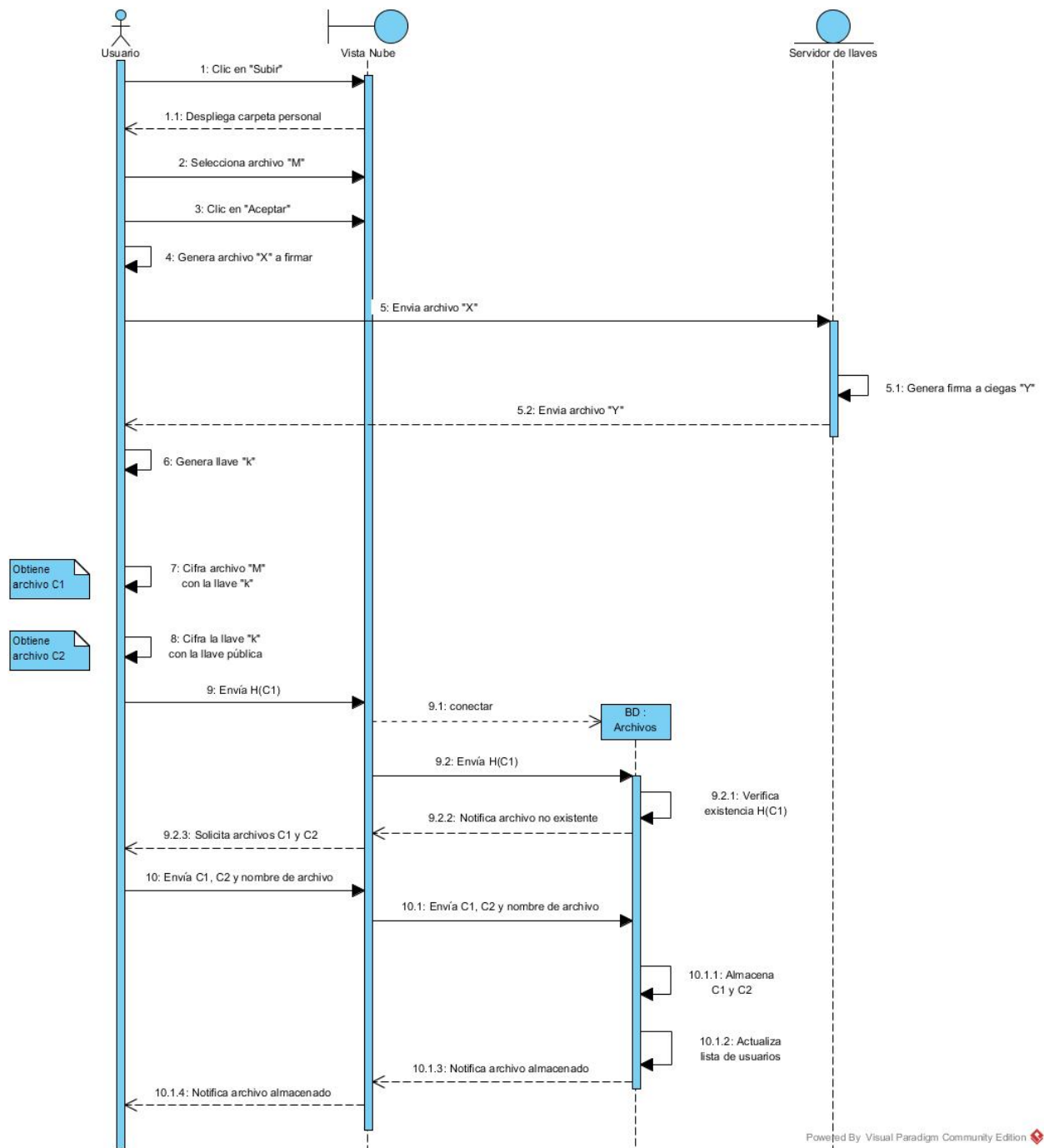


Figura 5.10: Diagrama de secuencias de subir un archivo nuevo.

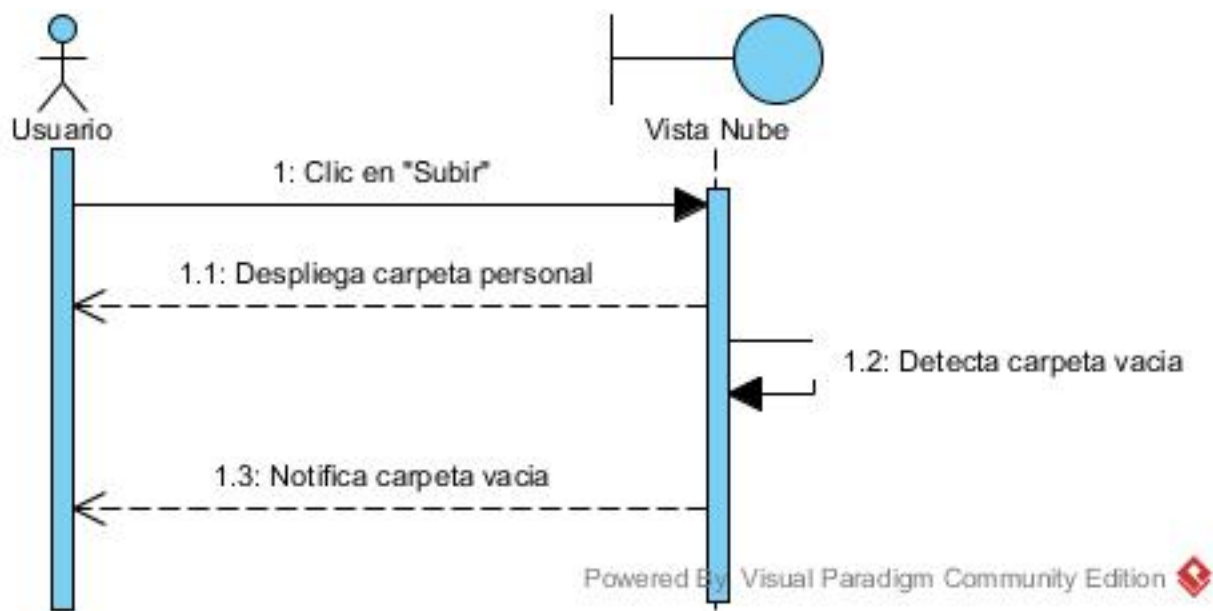


Figura 5.11: Diagrama de secuencias de subir un archivo con carpeta vacia.

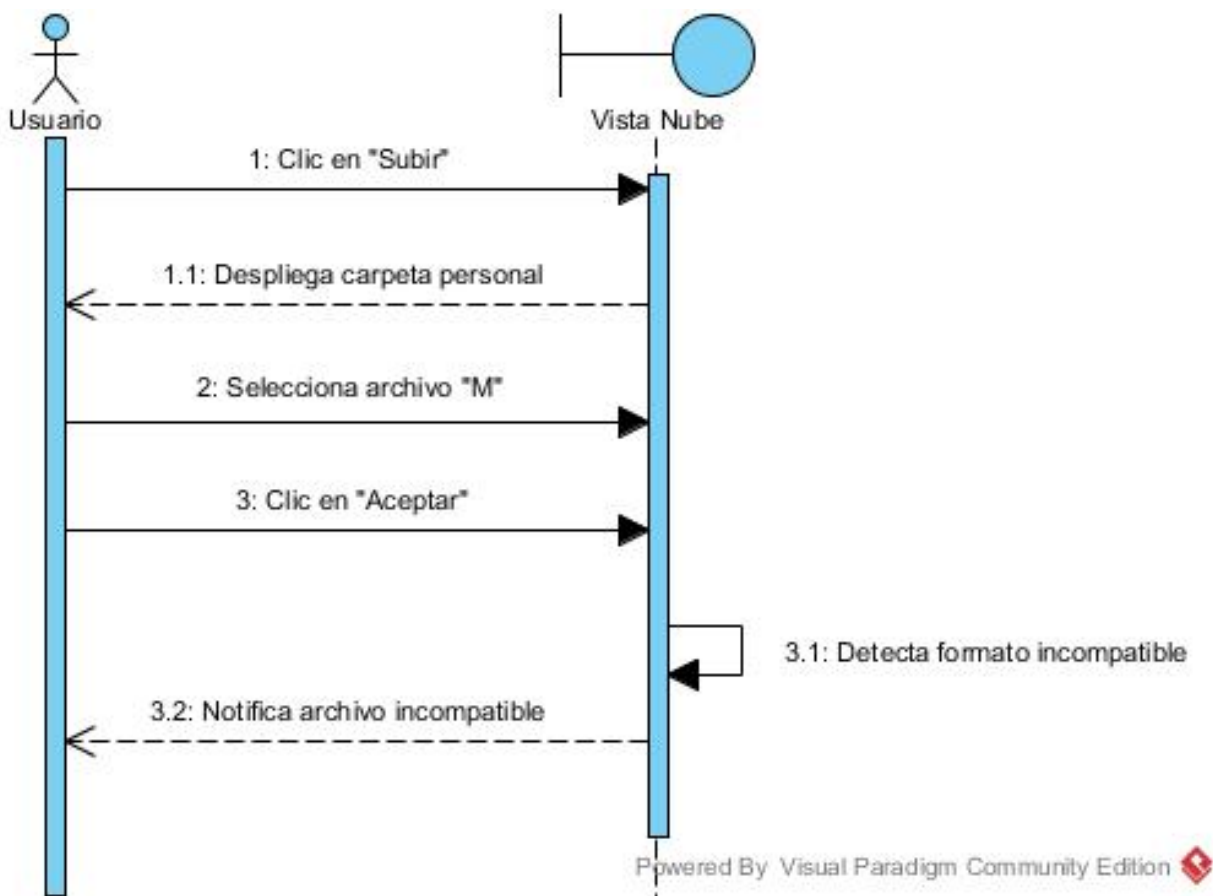


Figura 5.12: Diagrama de secuencias de subir un archivo incompatible.

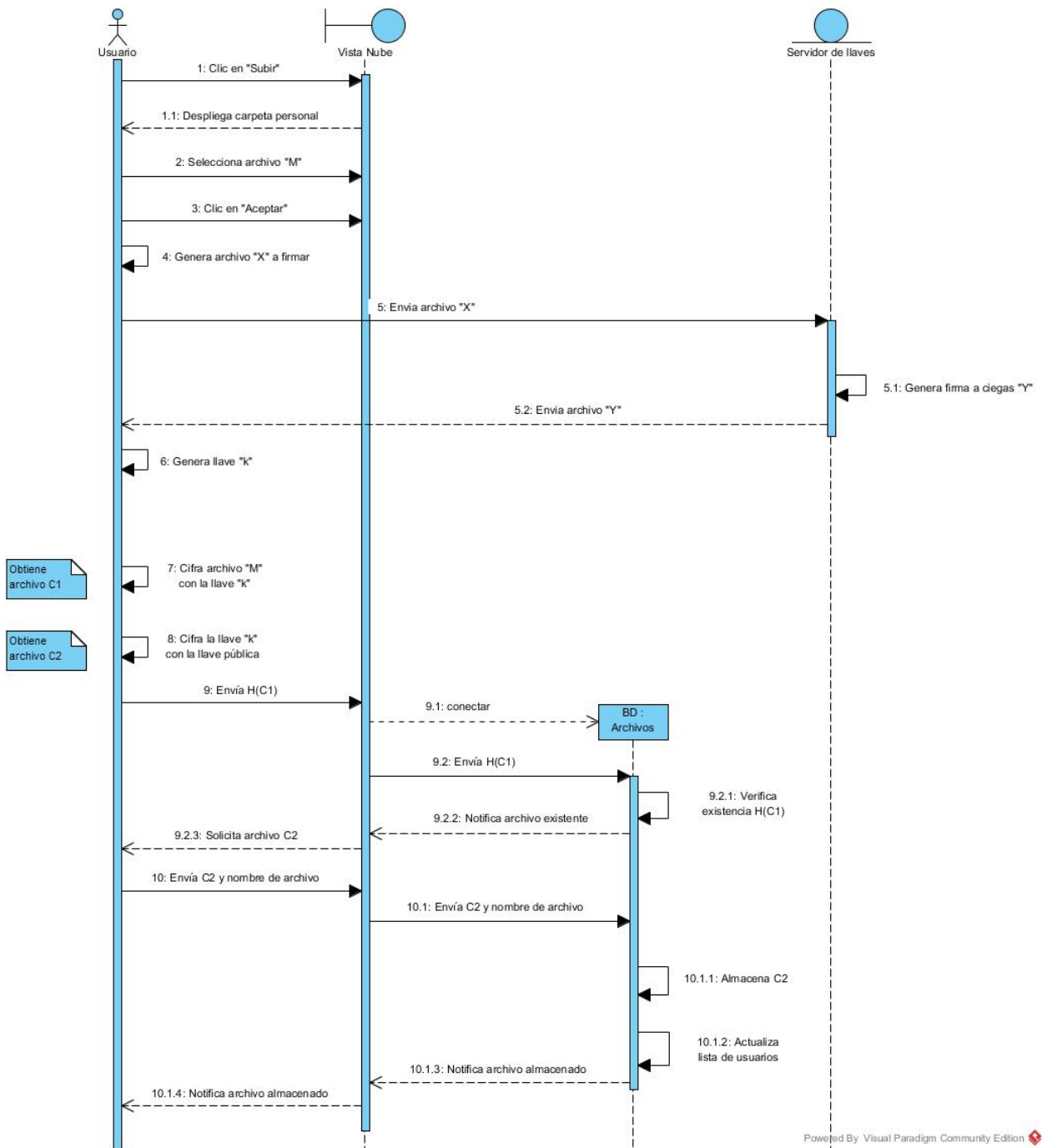


Figura 5.13: Diagrama de secuencias de subir un archivo existente.

### 5.3.4. Firma a ciegas

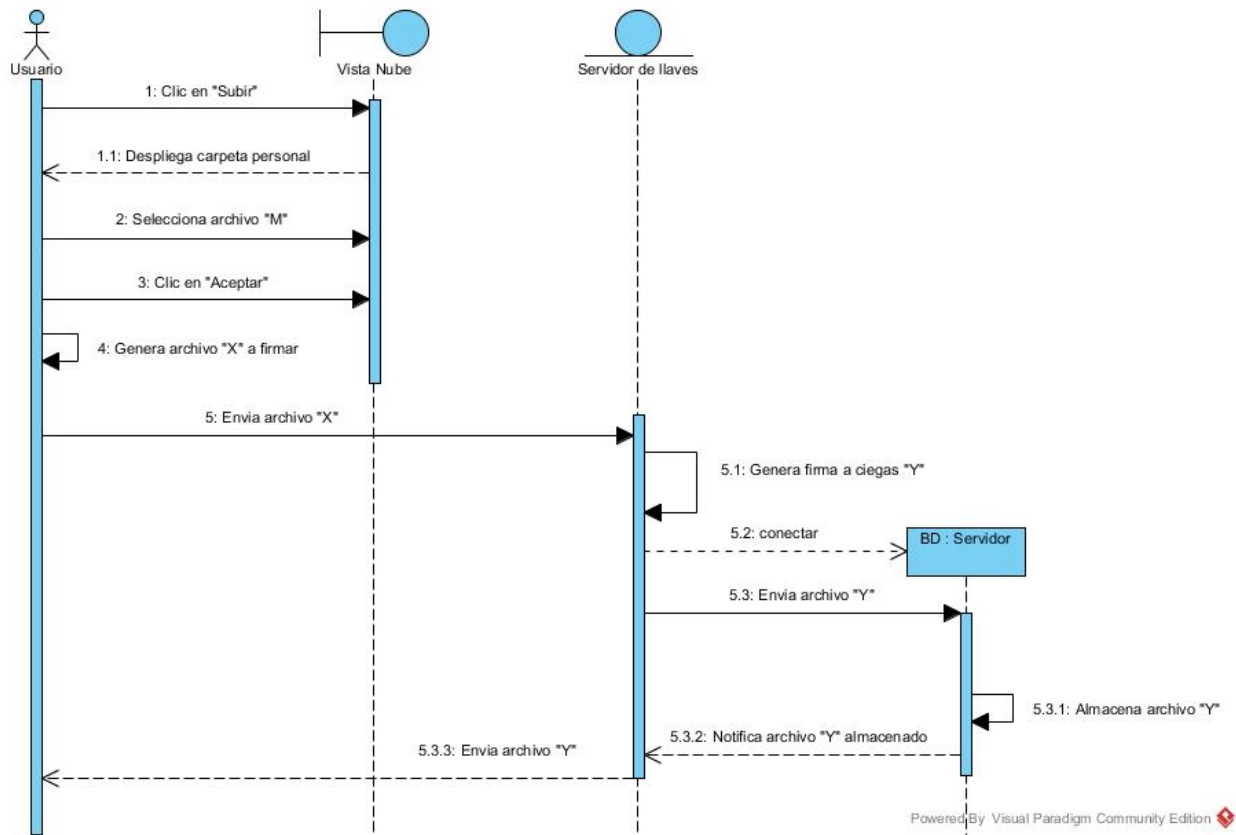


Figura 5.14: Diagrama de secuencias de la firma a ciegas del servidor de llaves.



### 5.3.5. Descargar Archivo

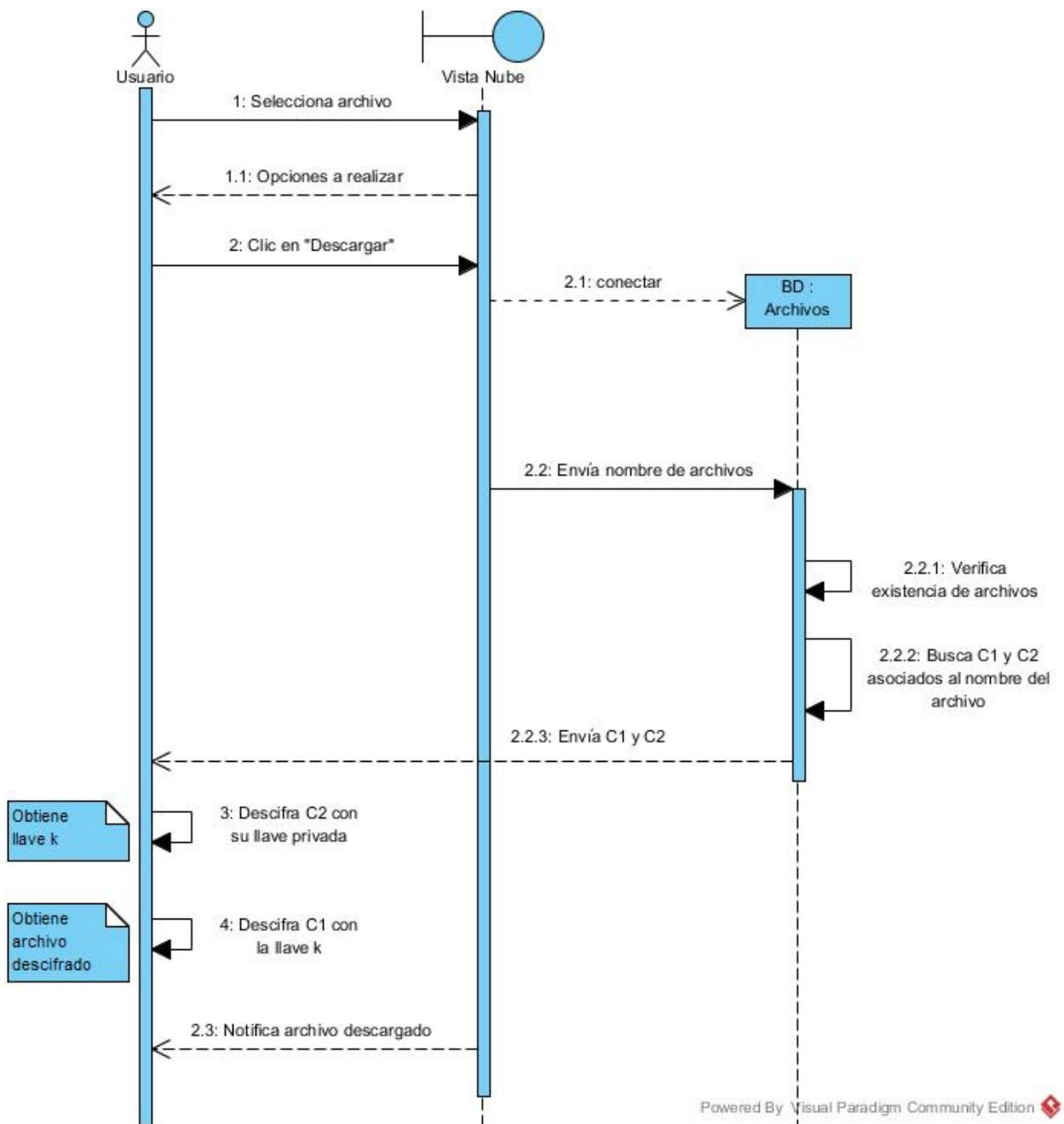


Figura 5.15: Diagrama de secuencias de descargar un archivo de la nube.

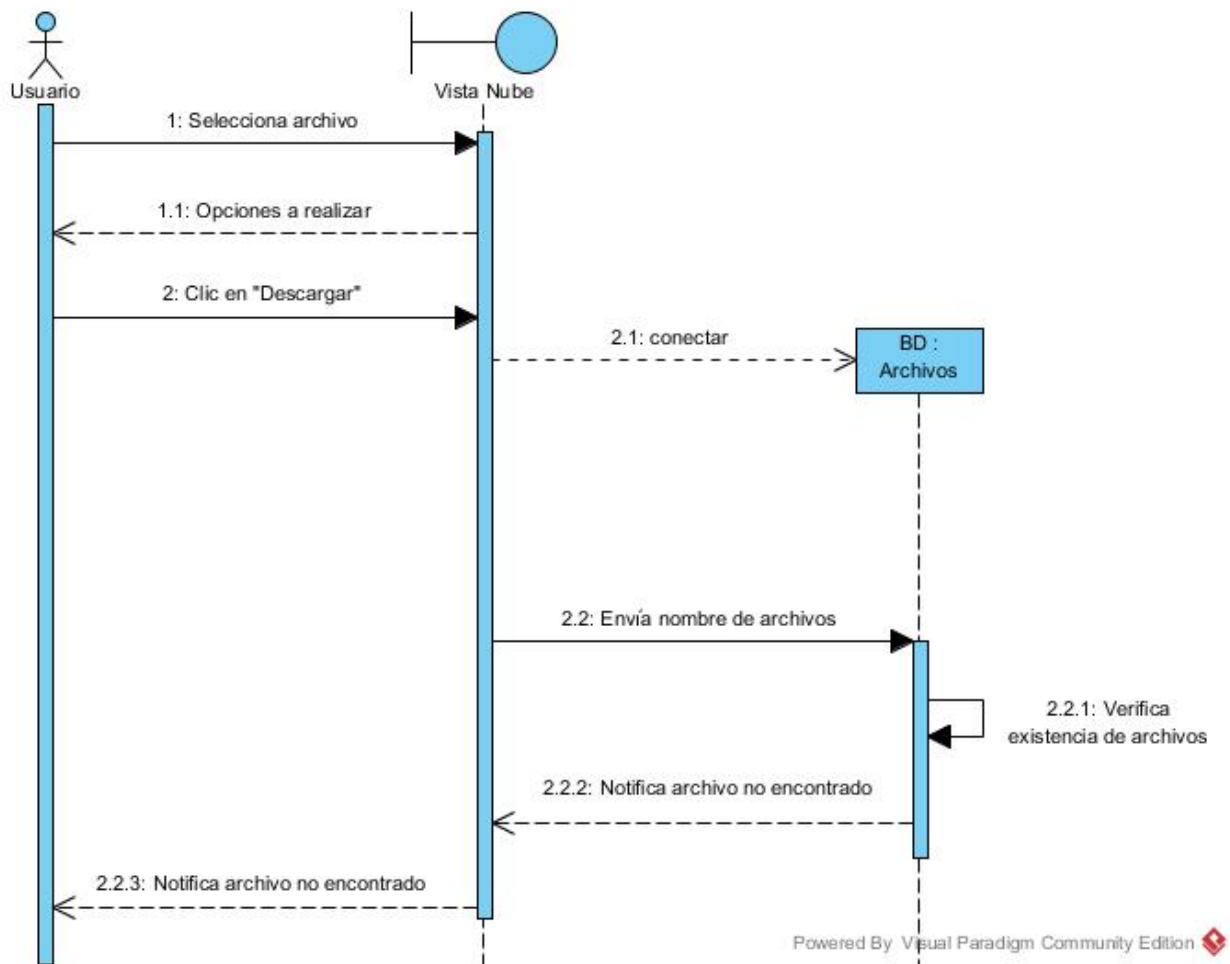


Figura 5.16: Diagrama de secuencias de descargar un archivo de la nube no encontrado.

### 5.3.6. Eliminar Archivo

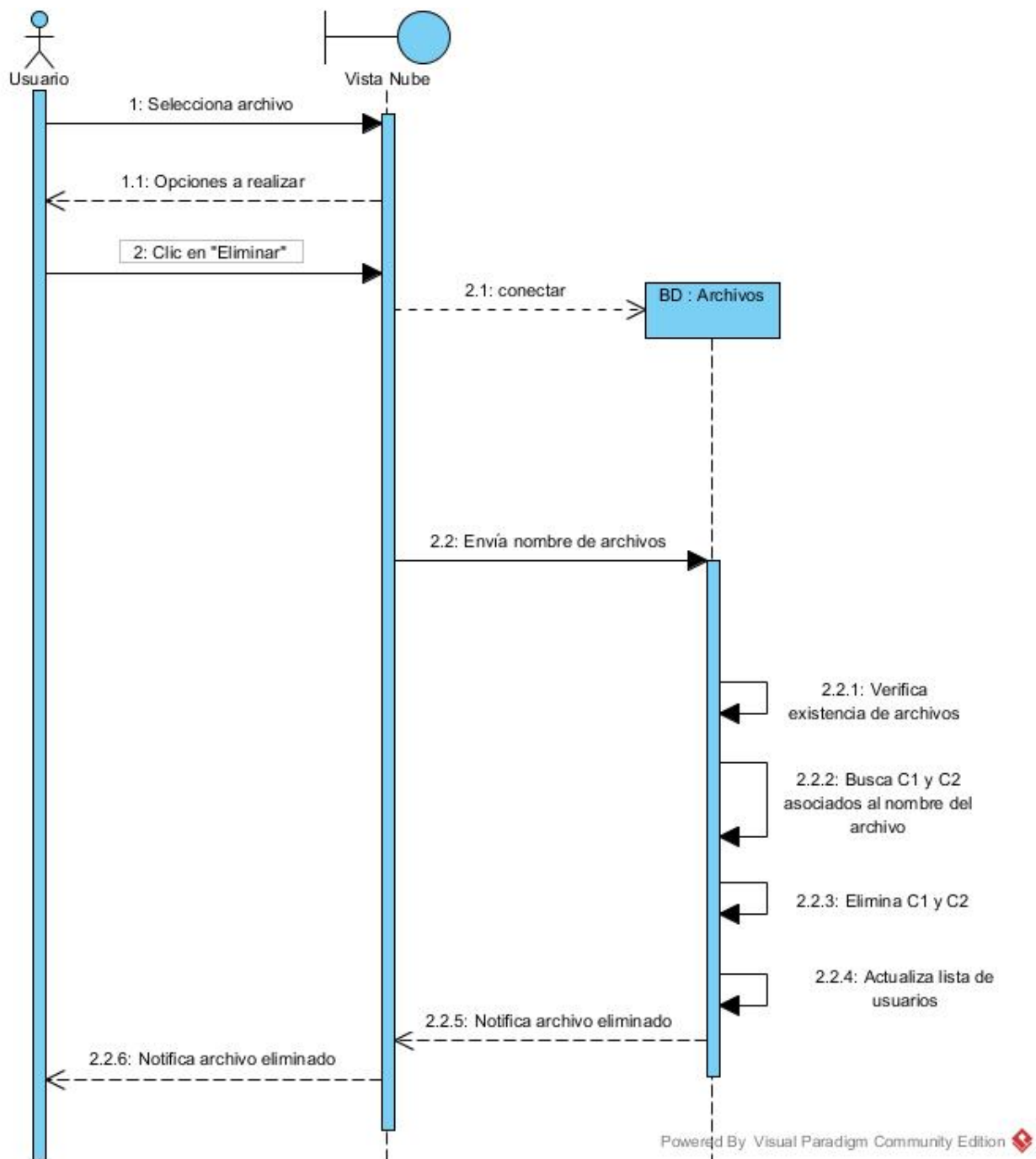


Figura 5.17: Diagrama de secuencias de eliminar un archivo de la nube.

## 5.4. Mensajes del sistema

Mensajes usados en el sistema, que se usan para informar al usuario mediante la interfaz de ciertas situaciones o eventos que ocurren en el prototipo y pueden ser de los siguientes tipos:

- **Notificación** : Estos mensajes se utilizan para indicar que la operación solicitada por el usuario se ejecutó correctamente.
- **Alerta** : Estos mensajes se utilizan para indicar alguna advertencia sobre la operación.
- **Error** : Estos mensajes se utilizan para indicar que ha ocurrido un error en la operación solicitada.

Varios mensajes se encuentran parametrizados. Es decir cuando algún mensaje es recurrente, hay palabras que pueden ser sustituidas por otras para transformar el mensaje a la situación.

Parámetros más comunes:

**ARTÍCULO:** Se refiere a un artículo el cual puede ser DETERMINADO (El | La | Lo | Los | Las) o INDETERMINADO (Un | Una | Uno | Unos | Unas) se aplica generalmente sobre una ENTIDAD, ATRIBUTO o VALOR.

**CAMPO:** Se refiere a un campo del formulario. Por lo regular es el nombre de un atributo en una entidad.

**CAUSA:** Un razón por lo que la operación aconteció de cierta manera.

**ENTIDAD:** Es un sustantivo y generalmente se refiere a una entidad del modelo estructural del negocio.

**OPERACIÓN:** Se refiere a una acción que se debe realizar sobre los datos de una o varias entidades. Por ejemplo: registrar, eliminar, modificar, etc.

**RESTRICCIÓN:** Se refiere a alguna restricción para un tipo de dato. Por ejemplo: máximo, mínimo, etc.

**TAMAÑO:** Es el tamaño del atributo de una entidad, el cual se encuentra definido en el modelo conceptual.

**VALOR:** Es un sustantivo concreto y generalmente se refiere a un valor en específico.

### 5.4.1. Mensajes

**Mensaje:** MSG-SLL1 Generación de llaves

**Tipo:** Notificación

**Objetivo:** Notificar al actor que la operación se ha realizado de forma exitosa.

**Redacción:** La generación exitosa de llaves del servidor.

**Ejemplo:**

**Mensaje:** MSG-SLL2 Números iguales

**Tipo:** Error

**Objetivo:** Notificar al actor que los números aleatorios elegidos se encuentran repetidos.

**Redacción:** Los números aleatorios se encuentran repetidos o no son primos.

**Ejemplo:**

**Mensaje:** MSG-SLL3 Número incorrecto

**Tipo:** Error

**Objetivo:** Notificar al actor que el número elegido no cumple con las características específicas.

**Redacción:** El número aleatorio no cumple con las especificaciones.

**Ejemplo:**

**Mensaje:** MSG-N1 Archivo no encontrado

**Tipo:** Notificación

**Objetivo:** Notificar al usuario que el archivo enviado no existe almacenado en la nube.

**Redacción:** El archivo solicitado no existe.

**Ejemplo:**

**Mensaje:** MSG-CL1 Carpeta vacía

**Tipo:** Notificación

**Objetivo:** Notificar al usuario que su carpeta personal no tiene archivos

**Redacción:** La carpeta personal se encuentra sin archivos.

**Ejemplo:**

**Mensaje:** MSG-CL2 Archivo incompatible

**Tipo:** Error

**Objetivo:** Notificar al usuario que el archivo que intenta subir no es válido

**Redacción:** El archivo no es compatible con el almacenamiento.

**Ejemplo:**

**Mensaje:** MSG-CL3 Número incorrecto

**Tipo:** Notificación

**Objetivo:** Notificar al actor que el número aleatorio no esta dentro del rango de tamaño.

**Redacción:** El número aleatorio no se encuentra dentro del rango establecido.

**Ejemplo:**

**Mensaje:** MSG-CL4 Error al generar la llave

**Tipo:** Notificación

**Objetivo:** Notificar al actor que la operación se ha realizado de forma exitosa.

**Redacción:**

**Ejemplo:**

**Mensaje:** MSG-CL5 Archivo almacenado

**Tipo:** Notificación

**Objetivo:** Notificar al actor que la operación se ha realizado de forma exitosa.

**Redacción:**

**Ejemplo:**

**Mensaje:** MSG-CL6 Archivo inexistente

**Tipo:** Notificación

**Objetivo:** Notificar al actor que la operación se ha realizado de forma exitosa.

**Redacción:**

**Ejemplo:**

**Mensaje:** MSG1 Operación exitosa

**Tipo:** Notificación

**Objetivo:** Notificar al actor que la operación se ha realizado de forma exitosa.

**Redacción:** DETERMINADO ENTIDAD ha sido OPERACIÓN exitosamente.

**Parámetros:** El mensaje se muestra con base en los siguientes parámetros:

- DETERMINADO ENTIDAD: Artículo determinado más el nombre de la entidad sobre la que se realiza la operación.
- OPERACIÓN: Es la acción que el actor solicitó realizar. Puede ser registro, eliminación, modificación o revisión.

**Ejemplo:** El Cliente ha sido registrado exitosamente.

**Mensaje:** MSG4 Registro repetido

**Tipo:** Error

**Objetivo:** Notificar al actor que la entidad que desea registrar ya existe en el sistema.

**Redacción:** DETERMINADO ENTIDAD que intentas registrar ya existe.

**Parámetros:** El mensaje se muestra con base en los siguientes parámetros:

- DETERMINADO ENTIDAD: Artículo determinado más el nombre de la entidad sobre la que se realiza la operación.

**Ejemplo:** El Cliente que intentas registrar ya existe.

**Mensaje:** MSG5 Dato incorrecto

**Tipo:** Error

**Objetivo:** Notificar al actor que el dato no tiene el tipo solicitado.

**Redacción:** DETERMINADO ENTIDAD debe ser INDETERMINADO TIPODATO.

**Parámetros:** El mensaje se muestra con base en los siguientes parámetros:

- DETERMINADO ENTIDAD: Artículo determinado más el nombre de la entidad sobre la que se realiza la operación.
- INDETERMINADO: Artículo indeterminado.
- TIPODATO: Indica el tipo de dato, por ejemplo cadena o número.

**Ejemplo:** El dato debe ser un número.

**Mensaje:** MSG6 Longitud inválida

**Tipo:** Error

**Objetivo:** Notificar al actor que el dato no tiene la longitud correcta.

**Redacción:** DETERMINADO ENTIDAD debe tener RESTRICCIÓN TAMAÑO TIPODATO.

**Parámetros:** El mensaje se muestra con base en los siguientes parámetros:

- DETERMINADO ENTIDAD: Artículo determinado más el nombre de la entidad sobre la que se realiza la operación.
- RESTRICCIÓN: Puede ser máximo, al menos, mínimo, etc.
- TAMAÑO: Tamaño del dato.
- TIPODATO: Indica el tipo de dato con el que se mide el campo.

**Ejemplo:** La contraseña debe tener mínimo 6 caracteres.

**Mensaje:** MSG9 Dato requerido

**Tipo:** Error

**Objetivo:** Notificar al actor que el dato es requerido y se ha omitido.

**Redacción:** Este dato es requerido.

**Mensaje:** MSG10 No existe información

**Tipo:** Error

**Objetivo:** Notificar al actor que aún no existe información registrada en el prototipo.

**Redacción:** DETERMINADO ENTIDAD no se encuentra en el sistema.

**Mensaje:** MSG11 Contraseña incorrecta

**Tipo:** Error

**Objetivo:** Notificar al actor que la contraseña que introdujo no fue correcta.

**Redacción:** No es correcta su contraseña.



# Apéndice A

## Lista de acrónimos

### A.1. Definiciones, acrónimos y abreviaturas

#### Acrónimos

- HP: Hewllet-Packard.
- DupLESS: Server-Aided Encryption for Deduplicated Storage (Cifrado Asistido por un Servidor para Almacenamiento Sin Duplicados).
- ABS: The Apportioned Backup System (Sistema de Respaldo Asignado).
- SIGOPS: Special Interest Group on Operating Systems (Grupo de Interés Especial sobre Sistemas Operativos).
- TahoeFS: The Least-Authority Filesystem (Sistema de Archivos de Menor Autoridad).
- AES: Advanced Encryption Standard (Estándar de Cifrado Avanzado).
- DES: Data Encryption Standard (Estándar de Cifrado de Datos).
- RSA: Rivest Shamir Adleman.
- MD: Message Digest (Resumen del Mensaje).
- SHA: Secure Hash Algorithm (Algoritmo Seguro de Hash).
- NIST: National Institute of Standards and Technology (Instituto Nacional de Estándares y Tecnología).
- SaaS: Software as a Service (Software como Servicio).
- PaaS: Platform as a Service (Plataforma como Servicio).
- IaaS: Infrastructure as a Service (Infraestructura como Servicio).

- API: Application Programming Interface (Interfaz de Programación de Aplicaciones).
- BPMN: Business Process Model and Notation (Modelo de Proceso Empresarial y Notación).

# Apéndice B

## Glosario de términos

### B.1. Glosario de Términos

**Usuario o Entidad:** Persona que utiliza el servicio de almacenamiento para guardar archivos en la nube.

**Archivo:** Conjunto de datos almacenados en la memoria de una computadora que puede manejarse con una instrucción única.

**Nube:** Espacio de almacenamiento y procesamiento de datos y archivos ubicado en internet, al que puede acceder el usuario desde cualquier dispositivo.

**Privacidad:** Capacidad que una organización o individuo tiene para determinar qué datos en un sistema informático pueden ser compartidos con terceros.

**Seguridad:** Conjunto de medidas preventivas y reactivas de las organizaciones o sistemas tecnológicos que permiten resguardar y proteger la información, buscando mantener la confidencialidad, disponibilidad e integridad de datos de la misma.

**Duplicación:** Acción y efecto de duplicar.

**Duplicar:** Repetir exactamente algo, hacer una copia de ello.

**Cifrar:** Escribir un mensaje en clave mediante un sistema de signos formado por números, letras, símbolos, etc.

**Descifrar:** Declarar lo que está escrito en cifra o en caracteres desconocidos, sirviéndose de clave dispuesta para ella, o sin clave, por conjeturas y reglas críticas.

**Conjetura:** Juicio que se forma de algo por indicios u observaciones.

**Mensaje:** Información transmitida.

**Algoritmo Criptográfico:** Es una función matemática usada en los procesos de cifrado y descifrado. Trabaja en combinación con una llave para cifrar y descifrar datos. Modifica los datos de un documento con el objeto de alcanzar algunas características de seguridad (autenticación, integridad y confidencialidad).

**Clave o Llave:** Código de signos convenidos para la transmisión de mensajes secretos o privados.

**Aritmética Modular:** Es un sistema aritmético para clases de equivalencia de números enteros llamadas clases de congruencia.

**Función Computacional:** Funciones que pueden ser calculadas por una máquina de Turing.

**Opacidad:** Cualidad de opaco.

**Opaco:** Oscuro.

# Referencias

- [1] R. Bellare, Keelveedhi. *Message-locked encryption and secure deduplication.*, volume 7881. EUROCRYPT, 2013.
- [2] R. Bellare, Keelveedhi. Dupless: Server-aided encryption for deduplicated storage., 2013:429.
- [3] T. C. y. P. A. Cooley J. Abs: the apportioned backup system. MIT Laboratory for Computer, 2004.
- [4] M. C. y. N. B. Cox L. *SIGOPS Oper. Syst.* Pastiche: making backup cheap and easy, 2002.
- [5] EFE/Cisco. Estimación del crecimiento del tráfico en la nube en el periodo 2013-2018., 2014.
- [6] L. B. Jaquelina. Fundamentos de criptografía. Universidad Nacional Autónoma de México, 2012. <http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/1-panorama-general/11-concepto-decriptografia>.
- [7] A. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.
- [8] C. Paar and J. Pelzl. *Understanding Cryptography - A Textbook for Students and Practitioners*. Springer, 2010.
- [9] G. Z. C. Patricia. Diseño y desarrollo de un sistema para elecciones electrónicas seguras (seles). Master's thesis, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, 2005.
- [10] D. J. R., A. A., B. W. J., S. D., and T. M. *Reclaiming space from duplicate files in a serverless distributed file system*. ICDCS, 2002.
- [11] S/A. Flud backup, 2011. [http://flud.org/wiki/Flud\\_Backup](http://flud.org/wiki/Flud_Backup).
- [12] s/a. Cifrado simetrico. Guía de Gnu Privacy Guard, 2015. <https://www.gnupg.org/gph/es/manual/c190.html#AEN201>.
- [13] T. O. Sergio. Introducción a la criptología. *InfoCentreUV*, 2003.

- [14] R. Sharma. Data de-duplication in cloud computing: A review. *International Journal of Engineering Applied Sciences and Technology*, 2017, 2017.
- [15] D. I. G. Sánchez. Seguridad en redes y criptografía. Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Monterrey, 2004. <https://repositorio.itesm.mx/ortec/handle/11285/571244>.
- [16] W. Stallings. *Cryptography and Network Security: Principles and Practice*. Pearson Education, 5a edition, 2002.
- [17] D. R. Stinson. *Cryptography - theory and practice*. Discrete mathematics and its applications series. CRC Press, 1995.
- [18] H. D. y. W. N. Wilcox-O'Hearn Z. *Tahoe: The least-authority*. In Proceedings of the 4th ACM, 2008.
- [19] E. A. M. y M.C. Ma. Jaquelina López Barrientos. Fundamentos de criptografía. Universidad Nacional Autónoma de México, 2012. <http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/1-panorama-general/11-concepto-de-criptografia>.
- [20] E. A. M. y M.C. Ma. Jaquelina López Barrientos. Fundamentos de seguridad informática. Universidad Nacional Autónoma de México, 2012. <http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/1-panorama-general/14-ataques/142-ataques-a-los-metodos-de-cifrado>.