



**UNIVERSIDAD JOSÉ CARLOS MARIÁTEGUI**

**VICERRECTORADO DE INVESTIGACIÓN**

**FACULTAD DE INGENIERÍA Y  
ARQUITECTURA**

**ESCUELA PROFESIONAL DE  
INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**TRABAJO DE SUFICIENCIA PROFESIONAL**

**“USO DE LA INGENIERÍA WEB PARA EL DESARROLLO DE  
UN SISTEMA DE APOYO PARA LOS ALMACENES DE  
SOUTHERN PERÚ COPPER CORPORATION”**

**PRESENTADO POR**

**BACHILLER WILMAR WERNER GARIBALDI VALDEZ**

**ASESOR**

**ING. WALTER DEMETRIO COAYLA MAMANI**

**PARA OPTAR TÍTULO PROFESIONAL DE  
INGENIERO DE SISTEMAS E INFORMÁTICA**

**MOQUEGUA – PERÚ**

**2016**

## CONTENIDO

	Pág.
Página de jurado.....	i
Dedicatoria.....	ii
Agradecimientos.....	iii
Contenido.....	iv
Índice de tablas.....	vii
Índice de figuras.....	viii
RESUMEN.....	ix
ABSTRACT.....	x

### CAPÍTULO I INTRODUCCIÓN

### CAPÍTULO II OBJETIVOS

2.1. Objetivo general.....	3
2.2. Objetivos específicos.....	3

### CAPÍTULO III DESARROLLO DEL TEMA

3.1. Marco teórico.....	4
3.1.1. Sistemas de información.....	4
3.1.2. Sistema de planificación de recursos empresariales.....	4
3.1.3. Sistema gestor de base de datos.....	5
3.1.4. SQL.....	5
3.1.5. Modelo entidad-relación.....	6
3.1.6. Lenguaje de modelamiento unificado (UML).....	6
3.1.7. Objeto.....	7

3.1.8. Clases de objetos .....	7
3.1.9. Casos de uso.....	8
3.1.10. Diagrama de secuencia.....	9
3.1.11. Diagrama de empaquetado.....	10
3.1.12. Diagrama de clases.....	10
3.1.13. Patrón modelo vista controlador .....	11
3.1.14. Cliente-servidor.....	12
3.1.15. Navegador web .....	12
3.1.16. Procesador .....	13
3.1.17. SAP .....	13
3.1.18. Sistema operativo.....	14
3.1.19. Memoria de acceso aleatorio .....	14
3.1.20. Hoja de cálculo .....	15
3.1.21. Archivo .....	15
3.1.22. HTML .....	16
3.1.23. PHP .....	16
3.1.24. Framework .....	16
3.1.25. Prototipo.....	17
3.1.26. Intranet .....	17
3.1.27. PDF .....	18
3.1.28. CodeIgniter .....	18
3.1.29. Metodología ágil .....	19
3.1.30. Ingeniería web.....	19
3.2. Caso práctico.....	20
3.2.1. Comunicación .....	22
3.2.2. Planificación .....	46
3.2.3. Modelamiento .....	48
3.2.4. Prototipo.....	57
3.2.5. Modelo de configuración .....	58
3.2.6. Construcción .....	59
3.2.7. Despliegue.....	66
3.2.8. Tecnología y herramientas .....	69

3.2.9. Pruebas .....	70
3.3. Representación de resultados .....	72

**CAPÍTULO IV**  
**CONCLUSIONES Y RECOMENDACIONES**

4.1. Conclusiones .....	75
4.2. Recomendaciones.....	76
REFERENCIAS BIBLIOGRÁFICAS.....	77

## ÍNDICE DE TABLAS

	Pág.
Tabla 1. Ambiente técnico .....	26
Tabla 2. Ambiente del proyecto .....	26
Tabla 3. Reglas de negocio .....	27
Tabla 4. Escenario de caso de uso: Iniciar sesión .....	29
Tabla 5. Escenario de caso de uso: Actualizar catálogo de materiales .....	30
Tabla 6. Escenario de caso de uso: Subir carga de reservas, parte 1 .....	31
Tabla 7. Escenario de caso de uso: Subir carga de reservas, parte 2 .....	32
Tabla 8. Escenario de caso de uso: Subir carga de reservas, parte 3 .....	33
Tabla 9. Escenario de caso de uso: Subir carga de reservas, parte 4 .....	34
Tabla 10. Escenario de caso de uso: Subir carga de reservas, parte 5 .....	35
Tabla 11. Escenario de caso de uso: Asignar personal .....	36
Tabla 12. Escenario de caso de uso: Imprimir reservas .....	37
Tabla 13. Escenario de caso de uso: Recuperar carga.....	38
Tabla 14. Escenario de caso de uso: Descargar archivo .....	39
Tabla 15. Escenario de caso de uso: Ver reserva anterior.....	40
Tabla 16. Escenario de caso de uso: Ver reporte .....	41
Tabla 17. Escenario de caso de uso: Registrar usuarios.....	42
Tabla 18. Escenario de caso de uso: Registrar usuarios.....	43
Tabla 19. Incrementos de la aplicación .....	44
Tabla 20. Tiempo de desarrollo de incrementos .....	45
Tabla 21. Riesgos en el desarrollo .....	47
Tabla 22. Requerimientos para el servidor de producción.....	58
Tabla 23. Requerimientos para el cliente .....	59

## ÍNDICE DE FIGURAS

	Pág.
Figura 1. La clase Caballo.....	8
Figura 2. Caso de uso de compra de un caballo.....	9
Figura 3. Envío de un mensaje.....	9
Figura 4. Ejemplo de diagrama de empaquetado con elementos encapsulados y visibles.....	10
Figura 5. Ejemplos de asociaciones descritas con sus cardinalidades.....	11
Figura 6. Una aplicación web entregada en incrementos.....	21
Figura 7. Casos de uso del sistema.....	28
Figura 8. Diagrama de clases - Controladores.....	49
Figura 9. Diagrama de clases - Modelos.....	50
Figura 10. Diagrama de secuencia – Iniciar sesión.....	51
Figura 11. Diagrama de secuencia – Actualizar catálogo.....	51
Figura 12. Diagrama de secuencia – Subir carga de reservas.....	52
Figura 13. Diagrama de secuencia – Asignar personal.....	52
Figura 14. Diagrama de secuencia – Imprimir reservas.....	53
Figura 15. Diagrama de secuencia – Recuperar carga.....	53
Figura 16. Diagrama de secuencia – Descargar archivo.....	53
Figura 17. Diagrama de secuencia – Ver reserva anterior.....	54
Figura 18. Diagrama de secuencia – Ver reporte.....	54
Figura 19. Diagrama de secuencia – Registrar usuarios.....	55
Figura 20. Diagrama de secuencia – Editar usuarios.....	55
Figura 21. Diagrama Entidad-Relación del sistema.....	56
Figura 22. Diagrama de Paquetes.....	57
Figura 23. Diagrama de Despliegue.....	57
Figura 24. Prototipo del sistema.....	58
Figura 25. Opciones del menú Reservas.....	62
Figura 26. Estética de la aplicación.....	64
Figura 27. Formulario Nueva Carga.....	73
Figura 28. Formulario Asignar Personal.....	73
Figura 29. Imprimir Reservas.....	74

## RESUMEN

El presente trabajo de suficiencia profesional describe el procedimiento realizado para desarrollar un sistema web de apoyo para los almacenes de la empresa Southern Perú Copper Corporation. El sistema se desarrolló con el fin de procesar e imprimir reservas de materiales obtenidas del sistema SAP. Se utilizó la ingeniería web como marco de trabajo para llevar un mejor manejo y control sobre el desarrollo del sistema. Con la aplicación del marco de trabajo se pudo finalizar el sistema a tiempo, logrando los objetivos deseados y cumpliendo con las expectativas de los usuarios. El uso de la ingeniería web redujo el riesgo de cometer errores, acortó el tiempo de entrega y mejoró la calidad del producto.

*Palabras clave:* Programación informática, sistema de información en línea, procesamiento de datos, análisis de datos, formato de datos.

## **ABSTRACT**

This professional proficiency work describes the procedure carried out to develop a web support system for the warehouses of the company Southern Peru Copper Corporation. The system was developed in order to process and print reserves of materials obtained from the SAP system. Web engineering was used as a framework to carry out better management and control over the development of the system. With the application of the framework, the system could be completed on time, achieving the desired objectives and meeting the expectations of the users. The use of web engineering reduced the risk of making mistakes, shortened delivery time and improved product quality.

*Keywords:* Computer programming, online information systems, data processing, data analysis, data formats.



## **CAPÍTULO I**

### **INTRODUCCIÓN**

La empresa Southern Perú Copper Corporation utilizaba un sistema de planificación de recursos empresariales (ERP) llamado Ellipse para manejar e integrar la información de sus operaciones. En este sistema los usuarios pueden solicitar materiales a los almacenes para la producción de cobre refinado o para realizar labores de mantenimiento.

Debido a la antigüedad del sistema y la falta de soporte por parte del fabricante, es que la empresa decide migrar a otro sistema, llamado SAP.

La migración al nuevo sistema trajo numerosos problemas a diferentes áreas de la empresa. Para los almacenes, el sistema SAP no cuenta con un formulario adecuado que muestre la información necesaria para atender las reservas de materiales que generan los usuarios en toda la empresa.

Si los almacenes no despachan los materiales que solicitan los usuarios, la producción de cobre paralizaría y el mantenimiento de las plantas y equipos no sería llevado a cabo, por lo que es de suma importancia la atención de las reservas.

Debido a la gran cantidad de solicitudes que el personal de la empresa

genera diariamente, no es posible completar la información faltante de forma manual ni realizar las operaciones necesarias para proceder con el despacho de las reservas.

Es por ello que se decidió desarrollar una aplicación que utilice la información obtenida de los formularios de SAP para completar la información faltante del formulario de despacho y así poder atender las reservas de materiales, para luego registrar el despacho en el sistema principal de la empresa.

La aplicación se desarrolla como aplicación web debido a los beneficios que ofrece sobre una aplicación de escritorio, los cuales son facilidad de mantenimiento, facilidad para distribuir las actualizaciones y compatibilidad con diferentes sistemas operativos y hardware.

## **CAPÍTULO II**

### **OBJETIVOS**

#### **2.1. Objetivo general**

Desarrollar un sistema web para brindar apoyo a las labores de despacho de materiales en los almacenes de la empresa Southern Peru Copper Corporation utilizando la ingeniería web como marco de trabajo.

#### **2.2. Objetivos específicos**

- a) Desarrollar un sistema web que permita procesar e imprimir las reservas de materiales obtenidas del sistema SAP.
- b) Utilizar la ingeniería web como marco de trabajo para llevar un mejor manejo y control sobre el desarrollo del sistema.

## **CAPÍTULO III**

### **DESARROLLO DEL TEMA**

#### **3.1. Marco teórico**

##### **3.1.1. Sistemas de información**

J. Laudon y K. Laudon (2012) afirman: “Podemos plantear la definición técnica de un sistema de información como un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización” (p. 15).

##### **3.1.2. Sistema de planificación de recursos empresariales**

Lara (2011) afirma:

Los sistemas de planificación de recursos empresariales (en inglés ERP, Enterprise Resource Planning) son sistemas de gestión de información que integran y automatizan muchas de las prácticas de negocio asociadas con los aspectos operativos o productivos de una empresa. Se caracterizan por estar compuestos por diferentes partes integradas en una única aplicación.

Estas partes son de diferente uso, por ejemplo: producción, ventas, compras, logística, contabilidad (de varios tipos), gestión de proyectos, GIS (sistema de información geográfica), inventarios y control de almacenes, pedidos, nóminas, etc. Sólo podemos definir un ERP como la integración de todas estas partes. El ERP integra todo lo necesario para el funcionamiento de los procesos de negocio de la empresa.

### **3.1.3. Sistema gestor de base de datos**

Nevado (2010) afirma:

Un sistema gestor de bases de datos o SGBD (aunque se suele utilizar más a menudo las siglas DBMS procedentes del inglés, Data Base Management System) es el software que permite a los usuarios procesar, describir, administrar y recuperar los datos almacenados en una base de datos. En estos Sistemas se proporciona un conjunto coordinado de programas, procedimientos y lenguajes que permiten a los distintos usuarios realizar sus tareas habituales con los datos, garantizando además la seguridad de los mismos. (p. 25).

### **3.1.4. SQL**

Aliaga, Aramburu, Marqués y Quintana (2008) afirman:

SQL (Structured Query Language) es un lenguaje de programación diseñado específicamente para el acceso a Sistemas de Gestión de Bases de Datos Relacionales (SGBDR). El lenguaje SQL es un lenguaje de cuarta generación. Es decir, en este lenguaje se indica qué información se desea obtener o procesar, pero

no cómo se debe hacer. Es labor interna del sistema elegir la forma más eficiente de llevar a cabo la operación ordenada por el usuario (p. 8).

### **3.1.5. Modelo entidad-relación**

Para Korth, Silberschatz y Sudarshan (2011):

El modelo de datos entidad-relación (E-R) fue desarrollado para facilitar el diseño de base de datos permitiendo especificar el esquema empresarial que representa la estructura lógica general de una base de datos.

El modelo E-R emplea tres conceptos básicos: conjunto de entidades, conjunto de relaciones y atributos (p. 262).

### **3.1.6. Lenguaje de modelamiento unificado (UML)**

Rumbaugh, Jacoson y Booch (2005):

El Lenguaje unificado de Modelado (UML) es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software.

Captura decisiones y conocimientos sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema.

Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo (p. 3).

### **3.1.7. Objeto**

Según Debrauwer, & Van Der Heyde (2013):

Un objeto es una entidad identificable del mundo real. Puede tener una existencia física (un caballo, un libro) o no tenerla (un texto de ley). Identificable significa que el objeto se puede designar.

En UML todo objeto posee un conjunto de atributos (estructura) y un conjunto de métodos (comportamiento). Un atributo es una variable destinada a recibir un valor. Un método es un conjunto de instrucciones que toman unos valores de entrada y modifican los valores de los atributos o producen un resultado.

Todo sistema concebido en UML está compuesto por objetos que interactúan entre sí y realizan operaciones propias de su comportamiento (p. 13).

### **3.1.8. Clases de objetos**

Debrauwer, et al. (2013) afirman:

Un conjunto de objetos similares, es decir, con la misma estructura y comportamiento, y constituidos por los mismos atributos y métodos, forma una clase de objetos. La estructura y el comportamiento pueden entonces definirse en común en el ámbito de la clase.

Todos los objetos de una clase, llamada también instancia de clase, se distinguen por tener una identidad propia y sus atributos les confieren valores específicos (p. 15).

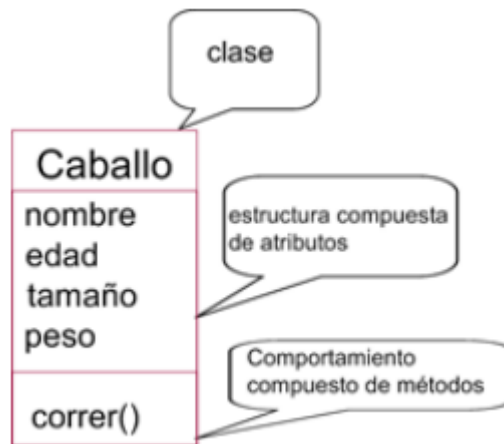


Figura 1. La clase Caballo

Fuente: Debrauwer, L., & Van Der Heyde, F., 2013

### 3.1.9. Casos de uso

Debrauwer, et al. (2013) afirman:

Los casos de uso describen en forma de acciones y reacciones el comportamiento del sistema, estudiado desde el punto de vista del usuario. Definen los límites del sistema y sus relaciones con el entorno. Los casos de uso explicitan los requisitos funcionales del sistema relativos a uno de los objetivos del usuario. Éstos se denominan también, de manera más precisa, casos de uso con objetivo usuario (p. 26).





Figura 2. Caso de uso de compra de un caballo

Fuente: Debrauwer, L., & Van Der Heyde, F., 2013

### 3.1.10. Diagrama de secuencia

Debrauwer, et al. (2013) afirman:

El diagrama de secuencia describe las interacciones entre un grupo de objetos mostrando de forma secuencial los envíos de mensajes entre objetos. El diagrama puede asimismo mostrar los flujos de datos intercambiados durante el envío de mensajes.

Para interactuar entre sí, los objetos se envían mensajes. Durante la recepción de un mensaje, los objetos se vuelven activos y ejecutan el método del mismo nombre. Un envío de mensaje es, por tanto, una llamada a un método.

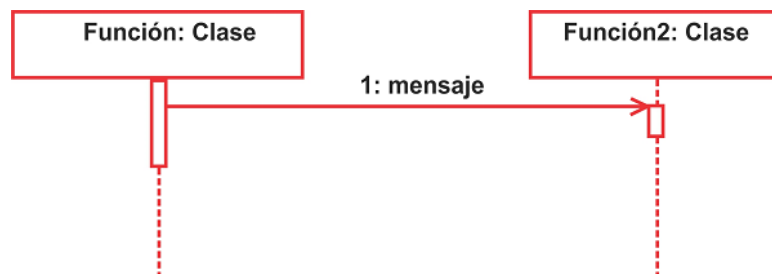


Figura 3. Envío de un mensaje

Fuente: Debrauwer, L., & Van Der Heyde, F., 2013

### 3.1.11. Diagrama de empaquetado

Debrauwer, et al. (2013) afirman:

Un empaquetado es una agrupación de elementos de modelado: clases, componentes, casos de uso, otros empaquetados, etc.

Los empaquetados de UML resultan útiles durante el modelado de sistemas importantes para reagrupar los diferentes elementos. La agrupación estructura de ese modo el modelado (p. 102).

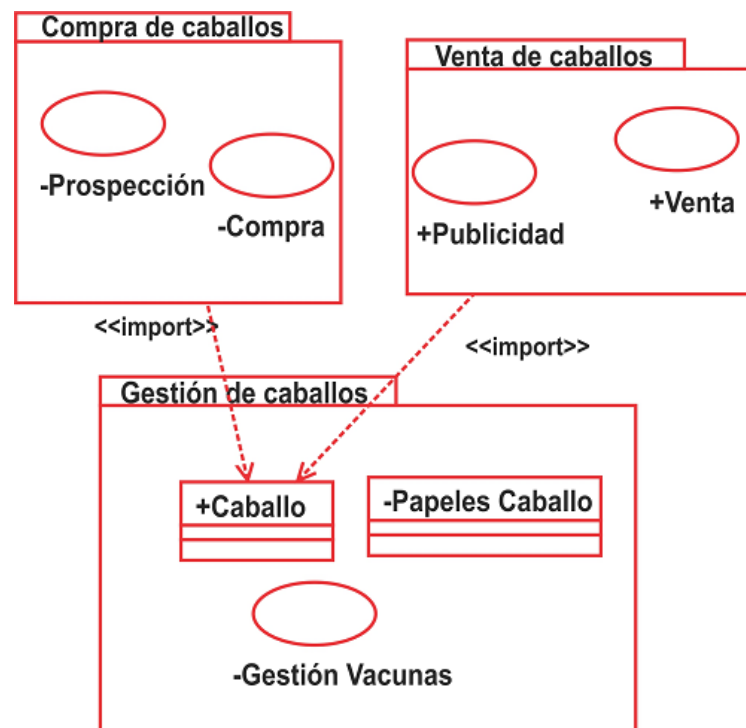


Figura 4. Ejemplo de diagrama de empaquetado con elementos encapsulados y visibles

Fuente: Debrauwer, L., & Van Der Heyde, F., 2013

### 3.1.12. Diagrama de clases

Debrauwer, et al. (2013) afirman:

En el mundo real, muchos objetos están vinculados entre sí. Dichos vínculos corresponden a una asociación existente entre los objetos. En UML, estos vínculos se describen mediante asociaciones, de igual modo que los objetos se describen mediante clases. Un vínculo es un elemento de una asociación. Por consiguiente, una asociación vincula a las clases. Los elementos de la asociación vinculan entre sí las instancias de las clases.

Las asociaciones tienen un nombre y, como ocurre con las clases, éste es un reflejo de los elementos de la asociación.

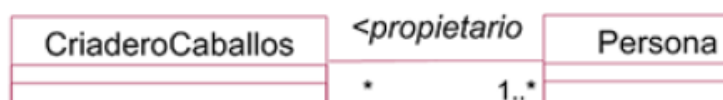


Figura 5. Ejemplos de asociaciones descritas con sus cardinalidades

Fuente: Debrauwer, L., & Van Der Heyde, F., 2013

### 3.1.13. Patrón modelo vista controlador

Según Urraco (2013):

Modelo: es la capa de acceso a datos, en ella se realizan las conexiones y consultas a la base de datos, y los datos que esta obtiene son recibidos por la capa Controlador.

Controlador: aquí encontramos la lógica del programa, esta capa manipula la información que recupera de la base de datos, y la envía a la Vista. También recibe las acciones del usuario capturadas por la Vista y responde a ellas.

Vista: en esta parte se define la interfaz de usuario, especifica cómo se mostrará la información obtenida del Controlador y captura los eventos del usuario.

#### **3.1.14. Cliente-servidor**

Según Alegsa (2016):

El modelo cliente-servidor (client-server), describe el proceso de interacción entre la computadora local (el cliente) y la remota (el servidor).

El cliente le hace peticiones (requests, solicitudes, requerimientos) al servidor, el cual procesa dicho requerimiento y retorna los resultados al cliente apropiado.

#### **3.1.15. Navegador web**

Según Informática Hoy (s. f.):

El navegador web o navegador de internet es el instrumento que permite a los usuarios de internet navegar o surfear entre las distintas páginas de sus sitios webs preferidos. Se trata de un software que posee una interfaz gráfica compuesta básicamente de: botones de navegación, una barra de dirección, una barra de estado (generalmente, en la parte inferior de la ventana) y la mayor parte, en el centro, que sirve para mostrar las páginas web a las que se accede.

### **3.1.16. Procesador**

Según Sánchez (2016): “El procesador, también conocido como CPU o micro, es el cerebro del PC. Sus funciones principales incluyen el manejo del sistema operativo, la ejecución de las aplicaciones y la coordinación de los diferentes dispositivos que componen el equipo.

### **3.1.17. SAP**

Según Chamorro (2016):

SAP es un programa informático de gestión empresarial que nació en Alemania en los años 70's y cuyas potencialidades y aplicaciones llegan hasta nuestros días. En la versión inglesa significa "Systems, Applications, Products in Data Processing" lo que, traducido al español, sería: “Sistemas, Aplicaciones y Productos para el procesamiento de datos”.

Como software no sólo se queda en su vertiente informática, sino que es tan potente que podríamos decir que es un sistema, una tecnología para gestionar los recursos de la empresa al más alto nivel. Su utilidad radica en que permite recopilar todo tipo de datos de la empresa y procesarlos para proporcionar a las diferentes áreas de la organización, información útil para tomar decisiones. Su sistema modular facilita que pueda llegar a actuar en un departamento muy concreto de la empresa o bien a nivel global y estratégico si se le suman más módulos al software.

Así, si tuviéramos que definir qué es SAP diríamos que es un software ERP (Enterprise Resource Planning), que permite planificar y gestionar los

recursos de todas las áreas de la empresa: desde logística a contabilidad, pasando por el departamento comercial y de marketing, finanzas, producción, gestión de proyectos, de la calidad, mantenimiento o dirección y administración general.

### **3.1.18. Sistema operativo**

Según Vasquez (2011):

Un Sistema Operativo (SO) es el software básico de una computadora que provee una interfaz entre el resto de programas del computador, los dispositivos hardware y el usuario. Las funciones básicas del Sistema Operativo son administrar los recursos de la máquina, coordinar el hardware y organizar archivos y directorios en dispositivos de almacenamiento.

### **3.1.19. Memoria de acceso aleatorio**

Según Wikipedia (2016):

La memoria de acceso aleatorio (Random Access Memory, RAM) se utiliza como memoria de trabajo de computadoras para el sistema operativo, los programas y la mayor parte del software. En la RAM se cargan todas las instrucciones que ejecuta la unidad central de procesamiento (procesador) y otras unidades del computador.

Se denominan «de acceso aleatorio» porque se puede leer o escribir en una posición de memoria con un tiempo de espera igual para cualquier posición, no siendo necesario seguir un orden para acceder (acceso secuencial) a la información de la manera más rápida posible.

### **3.1.20. Hoja de cálculo**

Según Muriel (s. f.):

Una hoja de cálculo es una retícula de casillas dispuestas en filas y columnas, las cuales contienen números y cálculos realizados a partir de ellos. Por ello la hoja de cálculo es un instrumento ideal para la realización organizada de cálculos en los que intervengan datos agrupados en diversas categorías.

Aunque el concepto expuesto es antiguo, la aparición de los ordenadores como máquinas que procesan automáticamente la información y con gran rapidez y precisión, ha revolucionado la elaboración de hojas de cálculo.

En concreto, existen aplicaciones informáticas de uso muy general que permiten la creación y gestión de las llamadas hojas de cálculo electrónico. La característica fundamental de este tipo de hojas electrónicas es que trabajan con fórmulas que se definen en función de las celdas (intersección de una fila y una columna de la hoja) y no con los datos contenidos en ellas, de forma que si variamos algún número de alguna celda inmediatamente se realiza el recálculo automático con los nuevos datos de las celdas, apareciendo el nuevo resultado en la celda donde se definió la fórmula.

### **3.1.21. Archivo**

Según Lanzillotta (s. f.):

Un archivo, también denominado Fichero, es una unidad de datos o información almacenada en algún medio que puede ser utilizada por aplicaciones de la computadora. Cada archivo se diferencia del resto debido a que tiene un nombre

propio y una extensión que lo identifica. Esta extensión sería como el apellido y es lo que permite diferenciar el formato del archivo y, asimismo, interpretar los caracteres que conforman el contenido del archivo.

### **3.1.22. HTML**

Según Castro (2015):

HTML es el acrónimo en inglés de HyperText Markup Language (en español se traduce como lenguaje de marcado de hipertexto). Es un lenguaje abstracto que aplicaciones pueden usar para representar documentos (se les llama documentos a instancias completas, como lo son las páginas web) y que puede ser transmitido fácilmente por algún medio, como lo es Internet. Los navegadores de Internet procesan e interpretan documentos descritos en HTML usando un analizador de HTML.

### **3.1.23. PHP**

Fumàs afirma (2014):

PHP (Hypertext Preprocessor) es un lenguaje de programación interpretado que se utiliza para la generación de páginas web de forma dinámica. Este código se ejecuta al lado del servidor y se incrusta dentro del código HTML. Cabe destacar que es un lenguaje de código abierto, gratuito y multiplataforma.

### **3.1.24. Framework**

Según Naula (2010):



Con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Es decir, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Entre los objetivos principales de un framework tenemos: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Entonces podemos definir a un framework Web, como un conjunto de componentes (por ejemplo, clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas web.

### **3.1.25. Prototipo**

Según Wikipedia (2016):

Un prototipo en software es un modelo del comportamiento del sistema que puede ser usado para entenderlo completamente o ciertos aspectos de él y así clarificar los requerimientos. Un prototipo es una representación de un sistema, aunque no es un sistema completo, posee las características del sistema final o parte de ellas.

### **3.1.26. Intranet**

Según Abad (2010):

Una intranet es una red propia de una organización, diseñada y desarrollada siguiendo los protocolos propios de Internet, en particular el protocolo TCP/IP.

Puede tratarse de una red aislada, es decir no conectada a Internet. Para entendernos, una intranet configurada para ofrecer servicios WWW permite a una red de ordenadores conectados entre sí, en el que se incluya uno de ellos como servidor, acceder al mismo tipo de contenidos que podemos encontrar por internet pero sin necesidad de conexión externa nuestra propia red.

### **3.1.27. PDF**

Según Adobe (2016):

El formato de documento portátil (PDF) se utiliza para presentar e intercambiar documentos de forma fiable, independiente del software, el hardware o el sistema operativo. Inventado por Adobe, PDF es ahora un estándar abierto y oficial reconocido por la Organización Internacional para la Estandarización (ISO). Los archivos PDF pueden contener vínculos y botones, campos de formulario, audio, vídeo y lógica empresarial.

### **3.1.28. CodeIgniter**

Según Vergara (2016):

CodeIgniter es un framework para el desarrollo de aplicaciones en PHP, que utiliza el MVC. Esto permite a los programadores o desarrolladores Web mejorar su forma de trabajar, además de dar una mayor velocidad a la hora de crear páginas Webs. Tiene las siguientes ventajas:

- Las páginas se procesan más rápido, el núcleo de CodeIgniter es bastante ligero.

- Es sencillo de instalar, basta con subir los archivos al ftp y tocar un archivo de configuración para definir el acceso a la base de datos.
- Existe abundante documentación en la red.
- Facilidad de edición del código ya creado.
- Facilidad para crear nuevos módulos, páginas o funcionalidades.
- Estandarización del código
- Separación de la lógica y arquitectura de la web, el MVC.
- Cualquier servidor que soporte PHP+MySQL sirve para CodeIgniter.
- CodeIgniter se encuentra bajo una licencia open source, es código libre.

### **3.1.29. Metodología ágil**

K. Kendall y J. Kendall (2011) afirman:

La metodología ágil es una metodología de desarrollo de software que se basa en valores, principios y prácticas básicas.

Los cuatro valores son comunicación, simpleza, retroalimentación y valentía. Recomendamos que los analistas de sistemas adopten estos valores en todos los proyectos que emprendan y no sólo cuando adopten la metodología ágil.

Dos palabras que caracterizan a un proyecto realizado mediante una metodología ágil son interactivo e incremental (p. 14-15).

### **3.1.30. Ingeniería web**

Según Pressman y Lowe (2009):

La Ingeniería web propone un ágil y disciplinado marco de trabajo para construir aplicaciones web de calidad industrial. Las siguientes actividades son parte de un marco de trabajo genérico y son aplicables a la vasta mayoría de proyectos de aplicaciones web:

- a) **Comunicación:** Implica bastante interacción y colaboración con el cliente y abarca la recolección de requerimientos y otras actividades relacionadas.
- b) **Planificación:** Establece un plan incremental para el trabajo de Ingeniería Web. Describe las acciones que ocurrirán, las tareas técnicas a ser conducidas, los riesgos que pueden ocurrir, los recursos que se requerirán, los productos a producir y un horario de trabajo.
- c) **Modelamiento:** Abarca la creación de modelos que ayudan al desarrollador y el cliente para entender mejor los requisitos de aplicación web y el diseño que va a alcanzar dichos requisitos.
- d) **Construcción.** Combina tanto la generación de HTML, XML, Java, y código similar con la prueba que se requiere para descubrir errores en el código.
- e) **Despliegue:** Entrega un incremento de la aplicación web al cliente quien lo evalúa y provee retroalimentación basada en la evaluación (p. 12-14).

### **3.2. Caso práctico**

Para la gestión del proyecto se utilizó la ingeniería web. Se decidió esto porque las propuestas, actividades y metodologías del libro “Web Engineering: A Practitioner's Approach” (1<sup>ra</sup> Edición, año 2009) cubren adecuadamente las fases de desarrollo del sistema desde el inicio hasta el final. La siguiente figura

presenta la propuesta de dividir una aplicación en incrementos y las actividades que se realizan por cada incremento.

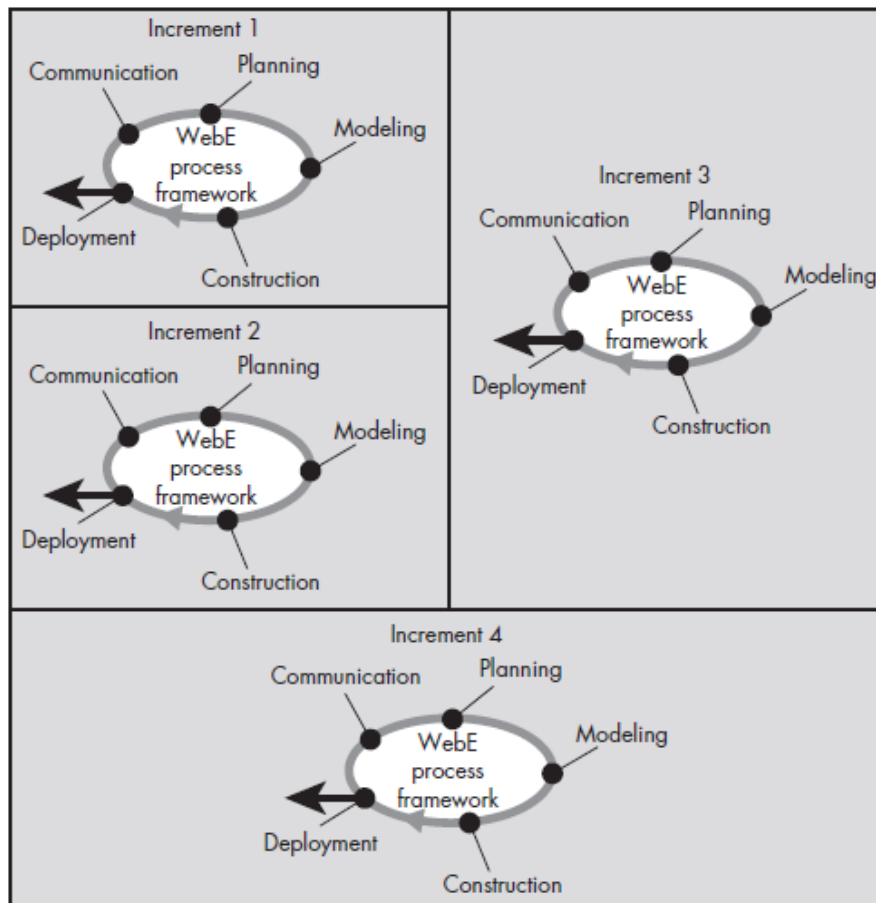


Figura 6. Una aplicación web entregada en incrementos

Fuente: Lowe, D., & Pressman, R., 2009

Es conveniente dividir la aplicación debido a que se pueden manejar mejor los cambios y nuevos requerimientos que los usuarios soliciten. Además, cada incremento terminado puede ser entregado al usuario para que lo evalúe y brinde las observaciones respectivas. Aplicar cambios y correcciones a un incremento es más rápido y económico que hacerlo en una aplicación completa.

Frente a la problemática de no poder despachar materiales por no tener disponible la información ni el apoyo necesario, se propuso la implementación de un sistema de información web para complementar las labores de despacho en los almacenes. Este proyecto se constituyó como un complemento al sistema ERP que posee la empresa, ofreciendo las funcionalidades claves para poder atender las reservas de materiales solicitadas por los usuarios.

La solución estará facultada para completar la información faltante en las reservas generadas en SAP, además de permitir asignar responsables de despacho, filtrar reservas que no se podrán imprimir y agrupar por usuario y punto de entrega.

El uso del sistema estará restringido sólo para el personal que cuente con un usuario y contraseña.

El proyecto reside en uno de los servidores de la empresa y podrá ser accedido desde cualquier computadora que se encuentre conectada a la intranet.

Se detalla a continuación las actividades que se realizaron en los incrementos del sistema.

### **3.2.1. Comunicación**

En esta actividad se obtienen los requerimientos de la aplicación. Se realizan 3 acciones.

### **3.2.1.1. Formulación**

Se enfoca en definir las necesidades y alcance del proyecto. Se describen los objetivos de la aplicación web y definen sus características. Se deben identificar a las partes interesadas (todo aquel que se beneficiará de la aplicación) para entrevistar y obtener los requerimientos de la aplicación. Se definió lo siguiente:

#### **a) *La necesidad de la aplicación***

La aplicación permitirá a los usuarios cargar, asignar e imprimir reservas de materiales para proceder con su atención.

#### **b) *Objetivos que la aplicación debe cumplir***

La aplicación permitirá asignar reservas al personal de despacho, agrupará las reservas por usuario, punto de entrega y locación del material, filtrará las reservas que no se puedan atender, dará especial atención a las reservas generadas por el área de Mantenimiento, generará reportes de las reservas procesadas y procederá a imprimirlas para su atención.

#### **c) *Usuarios***

Los usuarios que utilizarán la aplicación serán los supervisores de despacho de los almacenes de Southern Peru Copper Corporation.

#### **d) *Metas aplicativas***

- Permitir que sólo personal autorizado pueda utilizar la aplicación.

- Permitir la carga de reservas a través de archivos de hoja de cálculo.
- Descartar las reservas que el usuario no podrá atender por no haber sido solicitadas a su almacén.
- Permitir recuperar una carga en caso de no haberse podido imprimir. Permitir que se pueda descargar el archivo subido.
- Agrupar las reservas por usuario solicitante, punto de entrega y locación de materiales.
- No imprimir reservas que no se podrán atender por diferentes razones relacionadas con los materiales solicitados.
- Permitir asignar responsable de despacho a las reservas.
- No imprimir las reservas que se imprimieron anteriormente a la hora de realizar una nueva carga de reservas.
- Para las reservas de mantenimiento, no imprimir las reservas si no se cuenta con el 100% del material solicitado para todas las posiciones de la reserva agrupadas por punto de entrega y fecha de necesidad.
- Permitir ver y reimprimir las reservas almacenadas en la base de datos del sistema.
- Permitir la actualización de la información de los materiales mediante archivos de hoja de cálculo.
- Generar un reporte de las reservas procesadas por cada carga.

### ***3.2.1.2. Elicitación***

La meta de la elicitación es obtener los requerimientos para la aplicación, involucrando a los interesados.



**a) *Objetivo general del uso de la aplicación***

El objetivo general será cargar reservas, asignar las reservas al personal de despacho y finalmente imprimir las reservas.

**b) *Nivel de sofisticación***

La aplicación no requerirá que el usuario sea experto en computación. Sólo deberá seguir con el flujo de la aplicación para poder imprimir las reservas.

**c) *Acceso a la aplicación***

Los usuarios podrán acceder al sistema por medio de la intranet de la empresa, a través de un navegador web.

**d) *Categorías de usuarios***

- Usuario: Puede cargar e imprimir reservas. Puede ver los reportes.
- Administrador: Realiza las funciones de usuario. Además, puede registrar otros usuarios y configurar los filtros de reservas.

**e) *Restricciones de la aplicación***

- Restricciones internas: Detallan el ambiente técnico en donde la aplicación residirá y el ambiente del proyecto en donde la aplicación será desarrollada.
- Restricciones externas: Hace referencia a las reglas de negocio de la empresa.

**Tabla 1***Ambiente técnico*

<b>Descripción</b>	<b>Valor</b>
Navegador	Cualquier navegador web de preferencia en su última versión
Sistema operativo	Cualquier sistema operativo que pueda ejecutar un navegador web
Software	Lector de archivos PDF
Resolución de pantalla	800x600 píxeles o superior

Fuente: Elaboración propia

**Tabla 2***Ambiente del proyecto*

<b>Descripción</b>	<b>Valor</b>
Software	Microsoft SQL Server 2012
Sistema operativo	Microsoft Windows Server 2012
Servidor web	Apache HTTP Server
Lenguaje de programación	PHP

Fuente: Elaboración propia

**Tabla 3**

*Reglas de negocio*

<b>Descripción</b>	<b>Valor</b>
Procesamiento de reservas	No debe imprimir reservas solicitadas por áreas de mantenimiento si no se cuenta con el 100% del material solicitado para las posiciones agrupadas por punto de entrega y fecha de necesidad.
Acceso a bases de datos	Sólo debe acceder a la base de datos propia, no al ERP de la empresa por motivos de seguridad.
Acceso al sistema	Debe estar disponible sólo a través de la intranet.
Rendimiento	El sistema debe procesar las reservas lo más rápido posible para proceder al despacho de materiales de manera oportuna.

Fuente: Elaboración propia

**f) Casos de uso**

Los casos de uso ayudan a entender cómo los usuarios perciben la interacción con la aplicación y ayudan a separarla en incrementos.



Figura 7. Casos de uso del sistema

Fuente: Elaboración propia

**g) Escenarios de casos de uso**

**Tabla 4**

*Escenario de caso de uso: Iniciar sesión*

---

<b>Iniciar sesión</b>	
ID:	CU01
Actores:	Usuario
Descripción:	Permite el acceso al sistema del usuario al sistema
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario accede al sistema por medio de un navegador web.</li><li>2. El sistema le solicita los credenciales al usuario.</li><li>3. El usuario ingresa sus credenciales.<ol style="list-style-type: none"><li>3.1. Si la información es incorrecta, el sistema le vuelve a solicitar los credenciales al usuario.</li></ol></li><li>4. El sistema permite el acceso al usuario.</li></ol>
Precondiciones:	El usuario cuenta con credenciales válidas.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 5**

*Escenario de caso de uso: Actualizar catálogo de materiales*

---

<b>Actualizar catálogo de materiales</b>	
ID:	CU02
Actores:	Usuario
Descripción:	Permite actualizar la información de los materiales que el almacén ofrece.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario obtiene el archivo con información de los materiales de su almacén del sistema SAP.</li><li>2. El usuario ingresa al sistema y sube su archivo.<ol style="list-style-type: none"><li>2.1. Si el archivo no se puede leer o no tiene el formato correcto, muestra un mensaje de error.</li></ol></li><li>3. El sistema actualiza la información del catálogo.<ol style="list-style-type: none"><li>3.1. Si el sistema detecta un nuevo material, lo registra en la base de datos.</li></ol></li><li>4. Al finalizar el proceso de actualización, el sistema redirige al usuario al formulario de carga de reservas.</li></ol>
Precondiciones:	El archivo debe tener el formato correcto.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 6**

*Escenario de caso de uso: Subir carga de reservas, parte 1*

---

<b>Subir carga de reservas</b>	
ID:	CU03
Actores:	Usuario
Descripción:	Permite subir reservas para imprimir y ser despachadas.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario obtiene el archivo con las reservas a imprimir del sistema SAP.</li><li>2. El usuario ingresa al sistema y sube el archivo obtenido.<ol style="list-style-type: none"><li>2.1. Si el sistema detecta que el archivo no se puede leer o no tiene el formato correcto, muestra un mensaje de advertencia al usuario.</li></ol></li><li>3. El sistema almacena el archivo subido, registra el nombre del archivo, la fecha de subida y el usuario en la base de datos.</li><li>4. El sistema descarta reservas que no están dirigidas al almacén del usuario.</li></ol>
Precondiciones:	El archivo debe tener el formato correcto.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 7**

*Escenario de caso de uso: Subir carga de reservas, parte 2*

---

<b>Subir carga de reservas</b>	
ID:	CU03
Actores:	Usuario
Descripción:	Permite subir reservas para imprimir y ser despachadas.
Pasos a realizar:	<ol style="list-style-type: none"><li>5. El sistema realiza un filtro de reservas imprimidas anteriormente.<ol style="list-style-type: none"><li>5.1. Si el usuario especifica que desea reimprimir las reservas de la carga, el sistema no realiza este filtro.</li></ol></li><li>6. El sistema descarta las reservas con fecha de necesidad mayor a una semana desde el día de la carga.<ol style="list-style-type: none"><li>6.1. Si el usuario especifica ignorar el filtro, el sistema no filtra las reservas por fecha de necesidad.</li></ol></li><li>7. El sistema busca si cada reserva existe en la BD.<ol style="list-style-type: none"><li>7.1. Si existe, el sistema verifica si la reserva del archivo es diferente al de la base de datos.</li></ol></li></ol>
Precondiciones:	Debe haber reservas para imprimir.
Prioridad:	Alta

---

Fuente: Elaboración propia



**Tabla 8**

*Escenario de caso de uso: Subir carga de reservas, parte 3*

---

<b>Asignar personal</b>	
ID:	CU03
Actores:	Usuario
Descripción:	Permite subir reservas para imprimir y ser despachadas.
Pasos a realizar:	<p style="text-align: center;">7.1.1. Si la reserva es diferente, el sistema actualiza la reserva de la base de datos.</p> <p style="text-align: center;">7.2. Si no existe, el sistema registra la reserva.</p> <p>8. El sistema filtra reservas que solicitan materiales sin stock, sin stock disponible, con bloqueo de inventario o materiales no registrados en la base de datos.</p> <p>9. El sistema agrupa las reservas de mantenimiento por número de reserva, punto de entrega, fecha de necesidad y verifica las posiciones.</p>
Precondiciones:	Debe haber reservas para imprimir.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 9**

*Escenario de caso de uso: Subir carga de reservas, parte 4*

---

<b>Asignar personal</b>	
ID:	CU03
Actores:	Usuario
Descripción:	Permite subir reservas para imprimir y ser despachadas.
Pasos a realizar:	<p>9.1. Si una de las posiciones no se podrá atender en su totalidad (es decir, el 100% de la cantidad que el usuario requiere), el sistema devolverá el stock consumido por toda la reserva agrupada, volverá a analizar las reservas que soliciten los materiales que tenía esa reserva y marcará la reserva para no volver a ser analizada.</p> <p>9.2. Si las posiciones de la reserva se podrán atender en su totalidad, la reserva es marcada para imprimir.</p> <p>10. El sistema marca para imprimir las posiciones de las reservas que se podrán atender.</p>
Precondiciones:	Debe haber reservas para imprimir.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 10**

*Escenario de caso de uso: Subir carga de reservas, parte 5*

---

<b>Asignar personal</b>	
<b>ID:</b>	CU03
<b>Actores:</b>	Usuario
<b>Descripción:</b>	Permite subir reservas para imprimir y ser despachadas.
<b>Pasos a realizar:</b>	<p>11. El sistema registra como reporte todas las reservas que se imprimirán y el motivo de las otras que no se podrán.</p> <p>12. El sistema agrupa las reservas marcadas para imprimir por usuario, punto de entrega y separa las posiciones por locación de material en almacén.</p> <p>12.1. Si hay reservas para imprimir, el sistema redirigirá al usuario al formulario de asignación de responsables de atención.</p> <p>Si no hay reservas para imprimir, el sistema mostrará un mensaje de error y le dará un enlace al usuario para que vea el reporte.</p>
<b>Precondiciones:</b>	Debe haber reservas para imprimir.
<b>Prioridad:</b>	Alta

---

Fuente: Elaboración propia

**Tabla 11**

*Escenario de caso de uso: Asignar personal*

---

<b>Asignar personal</b>	
ID:	CU04
Actores:	Usuario
Descripción:	Permite asignar responsables para el despacho de reservas.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El sistema muestra al usuario las reservas que se imprimirán, agrupadas por zona de entrega y locación de materiales, además de la lista del personal de despacho disponible según el área en donde trabaja el usuario.</li><li>2. El usuario asigna el personal de despacho.</li><li>3. El sistema registra los datos y redirige al usuario a la pantalla de impresión de reservas.</li></ol>
Precondiciones:	Debe haber reservas para imprimir.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 12**

*Escenario de caso de uso: Imprimir reservas*

---

<b>Imprimir reservas</b>	
ID:	CU05
Actores:	Usuario
Descripción:	Permite asignar responsables para el despacho de reservas.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El sistema muestra las reservas agrupadas en formato PDF.</li><li>2. El usuario utiliza la función del navegador web para imprimir las reservas.</li></ol>
Precondiciones:	Debe haber reservas para imprimir.
Prioridad:	Alta

---

Fuente: Elaboración propia

**Tabla 13**

*Escenario de caso de uso: Recuperar carga*

---

<b>Recuperar carga</b>	
ID:	CU06
Actores:	Usuario
Descripción:	Permite recuperar una carga procesada anteriormente.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario visualiza todas las cargas que ha subido al sistema.</li><li>2. El usuario selecciona una carga a recuperar.</li><li>3. El sistema lo redirige al formulario de asignación de personal.</li></ol>
Precondiciones:	El usuario debe haber subido una carga anteriormente.
Prioridad:	Media

---

Fuente: Elaboración propia

**Tabla 14**

*Escenario de caso de uso: Descargar archivo*

---

<b>Descargar archivo</b>	
ID:	CU07
Actores:	Usuario
Descripción:	Permite descargar el archivo subido.
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario visualiza todas las cargas que ha subido al sistema.</li><li>2. El usuario selecciona el archivo que desea descargar.</li><li>3. El sistema entrega el archivo seleccionado.</li></ol>
Precondiciones:	El usuario debe haber subido una carga anteriormente.
Prioridad:	Baja

---

Fuente: Elaboración propia

**Tabla 15**

*Escenario de caso de uso: Ver reserva anterior*

---

<b>Ver reserva anterior</b>	
ID:	CU08
Actores:	Usuario
Descripción:	Permite ver e imprimir reservas registradas en el sistema
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario ingresa el número de reserva al sistema.</li><li>2. El sistema muestra los detalles de la reserva.</li><li>3. El usuario selecciona las posiciones a imprimir y procede a imprimir la reserva.</li></ol>
Precondiciones:	La reserva ingresada debe existir en el sistema.
Prioridad:	Baja

---

Fuente: Elaboración propia



**Tabla 16**

*Escenario de caso de uso: Ver reporte*

---

<b>Ver reporte</b>	
ID:	CU09
Actores:	Usuario
Descripción:	Permite ver el reporte de las reservas procesadas
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario visualiza todas las cargas que ha subido al sistema.</li><li>2. El usuario selecciona la carga de la cual desea ver un reporte.</li><li>3. El sistema muestra el reporte de la carga.</li></ol>
Precondiciones:	El usuario debe haber subido una carga anteriormente.
Prioridad:	Baja

---

Fuente: Elaboración propia

**Tabla 17**

*Escenario de caso de uso: Registrar usuarios*

---

<b>Registrar usuarios</b>	
ID:	CU10
Actores:	Administrador
Descripción:	Permite ver el reporte de las reservas procesadas
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario ingresa al formulario de registro de usuarios.</li><li>2. El usuario ingresa los datos del nuevo usuario y especifica si será administrador o usuario regular.</li><li>3. El sistema registra los datos.</li></ol>
Precondiciones:	El usuario debe ser administrador.
Prioridad:	Baja

---

Fuente: Elaboración propia

**Tabla 18**

*Escenario de caso de uso: Registrar usuarios*

---

<b>Editar usuarios</b>	
ID:	CU11
Actores:	Administrador
Descripción:	Permite editar usuarios
Pasos a realizar:	<ol style="list-style-type: none"><li>1. El usuario ingresa a la lista de usuarios.</li><li>2. El usuario selecciona al usuario a editar.</li><li>3. El usuario modifica los datos.</li><li>4. El sistema actualiza la información del usuario editado.</li></ol>
Precondiciones:	El usuario debe ser administrador.
Prioridad:	Baja

---

Fuente: Elaboración propia

### ***h) Identificación de incrementos de la aplicación***

Se definieron los incrementos basados en los casos de uso. Se definió el indicador de esfuerzo de 1 a 4, donde 4 es el incremento que demanda mayor esfuerzo.

**Tabla 19**

*Incrementos de la aplicación*

<b>Número</b>	<b>Descripción</b>	<b>Indicador de esfuerzo</b>
1	Iniciar sesión	2
2	Actualizar catálogo de materiales	2
3	Subir carga de reservas	4
4	Asignar personal	1
5	Imprimir reservas	3
6	Recuperar carga	1
7	Descargar archivo	1
8	Ver reserva anterior	2
9	Ver reporte	3
10	Registrar usuarios	2
11	Editar usuarios	2

Fuente: Elaboración propia

### ***i) Definición de incrementos de la aplicación***

Se agruparon los casos de uso de tal forma que la suma del indicador de esfuerzo sea aproximadamente 4.

- Incremento 1: Iniciar sesión, actualizar catálogo de materiales.

- Incremento 2: Subir carga de reservas.
- Incremento 3: Asignar personal, Imprimir reservas.
- Incremento 4: Recuperar carga, descargar archivo, ver reservas.
- Incremento 5: Ver reporte.
- Incremento 6: Registrar usuarios, editar usuarios.

**Tabla 20**

*Tiempo de desarrollo de incrementos*

<b>Incrementos</b>	<b>Duración</b>
1. Iniciar sesión, actualizar catálogo de materiales.	3 días
2. Subir carga de reservas.	3 días
3. Asignar personal, Imprimir reservas.	3 días
4. Recuperar carga, descargar archivo, ver reserva anterior.	3 días
5. Ver reporte.	1 día
6. Registrar usuarios, editar usuarios.	2 días

Fuente: Elaboración propia

### ***3.2.1.3. Negociación***

El objetivo de la negociación es balancear funcionalidad, rendimiento y otras características a cambio de costo y tiempo de entrega.

La aplicación requería estar lista en 15 días, por lo que se optó por incrementar el presupuesto y permitir que se trabaje horas extra para poder cumplir con el tiempo de entrega.

### **3.2.2. Planificación**

En la planificación se define el alcance del proyecto. Se define el equipo web que desarrollará la aplicación, se identifica y administran los riesgos que pueden ocurrir durante el proceso de desarrollo y se define un tiempo de desarrollo.

#### ***3.2.2.1. Construcción de equipo web***

No fue necesario construir un equipo web debido a que todo el análisis y desarrollo se realizó por una persona.

#### ***3.2.2.2. Administración de riesgo***

Se identificaron los riesgos con el fin de dar prioridad a los que tienen mayor impacto de retrasar la entrega de la aplicación. Se realizó una tabla de riesgos, la probabilidad de que ocurran (en porcentaje) y el impacto en la aplicación (del 1 al 4).

**Tabla 21**

*Riesgos en el desarrollo*

<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>
Personas: Sin experiencia para manipular hojas de cálculo en PHP.	100	4
Producto: Funciones de procesamiento de reservas tardan en ejecutarse	80	4
Proceso: Desarrollo de diagramas consume mucho tiempo	50	2

Fuente: Elaboración propia

### ***3.2.2.3. Plan de contingencia***

- a) Sin experiencia para manipular hojas de cálculo en PHP: Se utilizó una librería en PHP fácil de utilizar que manipula hojas de cálculo.
- b) Funciones de procesamiento de reservas tardan en ejecutarse: Se revisaron y mejoraron las funciones de procesamiento de reservas.
- c) Desarrollo de diagramas consume mucho tiempo: Se desarrollaron sólo los diagramas necesarios para entender mejor el proceso de desarrollo.

### ***3.2.2.4. Manejo de la calidad***

Una vez liberado un incremento, éste se evaluó por los usuarios y la retroalimentación se entregó al desarrollador. Esta retroalimentación brindó

información de lo que funcionaba y lo que no. Se evaluaron las opciones del menú, la facilidad de uso, los formularios y los resultados de las cargas de reservas.

#### ***3.2.2.5. Manejo de los cambios***

Debido al proceso incremental utilizado en la ingeniería web, es más fácil aplicar cambios. Todos los cambios solicitados se aplicaron antes de realizar el siguiente incremento.

#### ***3.2.2.6. Seguimiento del proyecto***

Se hizo seguimiento del proyecto determinando cuantos escenarios de usuario se implementaron y cuantos escenarios quedaban por implementar por cada caso de uso.

### **3.2.3. Modelamiento**

En esta etapa se crean modelos que asisten al desarrollador y usuario a entender mejor los requerimientos de la aplicación web.

Se utilizó la información obtenida de la etapa de Comunicación como entrada para la realización del análisis con el fin de obtener diagramas como salida. Se utilizó en Lenguaje de modelamiento unificado (UML) para la creación de los diagramas.



### 3.2.3.1. Diagrama de clases

Muestra las clases del sistema y sus dependencias.

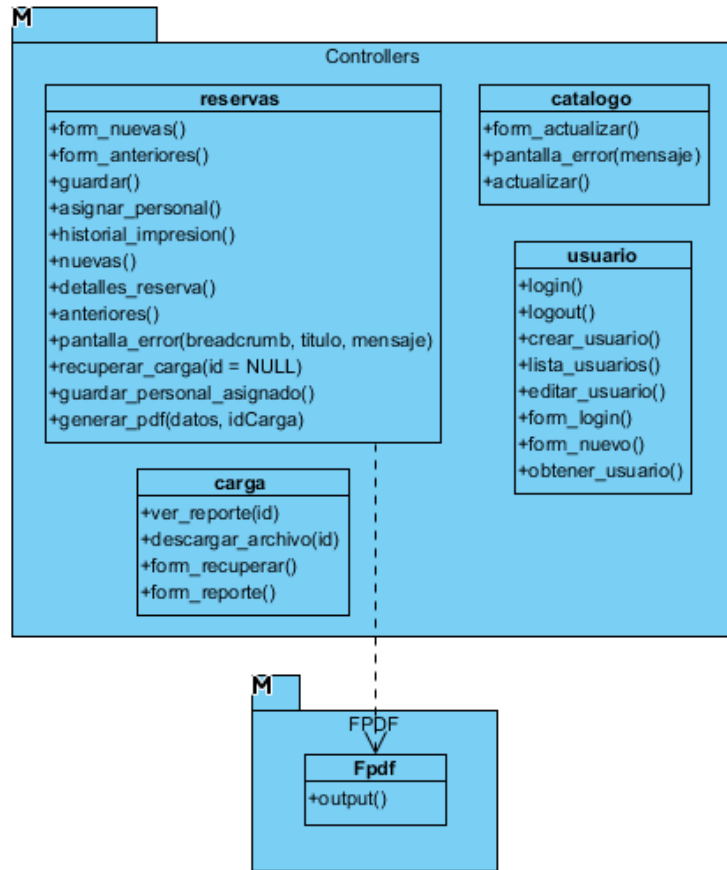


Figura 8. Diagrama de clases - Controladores

Fuente: Elaboración propia

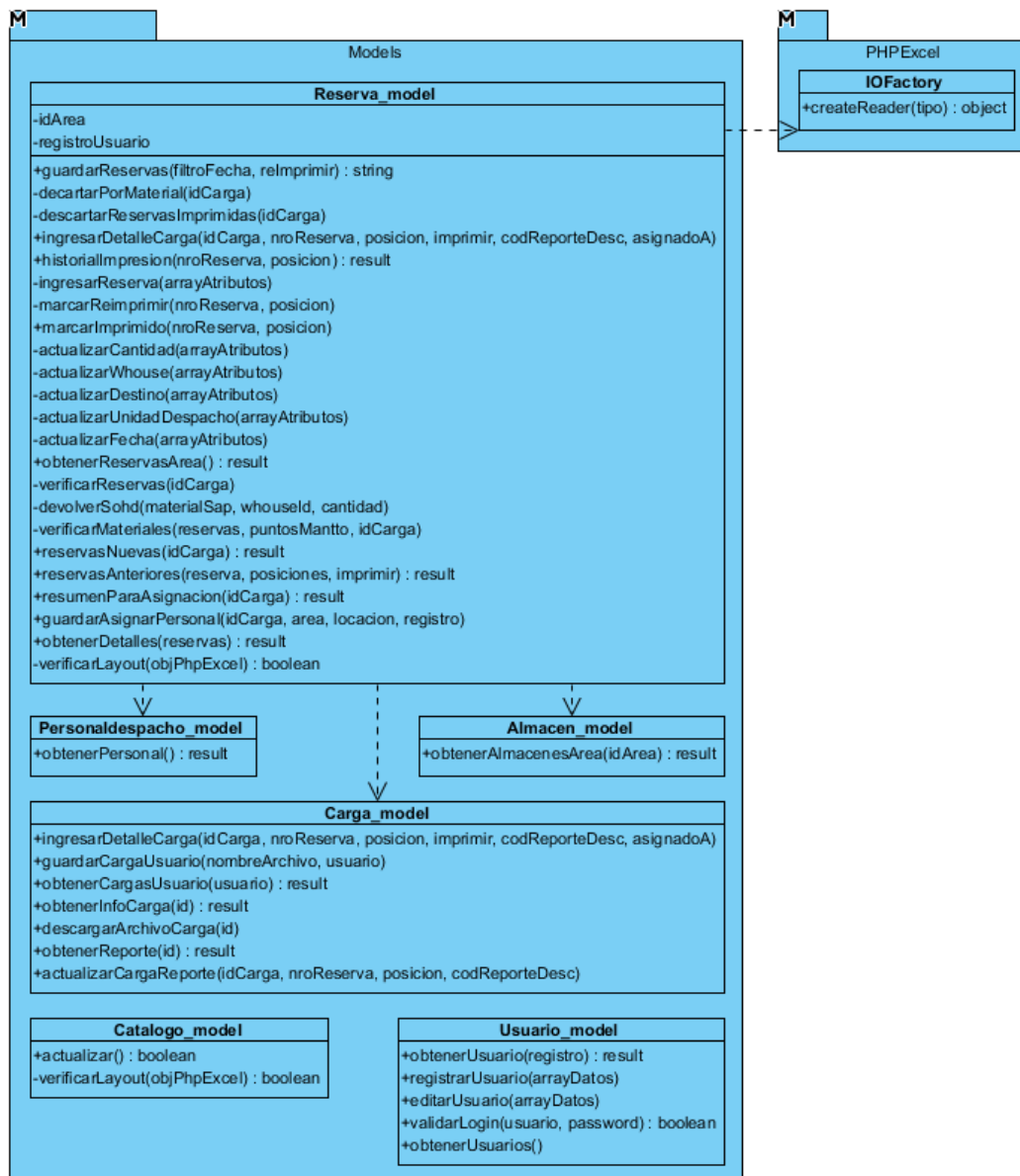


Figura 9. Diagrama de clases - Modelos

Fuente: Elaboración propia

### 3.2.3.2. Diagramas de secuencia

Describen cómo las acciones del usuario colaboran con las clases del sistema.

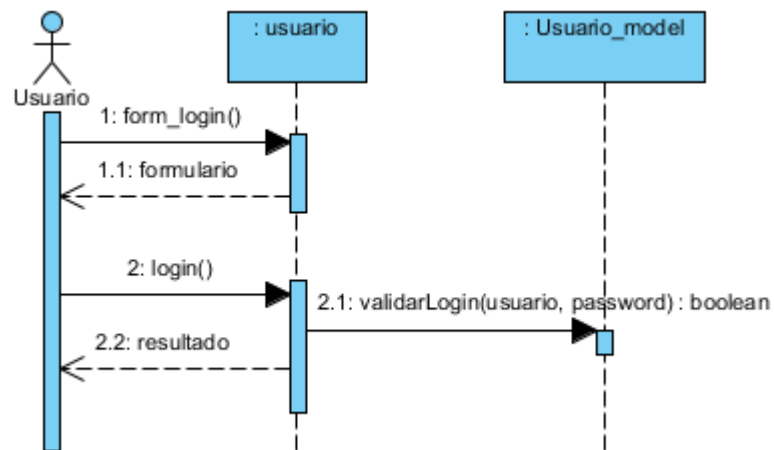


Figura 10. Diagrama de secuencia – Iniciar sesión

Fuente: Elaboración propia

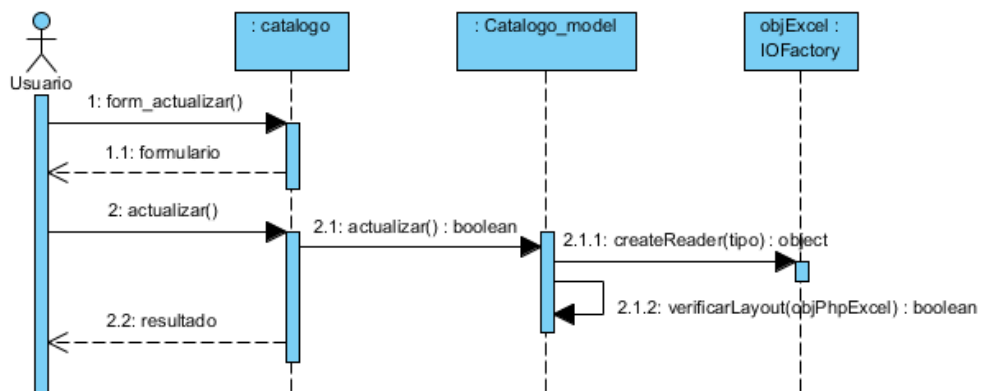


Figura 11. Diagrama de secuencia – Actualizar catálogo

Fuente: Elaboración propia

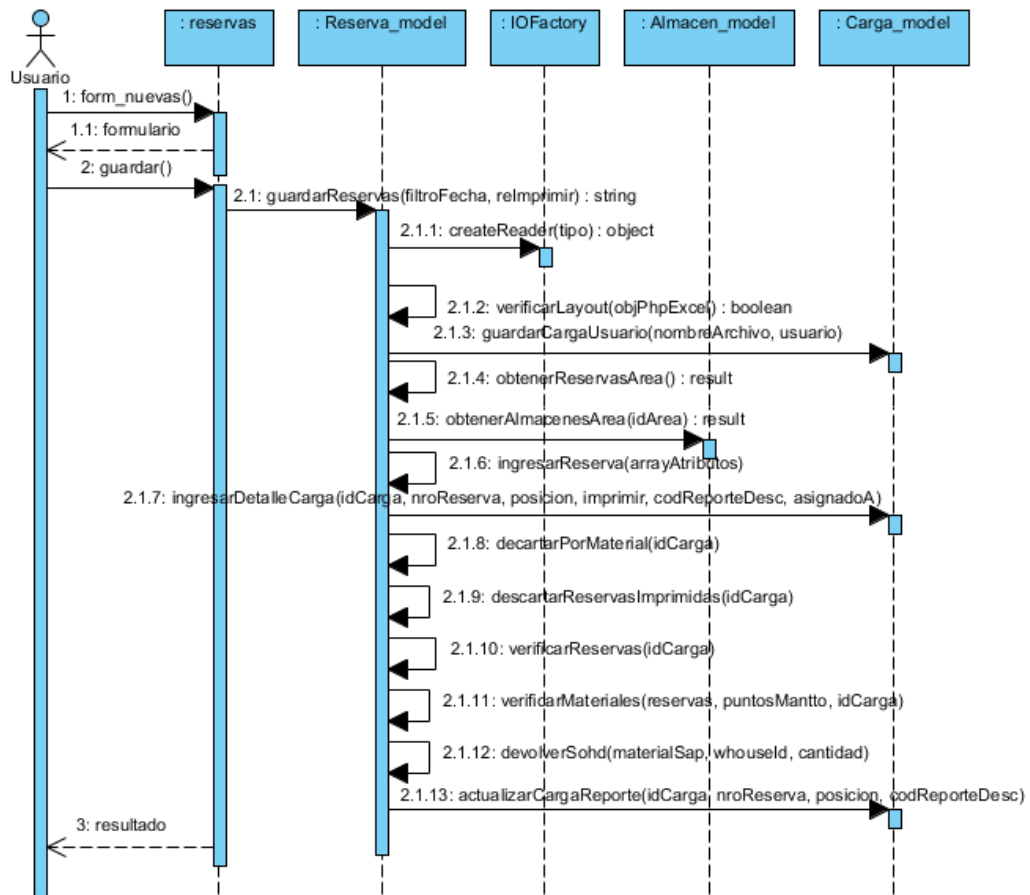


Figura 12. Diagrama de secuencia – Subir carga de reservas

Fuente: Elaboración propia

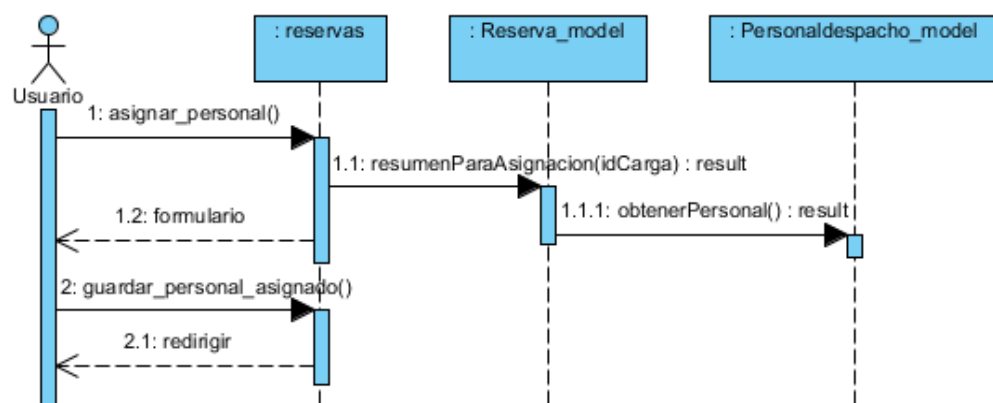


Figura 13. Diagrama de secuencia – Asignar personal

Fuente: Elaboración propia

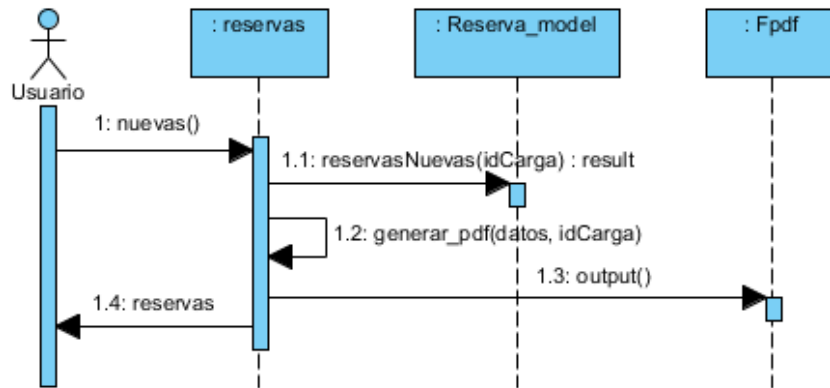


Figura 14. Diagrama de secuencia – Imprimir reservas

Fuente: Elaboración propia

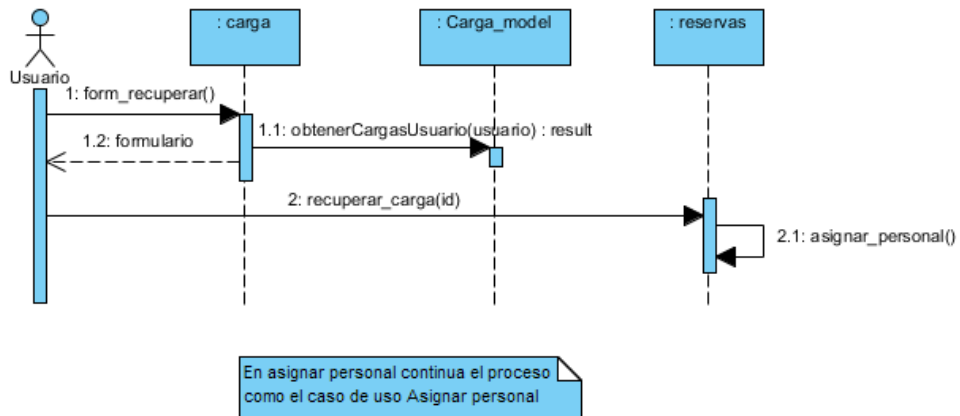


Figura 15. Diagrama de secuencia – Recuperar carga

Fuente: Elaboración propia

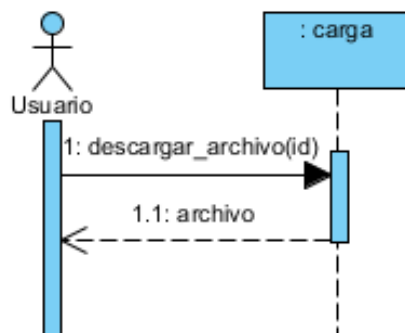


Figura 16. Diagrama de secuencia – Descargar archivo

Fuente: Elaboración propia

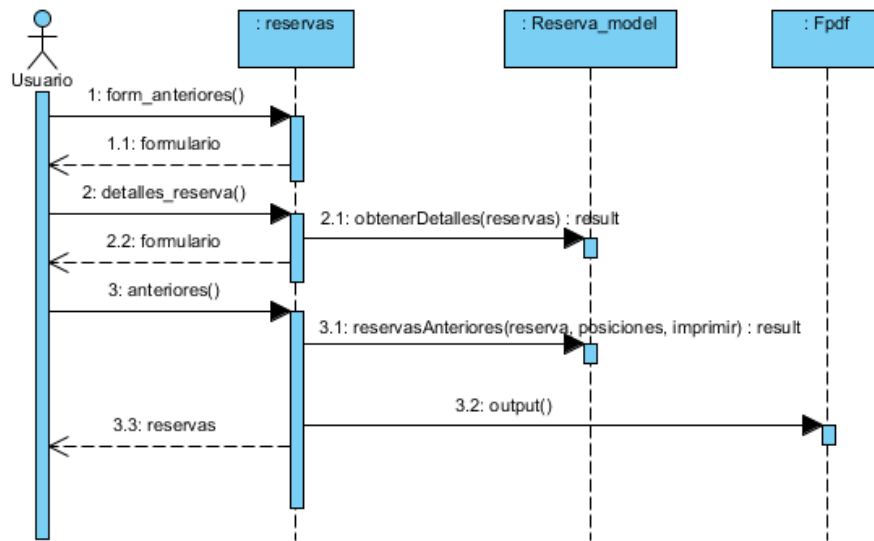


Figura 17. Diagrama de secuencia – Ver reserva anterior

Fuente: Elaboración propia

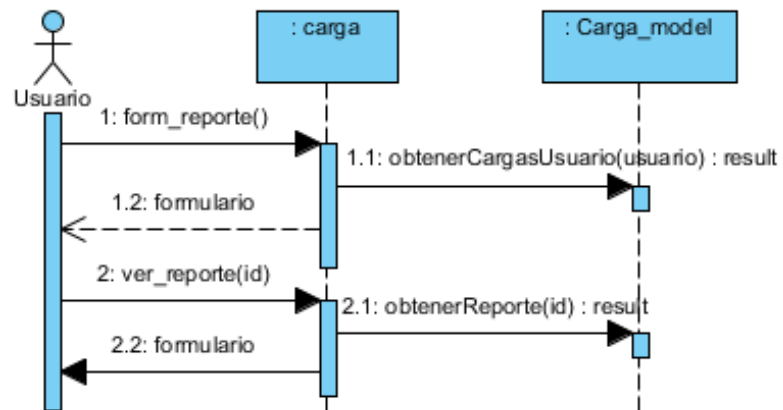


Figura 18. Diagrama de secuencia – Ver reporte

Fuente: Elaboración propia

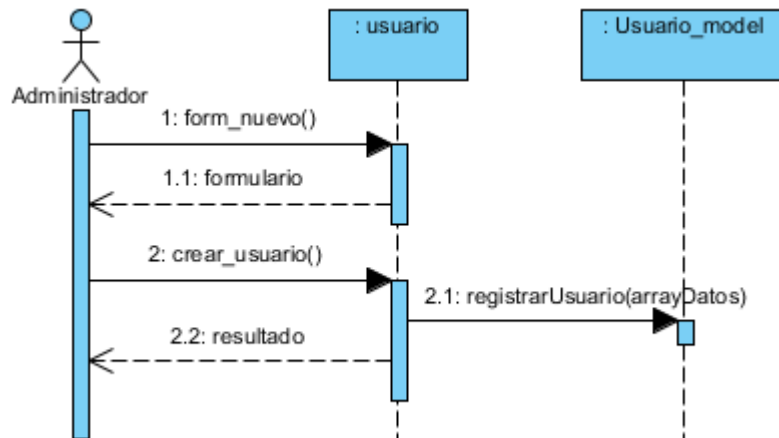


Figura 19. Diagrama de secuencia – Registrar usuarios

Fuente: Elaboración propia

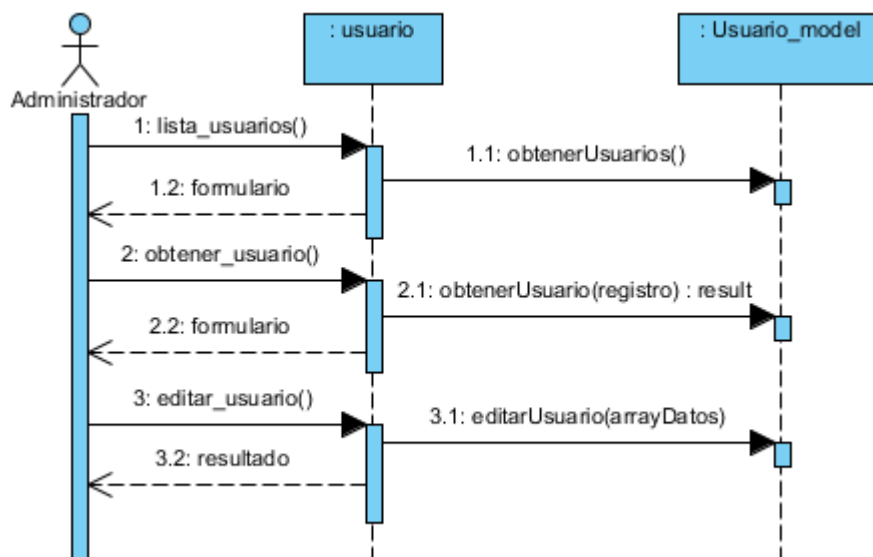


Figura 20. Diagrama de secuencia – Editar usuarios

Fuente: Elaboración propia

### 3.2.3.3. Diagrama entidad-relación

Muestra la estructura de la base de datos del sistema.

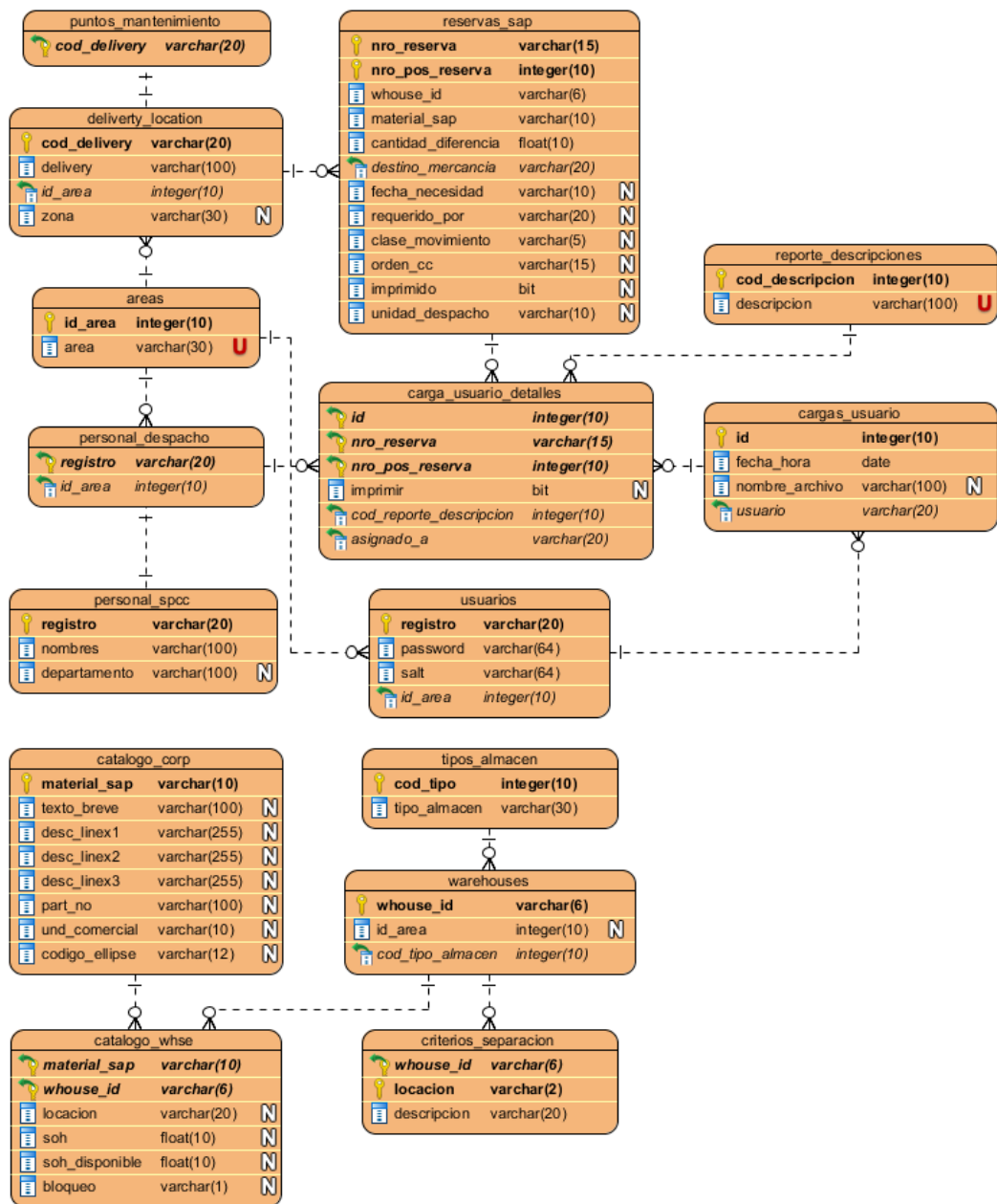


Figura 21. Diagrama entidad-relación del sistema

Fuente: Elaboración propia

### 3.2.3.4. Diagrama de paquetes

Muestra los paquetes del sistema y sus dependencias.



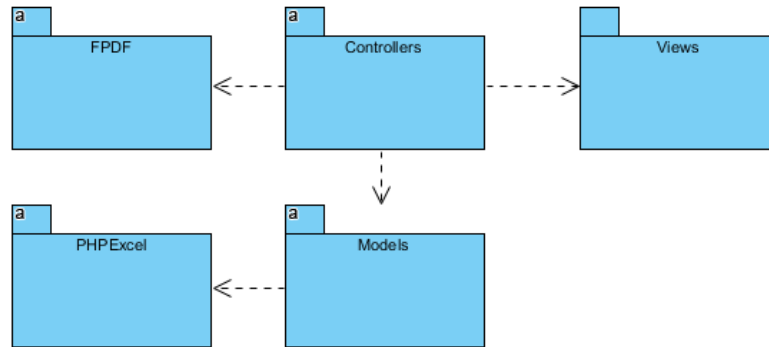


Figura 22. Diagrama de paquetes

Fuente: Elaboración propia

### 3.2.3.5. Diagrama de despliegue

Muestra el diseño del sistema, los componentes y el software que ejecutan.

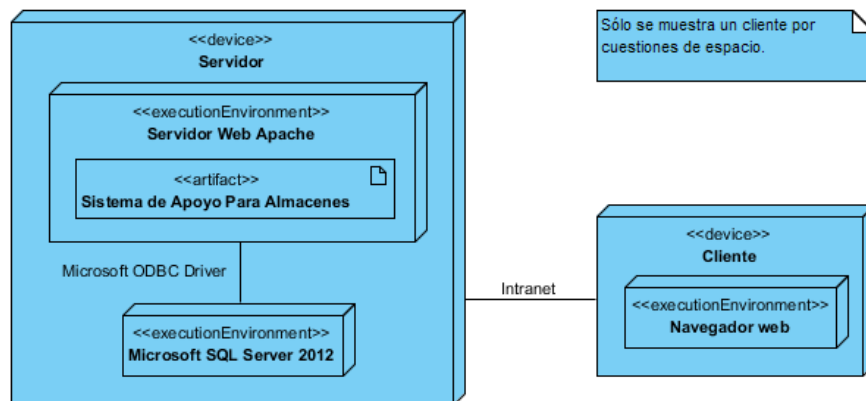


Figura 23. Diagrama de despliegue

Fuente: Elaboración propia

### 3.2.4. Prototipo

Se diseñó un prototipo basado en los escenarios de casos de uso de la Figura 6 para que los usuarios puedan revisar y proveer retroalimentación sobre el diseño, las opciones disponibles y la facilidad de uso.

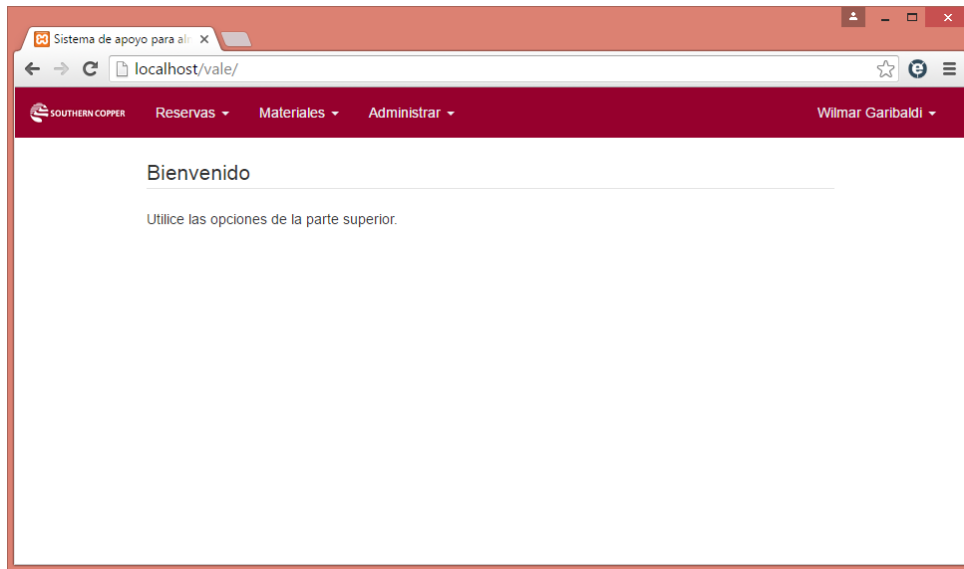


Figura 24. Prototipo del sistema

Fuente: Elaboración propia

### 3.2.5. Modelo de configuración

Tabla 22

Requerimientos para el servidor de producción

Hardware	Software
Procesador de 2GHz o superior	Microsoft Windows Server 2012
Memoria RAM de 3GB o superior	Base de datos SQL Server 2012
Espacio libre en disco de 5GB o superior	Servidor web Apache con PHP
Conectividad a red de 100Mbps o superior	

Fuente: Elaboración propia

**Tabla 23**

*Requerimientos para el cliente*

<b>Hardware</b>	<b>Software</b>
Procesador de 1GHz o superior	Navegador web con soporte para HTML5.
Memoria RAM de 1GB o superior	Lector de archivos PDF
Impresora	
Conectividad a red de 10Mbps o superior	

Fuente: Elaboración propia

### **3.2.6. Construcción**

La construcción desarrolla todos los componentes que deben ser entregados por incremento de la aplicación. La actividad involucra generación de código y pruebas.

#### **3.2.6.1. Diseño de la aplicación**

##### **a) Metas de diseño**

- **Simplicidad:** El diseño de la aplicación se realizó con una combinación de sólo 3 colores: rojo, blanco, negro. Posee sólo un menú que se encuentra en la parte superior de la pantalla y el contenido mostrado es el suficiente para que el usuario pueda trabajar con la aplicación.

- **Consistencia:** Todas las páginas poseen el menú en la parte superior y debajo de éste se encuentra el título de la página en letra grande y subrayada.
- **Identidad:** La estética de la página no posee colores o elementos llamativos, reflejando la seriedad de la empresa.
- **Navegabilidad:** Todas las opciones de navegación se encuentran en el menú ubicado en la parte superior de la pantalla, lo que facilita la navegación entre los formularios de la aplicación.
- **Compatibilidad:** Se utiliza un marco de trabajo para que la página pueda ser utilizada en casi cualquier navegador web sin generar problemas a la hora de su uso.

***b) Dimensiones de la calidad***

- **Seguridad:** El acceso a la aplicación requiere del uso de una contraseña. Las contraseñas están cifradas en la base de datos.
- **Disponibilidad:** La aplicación estará alojada en un servidor de la empresa, el cual es administrado por el área de Tecnologías de Información. Generalmente estos servidores están disponibles todo el día.
- **Escalabilidad:** Debido a que la aplicación se encuentra en servidores de alto rendimiento, no habrá problema si se eleva el número de usuarios del sistema.
- **Estructura:** Todos los enlaces dentro y fuera de la página funcionan. Todos los formularios son accesibles desde el menú.
- **Tiempo de respuesta:** El servidor no necesita parámetros especiales para mostrar la página. El tiempo de respuesta del formulario Asignación de

personal es elevado, dependiendo de la cantidad de reservas a procesar por la aplicación.

- Rendimiento: La conexión al servidor que aloja la aplicación es rápida (1Gbps). El rendimiento de la aplicación se reduce en las cargas de la mañana (aprox 7:30am) y la tarde (aprox 3:30pm), debido a que los almacenes realizan la carga de sus reservas.

**c) *Diseño de interacción***

- Anticipación: La pantalla inicial ofrece un enlace directo a la actualización del catálogo de materiales y de carga de reservas. Cuando el usuario actualiza el catálogo, la página lo redirige a la carga de reservas.
- Comunicación: El sistema muestra mensajes cuando está procesando una operación. También muestra el nombre del usuario que inició sesión.
- Consistencia: La aplicación mantiene su estilo visual en todas las páginas (por ejemplo botones rojos para “Aceptar” o “Siguiente”, tamaño y fuente, menú en la parte superior, nombres de la página subrayados).
- Enfoque: la aplicación muestra en los formularios lo que el usuario necesita, sin mostrar contenido irrelevante.
- Opciones relacionadas: Las opciones que guardan relación entre sí (como *Seleccionar archivo* y *Aceptar*) se encuentran cerca una de otra, con el fin de agilizar el trabajo que realiza el usuario.

- **Facilidad de aprendizaje:** El sistema muestra sólo las opciones necesarias y a veces un texto breve de ayuda para que el usuario pueda aprender fácilmente a usar la aplicación.
- **Mantener integridad del trabajo:** En caso de pérdida de conexión o algún otro fallo, la aplicación permite al usuario recuperar su carga de reservas.

**d) Diseño de interfaz**

Para el diseño de la interfaz, se revisaron las categorías de los usuarios y los casos de uso, y se definieron las siguientes opciones para el menú principal.

- **Reservas:** Nueva carga, Recuperar carga, Reserva anterior, Ver reporte.
- **Materiales:** Actualizar catálogo.
- **Administrar (sólo para usuarios administradores):** Usuarios, Filtro de reservas.

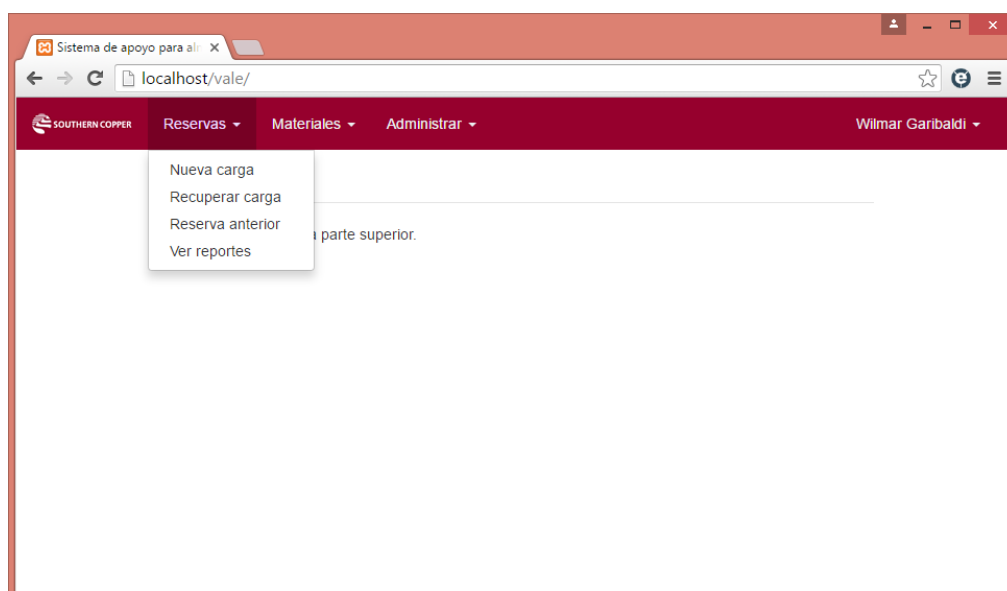


Figura 25. Opciones del menú Reservas

Fuente: Elaboración propia

Es importante que la interfaz sea consistente interna y externamente. Internamente para que el grado de usabilidad sea alto y externamente debido a que existen convenciones sobre las interfaces de aplicaciones y si no se siguen, el usuario tardará más en aprender a utilizar la aplicación.

*e) Diseño de estética*

Complementa los aspectos técnicos del diseño de interfaz y de contenido. Se utilizaron las siguientes pautas para el diseño.

- No se utilizó todo el espacio de la pantalla disponible, sólo lo que el usuario necesita ver.
- Se agrupó la navegación en la parte superior, esto facilita al usuario a entrar a los formularios.
- Se han considerado diferentes resoluciones de pantalla para que los formularios puedan adaptarse.
- Se utiliza una hoja de estilo en cascada, de tal forma que, si se desea cambiar la apariencia, sólo es necesario modificar un archivo para que los cambios se vean en toda la aplicación.

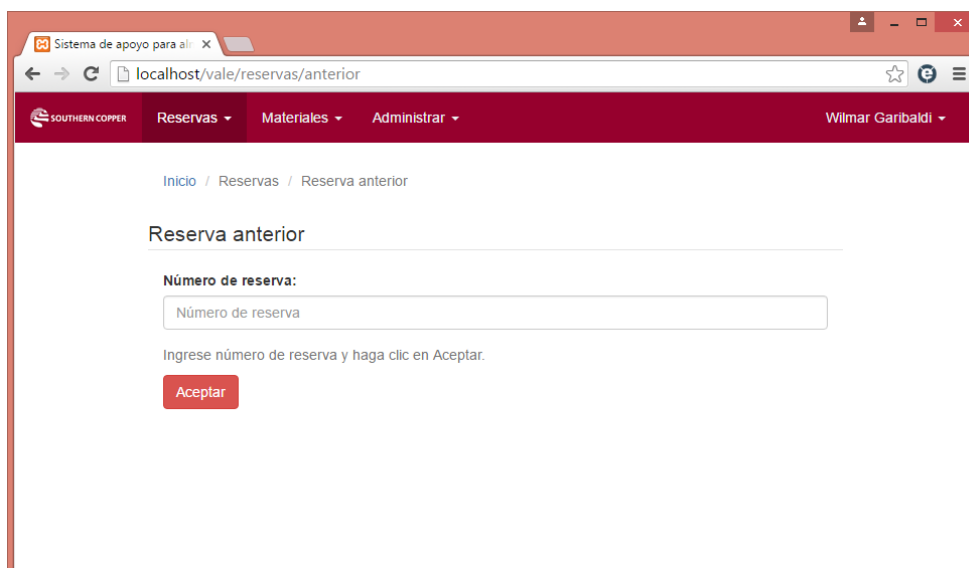


Figura 26. Estética de la aplicación

Fuente: Elaboración propia

#### f) Usabilidad

La usabilidad es la medida de cuan bien un sistema computacional facilita el aprendizaje, reduce la probabilidad de cometer errores y ayuda a ser más eficiente. Para determinar si la usabilidad es alta en la aplicación, se observó a los usuarios interactuar con la aplicación y se respondieron las siguientes preguntas.

- **¿Es la aplicación usable sin ayuda o instrucciones continuas?**

Sí.

- **¿Las reglas de navegación ayudan al usuario experto a trabajar eficiente?**

Sí, debido a que no son muchas las funciones que realiza la aplicación.



- **¿Los usuarios saben en donde se encuentran todo el tiempo?**  
Sí, se sombrea el menú al que accedieron y se muestra un rastro en la parte superior de la pantalla.
- **¿Son los mecanismos de interacción y navegación consistentes en la interfaz?**  
Sí.
- **¿La interacción anticipa los errores y ayuda a los usuarios a corregirlos?**  
Sí, marcando en rojo los campos no llenados además de mostrar mensajes de advertencia.

Para los mensajes de error, se utilizaron las siguientes recomendaciones:

- El mensaje debe describir el problema en términos que los usuarios puedan entender
- El mensaje debe indicar qué opciones tiene el usuario para corregir el problema.
- El mensaje debe estar acompañado de una pista visual, como un ícono o color.
- El mensaje no debe culpar al usuario.

### ***3.2.6.2. Principios de codificación***

Se tomaron en cuenta los siguientes principios para la codificación

- Entender el problema que se intenta resolver.

- Escoger un lenguaje que cumpla con las necesidades de los componentes a construir y el ambiente en el que operará.
- Considerar las habilidades y conocimiento del programador.
- Considerar problemas de propiedad intelectual.
- Escribir código que es auto documentado.
- Mantener la lógica tan simple como sea posible.
- Realizar copias de seguridad.
- Identificar componentes que requieran refactorización.

### ***3.2.6.3. Refactorización***

En la refactorización, se revisa la codificación para mejorar su estructura y claridad, removiendo o modificando código repetido o ineficiente. Se aplicó refactorización a cada incremento antes de ser entregado.

### **3.2.7. Despliegue**

La aplicación es configurada para su entorno operacional, luego es entregada a los usuarios finales y comienza un periodo de evaluación. Los resultados de la evaluación son presentados y el incremento es modificado a petición. Debido a que el desarrollo de una aplicación web es incremental, el despliegue ocurre varias veces.

#### ***3.2.7.1. Ambientes de despliegue***

Para el desarrollo de la aplicación se tuvieron dos ambientes de despliegue.

- a) Computadora de desarrollo: En esta computadora se desarrollaba y probaba la aplicación web, tenía instalado lo necesario para codificar el sistema. Esta computadora se encontraba en red y los usuarios podían acceder a la aplicación para realizar las pruebas respectivas. Una vez terminadas las pruebas, el acceso para los usuarios era cancelado hasta realizar nuevas pruebas.
- b) Servidor de producción: En este equipo se subían las iteraciones una vez desarrolladas.

#### ***3.2.7.2. Tareas de despliegue***

Debido a la naturaleza de la web, una vez que se incluye un incremento de la aplicación, éste inmediatamente se encuentra disponible para todos los usuarios, por lo que se consideró lo siguiente.

***a) ¿Cómo serán anunciados los cambios?***

A través de correo electrónico

***b) ¿Cómo serán los usuarios guiados para el uso de los nuevos cambios?***

Los usuarios aprenden a usar los nuevos cambios en las sesiones de pruebas.

***c) ¿Debería proveerse un mecanismo de retroalimentación para reportes de errores u otros problemas relacionados con los cambios?***

Todos los errores deben reportarse por correo electrónico, adjuntando una captura de la pantalla del mensaje y explicando qué pasos se realizaron para suceda el problema.

### **3.2.7.3. Planificación**

Para la planificación de cada incremento se realizó lo siguiente.

- a) Las fechas de entrega se dieron a conocer por correo electrónico.
- b) Todos los cambios y novedades se avisaron por correo electrónico.
- c) El plan de contingencia fue volver a la versión anterior si es que no se podía corregir el problema de forma oportuna.
- d) Se estableció un canal de soporte por correo electrónico.

### **3.2.7.4. Principios para un despliegue**

Para asegurar un despliegue exitoso, se siguieron los siguientes principios.

- a) La expectativa de los clientes por el incremento debe ser manejada: Generalmente el cliente tiene altas expectativas del incremento y si no ve lo que pensaba, se desilusionará, lo que hará que sea menos colaborativo a la hora de despejar dudas. Para manejar las expectativas es necesario ser claro al comunicar de lo que tratará cada incremento.
- b) Un incremento debe ser probado antes de entregarse.
- c) Debe establecerse un régimen de soporte antes que la aplicación sea entregada: Los usuarios esperan que se resuelvan sus problemas cuando ocurran. Si el soporte es malo o inexistente, los usuarios quedarán insatisfechos con la aplicación.
- d) Las aplicaciones defectuosas deben corregirse: Cuando un incremento es puesto en uso, los usuarios deben reportar todo error o problema que

encuentren para realizarse las modificaciones de manera inmediata o ser incorporadas para el siguiente incremento.

#### ***3.2.7.5. Control de versiones***

Con cada versión liberada, se guardó una copia de seguridad en caso haya que revertir los cambios. Estas copias se guardaron en una memoria USB y en la computadora de desarrollo.

#### **3.2.8. Tecnología y herramientas**

En este apartado se describen las herramientas de implementación y las herramientas de desarrollo utilizadas en la aplicación web.

##### ***3.2.8.1. Marco de trabajo***

El marco de trabajo utilizado que implementa el Modelo Vista Controlador (MVC) es CodeIgniter, versión 3.0.6.

##### ***3.2.8.2. Software gratuito***

Se utilizaron los siguientes programas gratuitos: Oracle Netbeans para la interfaz de desarrollo y XAMPP para el servidor web.

##### ***3.2.8.3. Software licenciado***

Para el SGBD, se utilizó Microsoft SQL Server 2012, debido a que la empresa cuenta con la licencia respectiva. Para los diagramas UML, se utilizó Visual Paradigm Professional Edition en periodo de evaluación.

### **3.2.9. Pruebas**

Se realizaron diferentes pruebas al sistema para asegurar que cumpla con los requerimientos del usuario y así lograr un producto de calidad.

#### ***3.2.9.1. Dimensiones de la calidad***

Se revisaron las siguientes dimensiones.

- a) Gramática y puntuación: Errores ortográficos.
- b) Resultados de las funciones: Separar las reservas de forma manual y ver si los resultados de la aplicación coinciden con los resultados obtenidos.
- c) Usabilidad: Revisar que cada categoría de usuario este soportada.
- d) Navegabilidad: Ver si todos los enlaces funcionan correctamente.
- e) Rendimiento: Verificar el tiempo de respuesta cuando múltiples usuarios realizan la carga de sus reservas.
- f) Compatibilidad: Revisar si la interfaz de la aplicación funciona en diferentes navegadores web.
- g) Seguridad: Ver si las opciones de la aplicación pueden ser utilizadas sin haber iniciado sesión, ver si las opciones de administrador pueden ser accedidas por un usuario regular.

#### ***3.2.9.2. Pruebas de contenido***

Se realizaron pruebas de contenido, que revisan errores tipográficos, información incorrecta y organización inapropiada, con el fin de descubrir errores de sintaxis,

semánticos (información incompleta no imprecisa) y de organización del contenido.

- a) Errores ortográficos. Información valida es pasada entre el cliente y la aplicación.
- b) La aplicación envía al servidor de base de datos los scripts de forma correcta.
- c) La base de datos devuelve el resultado obtenido de la consulta sin problemas.
- d) La aplicación recibe, formatea y transfiere el resultado al cliente de manera correcta.

### ***3.2.9.3. Pruebas de la interfaz de usuario***

Se verificó lo siguiente.

- a) Enlaces: Cada enlace se probó para verificar si ingresaba al formulario apropiado.
- b) Formularios: Se verificó que las etiquetas identifican correctamente a los campos y que todos los campos necesarios están disponibles para el usuario. Se verificó que el servidor recibe toda la información contenido en el formulario.
- c) Contenido dinámico: Cada página que genera contenido dinámico se ejecutó para verificar si el contenido es correcto.
- d) Cookies: Se verificó si la sesión sigue abierta a pesar que el usuario cerró sesión o si cerró la ventana del navegador.

#### ***3.2.9.4. Pruebas de compatibilidad***

Se realizaron pruebas ejecutando la aplicación en diferentes navegadores: Microsoft Internet Explorer 8 y superior, Google Chrome y Mozilla Firefox. La aplicación se ejecutó sin problemas en los navegadores mencionados.

#### ***3.2.9.5. Pruebas de navegación***

Se probó lo siguiente.

- a) Enlaces de navegación: Todos los enlaces funcionaron correctamente.
- b) Redirecciones: Las redirecciones que realizaba la aplicación funcionan correctamente.
- c) Accesos directos: Los accesos directos que creaban los usuarios en sus escritorios para acceder a la aplicación funcionan correctamente.

### **3.3. Representación de resultados**

Con la utilización de la ingeniería web se pudo desarrollar la aplicación en el tiempo establecido, cumpliendo con todos los requerimientos solicitados y superando las expectativas de los usuarios.



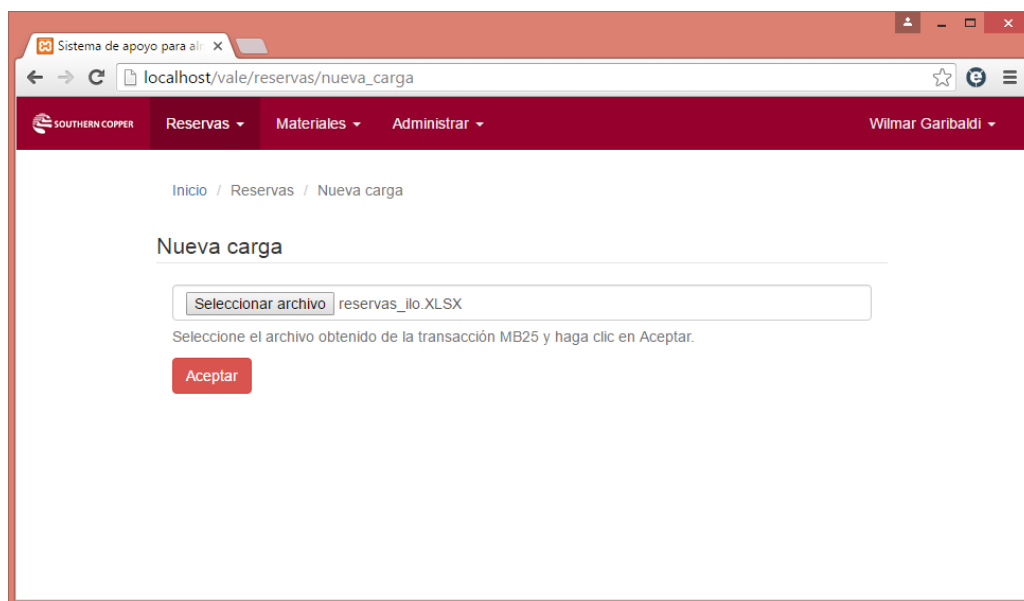


Figura 27. Formulario Nueva carga

Fuente: Elaboración propia

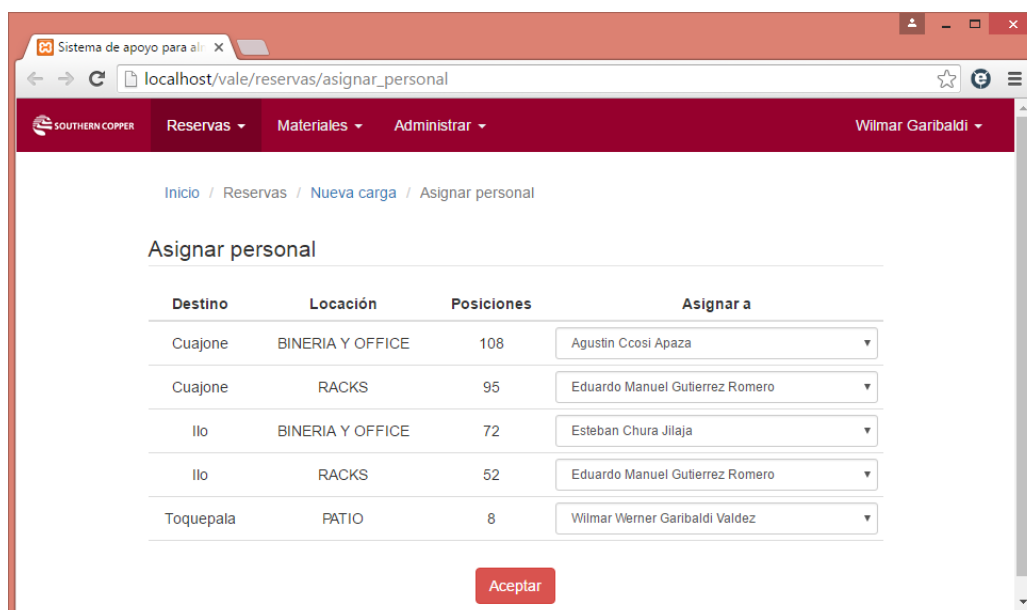


Figura 28. Formulario Asignar personal

Fuente: Elaboración propia

**RESERVA SAP - S3IL - 261**

<b>Área destino:</b>	Cujajone	<b>Locación:</b>	BINERIA Y OFFICE
<b>Punto de entrega:</b>	402M/CUA-TRAC-02-TRACTORES	<b>Asignado a:</b>	9710 Agustín Ccosi Apaza
<b>Requerido por:</b>	92361 Jorge Abraham Salas Matos	<b>Fecha de despacho:</b>	

Reserva	Pos.	Material	C. Ellip.	Locación	SOH	C. Dif.	Desp.	Descripción	U. M.	Part N°	Orden	Fecha
750203	106	30253265	000091074	B1AD0214	10	1		DADO DE ENCASTRE DE IMPACTO	SET	311MYA	8000010983	2016-01-11
750203	97	30266306	000326702	B1AD03H1	10	1		DADO DE ENCASTRE	CAJ	5528	8000010983	2016-01-11
750203	98	30265907	000319707	B1AD03H2	10	1		DADO DE ENCASTRE	CAJ	5530	8000010983	2016-01-11
750203	100	30259254	000199828	B1AD03H4	10	1		DADO DE ENCASTRE	CAJ	5534	8000010983	2016-01-11
750203	101	30249915	000030122	B1AD03H1	10	1		DADO DE ENCASTRE	CAJ	5536	8000010983	2016-01-11
750203	105	30254683	000116285	B1AD0314	10	1		DADO DE ENCASTRE	CAJ	5542	8000010983	2016-01-11
750203	81	30252683	000080663	B1AD04E1	10	1		DADO DE ENCASTRE	CAJ	5412	8000010983	2016-01-11
750203	82	30260121	000215194	B1AD04E2	10	1		DADO DE ENCASTRE	CAJ	5414	8000010983	2016-01-11
750203	83	30256607	000151019	B1AD04E3	10	1		DADO DE ENCASTRE	CAJ	5416	8000010983	2016-01-11
750203	85	30001938	000138883	B1AD04E5	10	1		SOCKET, STANDARD, INCH	CAJ	5420	8000010983	2016-01-11
750203	88	30257060	000159939	B1AD04E6	10	1		DADO DE ENCASTRE	CAJ	5422	8000010983	2016-01-11
750203	89	30265229	000307769	B1AD04F1	10	1		DADO DE ENCASTRE	CAJ	5424	8000010983	2016-01-11
750203	90	30263508	000276568	B1AD04F2	10	1		DADO DE ENCASTRE	CAJ	5426	8000010983	2016-01-11
750203	92	30001256	000088682	B1AD04F5	10	1		DADO DE ENCASTRE	CAJ	5432	8000010983	2016-01-11

Figura 29. Imprimir reservas

Fuente: Elaboración propia

El uso de metodologías ágiles y disciplinadas redujo el riesgo de cometer errores, acortaron el tiempo de entrega, y aumentaron la calidad del producto.

Las pautas brindadas por la ingeniería web sobre diseño, despliegue y pruebas permitieron que se desarrolle una aplicación fácil de utilizar, que cumpla las expectativas de los usuarios y que satisfaga sus necesidades.

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1. Conclusiones

**Primera.** Con el uso del sistema web se puede completar la información requerida para proceder con el despacho de las reservas de materiales. Los filtros aplicados en el procesamiento de las cargas ahorran tiempo al personal de despacho a no atender reservas sin stock o con problemas de inventario. La agrupación de reservas por usuario y punto de entrega permite el ahorro de papel a la hora de imprimir las reservas. La separación de reservas por locación de materiales permite asignar las reservas al personal responsable por locación en almacén.

**Segunda.** El uso del marco de trabajo propuesto por la Ingeniería Web redujo el riesgo de cometer errores, acortó el tiempo de entrega y aumentó la calidad del producto. Las pautas brindadas sobre diseño, despliegue y pruebas permitieron que se desarrolle una aplicación fácil de utilizar, que cumpla las expectativas de los usuarios y que satisfaga sus necesidades.

## **4.2. Recomendaciones**

**Primera.** Es recomendable el uso de sistemas informáticos para automatizar tareas repetitivas, debido a que se reduce el tiempo de ejecución y la tasa de errores en comparación con la realización de estas tareas de forma manual.

**Segunda.** Es recomendable utilizar un marco de trabajo para desarrollar sistemas informáticos por los beneficios que otorga como mejora de calidad, seguimiento, entregas a tiempo, reducción de errores, entre otros.

## REFERENCIAS BIBLIOGRÁFICAS

- Abad, S. (2010). *Intranet*. Recuperado de <https://elblogenelaula.wordpress.com/2010/01/30/%C2%BFque-es-una-intranet-y-que-aporta-a-un-centro-educativo/>
- Adobe. (2016). *Qué es el formato PDF*. Recuperado de <https://acrobat.adobe.com/la/es/why-adobe/about-adobe-pdf.html>
- Alegsa, L. (2016). *Definición de cliente/servidor*. Recuperado de <http://www.alegsa.com.ar/Dic/cliente%20servidor.php>
- Aliaga, J., Aramburu, M., Marqués, M., & Quintana, G. (2008). *Aprende SQL*. (1a ed.). España: Universitat Jaume.
- Booch, G., Jacobson, I., & Rumbaugh, J. (2005). *The Unified Modeling Language Reference Manual* (2a ed.). Massachusetts: Pearson.
- Castro, L. (2015). *¿Qué es HTML?*. Recuperado de <http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Html.htm>
- Chamorro, S. (2016). *¿Qué es SAP y para qué sirve?*. Recuperado de <http://www.deustoformacion.com/blog/gestion-empresas/que-es-sap-para-que-sirve>
- Debrauwer, L., & Van Der Heyde, F. (2013). *UML 2. Iniciación, Ejemplos y Ejercicios Corregidos* (3a ed.). Luxemburgo: Eni.
- Fumàs, E. (2014). *¿Qué es PHP? ¿Para qué sirve?*. Recuperado de <http://www.ibrugor.com/blog/que-es-php-para-que-sirve/>
- Informática Hoy. (s. f.). *Qué es un navegador web*. Recuperado de <http://www.informatica-hoy.com.ar/aprender-informatica/Que-es-un-navegador-web.php>
- Kendall, J., & Kendall, K. (2011). *Análisis y diseño de sistemas* (8a ed.). México: Pearson.
- Korth, H., Silberschatz, A., & Sudarshan, S. (2011). *Database System Concepts* (6a ed.). New York: McGraw-Hill.
- Lanzillotta, A. (s.f.). *Definición de Archivo*. Recuperado de <http://www.mastermagazine.info/termino/3900.php>
- Lara, O. (2011). *ERP: Planificación de recursos empresariales*. Recuperado de <http://www.gestiopolis.com/erp-planificacion-de-recursos-empresariales/>
- Laudon, J., & Laudon, K. (2012). *Sistemas de información gerencial* (12a ed.). México: Pearson.

- Lowe, D., & Pressman, R. (2009). *Web Engineering: A practitioner's approach* (1a ed.). New York: McGraw-Hill.
- Muriel, J. (s.f.). *Aplicaciones de las hojas de cálculo electrónico a la enseñanza de las matemáticas*. Recuperado de <http://thales.cica.es/rd/Recursos/rd98/Matematicas/18/matematicas-18.html>
- Naula, J. (2010). *¿Qué es un framework de desarrollo web?*. Recuperado de <http://kokiing.blogspot.pe/2010/12/que-es-un-framework-de-desarrollo-web.html>
- Nevado, V. (2010). *Introducción a las bases de datos relacionales*. (1a ed.). España: Vision Libros.
- Sánchez, A. (2016). *Procesador, ¿Qué es, cómo funciona y cuál comprar?*. Recuperado de <http://computadoras.about.com/od/conoce-procesadores/a/Que-Es-Un-Procesador.htm>
- Urraco, J. (2013). *Tutorial CodeIgniter 1º parte: Modelo MVC y primeros pasos*. Recuperado de <http://www.adwe.es/codigo/tutorial-codeigniter-1o-parte-modelo-mvc-y-primeros-pasos>.
- Vasquez, B. (2011). *Sistema operativo*. Recuperado de <https://solvasquez.wordpress.com/2011/01/24/definicion-de-sistema-operativo/>
- Vergara, J. (2016). *Qué es CodeIgniter y cuáles son algunas de sus ventajas*. Recuperado de <https://www.coriaweb.hosting/codeigniter-cuales-algunas-ventajas/>
- Wikipedia. (2016). *Memoria de acceso aleatorio*. Recuperado de [https://es.wikipedia.org/wiki/Memoria\\_de\\_acceso\\_aleatorio](https://es.wikipedia.org/wiki/Memoria_de_acceso_aleatorio)
- Wikipedia. (2016). *Prototipo*. Recuperado de <https://es.wikipedia.org/wiki/Prototipo>