SERVICIO NACIONAL DE APRENDIZAJE – CENTRO ELECTRICIDAD

ELECTRONICA Y TELECOMUNICACIONES

SORTING SOFTWARE

NUPIA MARTINEZ GABRIEL

RODRIGUEZ BAQUERO CAMILO ANDRES

SALAMANCA SANTOS CRISTIAN SEBASTIAN

SIERRA WATSON ANGYE VANESSA

ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN

(ADSI)

FABIAN ALEJANDRO RODRIGUEZ

17/06/2020

BOGOTÁ D.C

**HOJA DE CONTROL**

|  |  |  |  |
| --- | --- | --- | --- |
| **Organismos** | Servicio Nacional de Aprendizaje (SENA) | | |
| **Proyecto** | Sorting Software | | |
| **Entregable** | Informe técnico | | |
| **Autor** | Grupo 7 – Ficha 1834732 (ADSI) | | |
| **Aprobado por** |  | **Fecha Aprobación** |  |
|  |  | **N° Total de Páginas** | 23 |

REGISTRO DE CAMBIOS

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Causa del cambio** | **Responsable del cambio** | **Fecha del cambio** |
| 2.0 | Nuevos elementos | Todos los integrantes | 17/06/2020 |
|  |  |  |  |
|  |  |  |  |

CONTROL DE DISTRIBUCIÓN

|  |
| --- |
| **Nombres y Apellido** |
| Angye Sierra Watson |
| Camilo Baquero Rodriguez |
| Gabriel Nupia Martinez |
| Sebastian Salamanca Santos |
|  |

CAPITULO I …………………………………………………………………...……… pág4

OBJETIVOS …………………………………………………………………...…….… pág5

1. OBJETIVO GENERAL ………………………………...……………...……… pág5
2. OBJETIVOS ESPECÍFICOS ……………………..…………………...……… pág5

INTRODUCCIÓN ……………………………………………………………...……… pág6

PLANTEAMIENTO DEL PROBLEMA ………………….…………………...……… pág6

¿A QUIÉN AFECTA? ……………………..…………………………………...……… pág6

SOLUCIÓN ………………..…………………………………………………...……… pág7

CAPITULO II ………………...………………………………………………...……… pág7

[2.1 HERRAMIENTAS](#HERRAMIENTAS) …………………….…………………………………...……… pág7

2.2 PROGRAMACIÓN EN PHP ……………………………………………...……… pág7

2.2.1 INTRODUCCION ……………………..………………………………...……… pág7

2.2.2 VENTAJAS ……………….……………………………………………...……… pág7

2.2.3 DESVENTAJAS ………………………………………………………....……… pág8

2.3 PROGRAMACIÓN EN JAVA ……………………………………..……...……… pág8

2.3.1 INTRODUCCIÓN ……………………………...………………………...……… pág8

2.3.2 VENTAJAS ………………………….…………………………………...……… pág9

2.3.3 DESVENTAJAS ………………………….……………………………...……… pág9

CAPITULO III ……………….……………………………………..………...……… pág11

3.1 CASOS DE USO …………………………………………………………….…… pág11

3.2 BPMN …………………………………………………………………………….. pág18

3.3 MAPA DE PROCESOS ………………………………………………………….. pág19

3.4 DIAGRAMA DE DESPLIEGUE ………………………………………………... pág20

3.5 DIAGRAMA DE CLASES ………………………………………………………. pág21

3.6 MODELO ENTIDAD RELACION ……………………………………………… pág22

3.7 CRONOGRAMA Y USO DE RECURSOS DE PROYECTO …………………... pág21

3.7.1 COSTOS DEL PROYECTO …………………………………………………… pág21

CAPITULO IV …………….……………………………..…………………...……… pág24

4.1 ELABORACIÓN DEL SOFTWARE ………………………..…………...……… pág24

4.1.1 HOSTING ……………………………...…………..…………………...……… pág24

4.1.2 SERVIDOR WEB ………………………………….....………………...……… pág25

CAPITULO V ……………………..…………………………………..……...……… pág26

5.1 IMPLEMENTACIÓN DE PRUEBAS …………………..……………….……… pág26

5.2 ESTÁNDARES DE PROGRAMACIÓN ……………………..…..……...……… pág26

5.3 PRUEBAS DE CAJA NEGRA …………………..………………..……...……… pág27

CONCLUSIÓN …………………………………………………………..…...……… pág28

BIBLIOGRAFIAS ………………………………………………………..…...……… pág29

**OBJETIVOS GENERALES**

Generar un software para la gestión de inventarios capaz de producir reportes, que cuente con gran eficacia desarrollando procesos sin dejar de lado una interfaz simple y clara que le permita al usuario navegar por ella sin fallas, así facilitando su trabajo y optimizando el tiempo.

**OBJETIVOS ESPECIFICOS**

* Proveer un software capaz de generar reportes de los procesos implícitos tanto en la entrada como en la salida de los libros
* Agregar nuevas características a nuestro software de acuerdo con las necesidades que solicite el usuario y vayan surgiendo al trascurso de la elaboración de este.
* Implementar un sistema eficaz al momento de registrar los libros obtenidos

**INTRODUCCIÓN**

La biblioteca de la institución educativa José Acevedo Y Gómez se enfoca en brindarle a esta, información a partir de artículos, revistas, libros y demás; ya que es el único punto donde ellos pueden tener un espacio adecuado para obtener todo tipo de información.

En la actualidad la biblioteca del colegio no se encuentra automatizada en el proceso general de la gestión de inventarios como son entrada de libros, salida de libros, organización, codificación etc. Esta cuenta con una pequeña base de datos de excel donde se registran los libros que van llegando a la biblioteca y los préstamos de los libros.

**PLANTEAMIENTO DEL PROBLEMA**

La biblioteca presenta falencias notorias, debido a que no cuenta con un sistema el cual regule y controle su estructura, generando inconvenientes en el desarrollo de procesos tan esenciales como el registro del muy constante flujo de libros; por ende, se requiere de un sistema el cual permita una adecuada organización sin pasar por alto una interfaz clara y sencilla.

**¿A QUIEN AFECTA?**

* A los estudiantes
* Maestros
* A los encargados de la biblioteca
* A los administradores de la biblioteca

**SOLUCIÓN**

**HERRAMIENTAS**

En nuestro software vamos a utilizar las siguientes herramientas (Cabe aclarar que las siguientes herramientas son libres):

**PROGRAMACIÓN EN PHP**

**INTRODUCCIÓN**

**PHP** es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor, pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica.

**VENTAJAS**

* Es un lenguaje multiplataforma.
* Completamente orientado a la web.
* Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
* Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
* Permite aplicar técnicas de programación orientada a objetos.
* Biblioteca nativa de funciones sumamente amplia e incluida.

**DESVENTAJAS**

* Consume muchos recursos (y por tanto es ligeramente más lento) llamando y ejecutando una función que ejecutando código que encuentra en línea, embebido en el script. Por eso no tiene sentido usar funciones salvo que efectivamente se vaya a utilizar esa porción de código varias veces.

**PROGRAMACIÓN EN JAVASCRIPT**

**INTRODUCCIÓN**

**JavaScript (JS)** es un lenguaje de programación ligero e interpretado, orientado a objetos con [funciones de primera clase](https://en.wikipedia.org/wiki/First-class_functions). Aunque es más conocido como el lenguaje de scripting para páginas web, muchos [entornos no relacionados con el navegador](http://en.wikipedia.org/wiki/JavaScript#Uses_outside_web_pages)también lo usan, tales como [node.js](http://nodejs.org/), [Apache CouchDB](http://couchdb.apache.org/) y [Adobe Acrobat](https://www.adobe.com/devnet/acrobat/javascript.html). Es un lenguaje script multiparadigma, [basado en prototipos](https://en.wikipedia.org/wiki/Prototype-based), dinámico, soporta estilos estilos orientados a objetos, imperativos y declarativos. [Leer más sobre JavaScript](https://developer.mozilla.org/es/docs/JavaScript/Acerca_de_JavaScript).

JavaScript no debe ser confundido con el [lenguaje de programación Java](http://en.wikipedia.org/wiki/Java_(programming_language)). Ambos "Java" y "Javascript" son marcas registradas de Oracle en Estados Unidos y otros países. Sin embargo, los dos lenguajes de programación tienen sintaxis, semántica y usos muy diferentes.

**VENTAJAS**

* Con JavaScript, podrá hacer muchas cosas para que sus páginas Web sean más interactivas y proporcionen una experiencia más interesante a los visitantes de su sitio.
* JavaScript permite crear una interfaz de usuario activa, lo que ofrece retroalimentación a los visitantes según navegan por sus páginas.
* Puede utilizar JavaScript para asegurarse de que los usuarios introducen información válida en los formularios, lo que dará como resultado ahorro en tiempo.
* Si sus formularios requieren cálculos, puede realizarlos con JavaScript en la computadora del usuario, sin necesidad de realizar procesos del lado del servidor.

**DESVENTAJAS**

* JavaScript no permite leer o escribir archivos en las máquinas cliente. En gran medida eso es bueno, porque seguramente no le gustaría que una página Web pudiera leer los archivos de su disco duro, o que fuera capaz de escribir virus en su disco o capaz de manipular los archivos de su computadora. La única excepción es que Javascript puede escribir en el archivo cookie del navegador, pero incluso esta característica tiene sus limitaciones.
* Javascript no permite la escritura de archivos en las máquinas servidor. Hay varias formas de hacer esto (como almacenando los contadores de impactos de la página o rellenando formularios de datos), pero JavaScript no tiene permitido hacer estas cosas. En cambio, necesitará tener un programa en su servidor para manipular y almacenar estos datos. Por ejemplo, puede tratarse de un CGI escrito en un lenguaje como Perl o [PHP](https://disenowebakus.net/aprender-php-hypertext-preprocessor.php).
* Javascript no puede cerrar una ventana que no ha abierto. Esto es así para evitar que un sitio tome el control de su navegador y empiece a cerrar ventanas de otros sitios.
* JavaScript no puede leer información de una página Web abierta procedente de otro servidor.

**CASOS DE USO**

Muestra las interrelaciones entre el sistema y su ambiente, además sirve como un contrato entre el cliente y los diseñadores. Es considerado esencial al iniciar las actividades de análisis, diseño y prueba; este modelo es realizado en la disciplina de Requerimientos.

A picture containing text

Description automatically generated A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A picture containing text

Description automatically generated A close up of a piece of paper

Description automatically generated A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

A close up of text on a white background

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A close up of a map

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

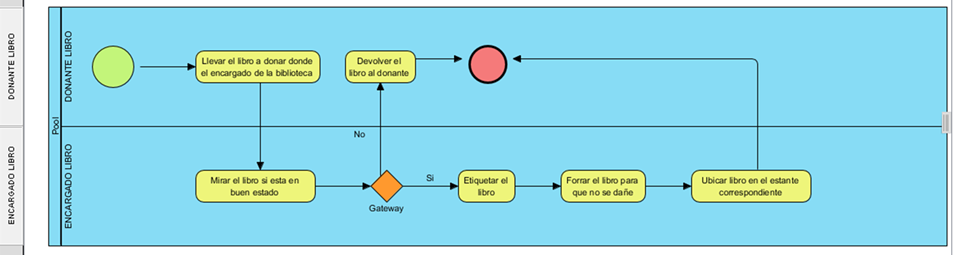
A screenshot of a cell phone

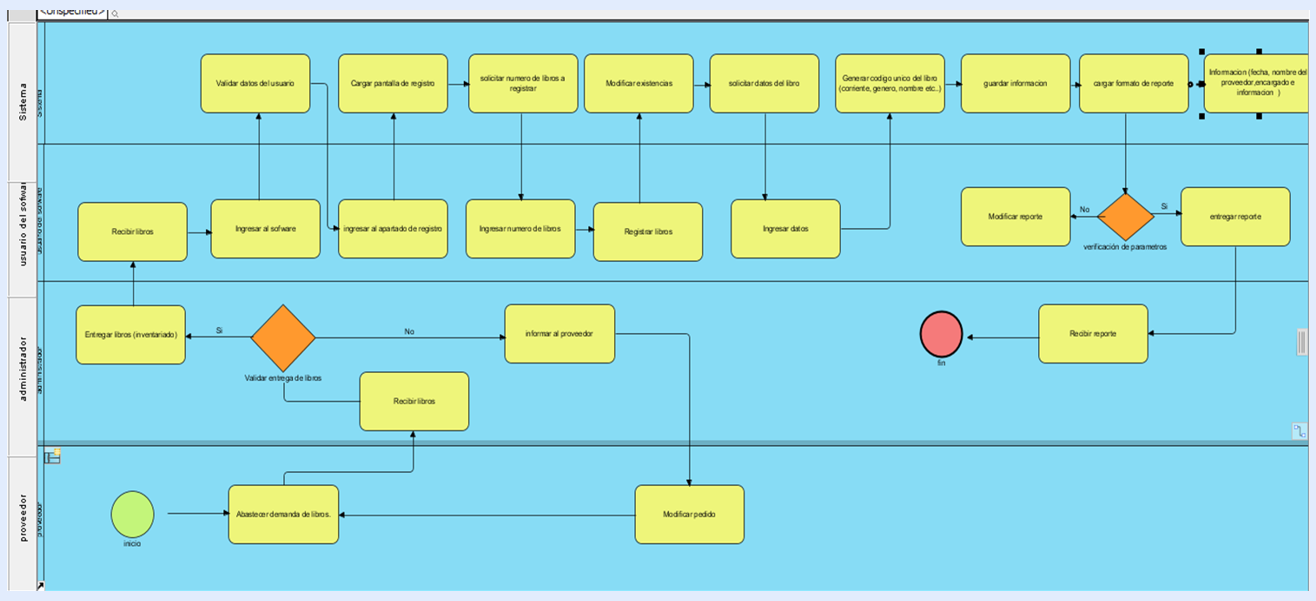
Description automatically generated

A screenshot of a cell phone

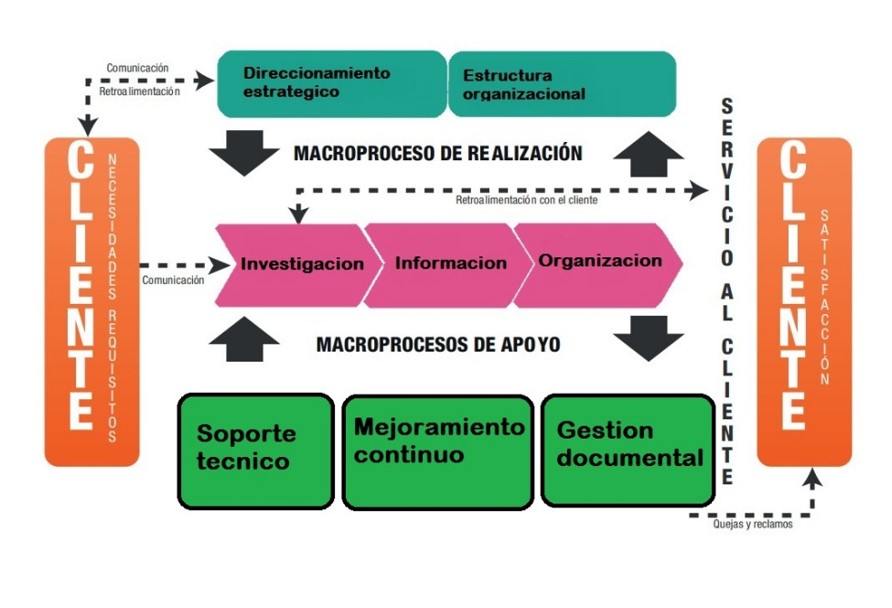
Description automatically generated

**BPMN**

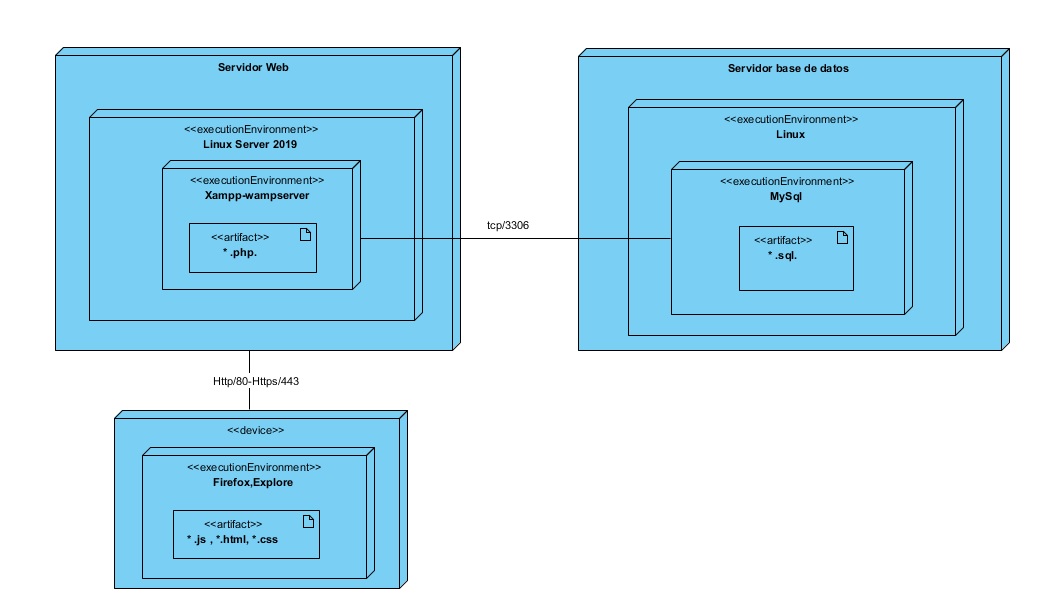
****

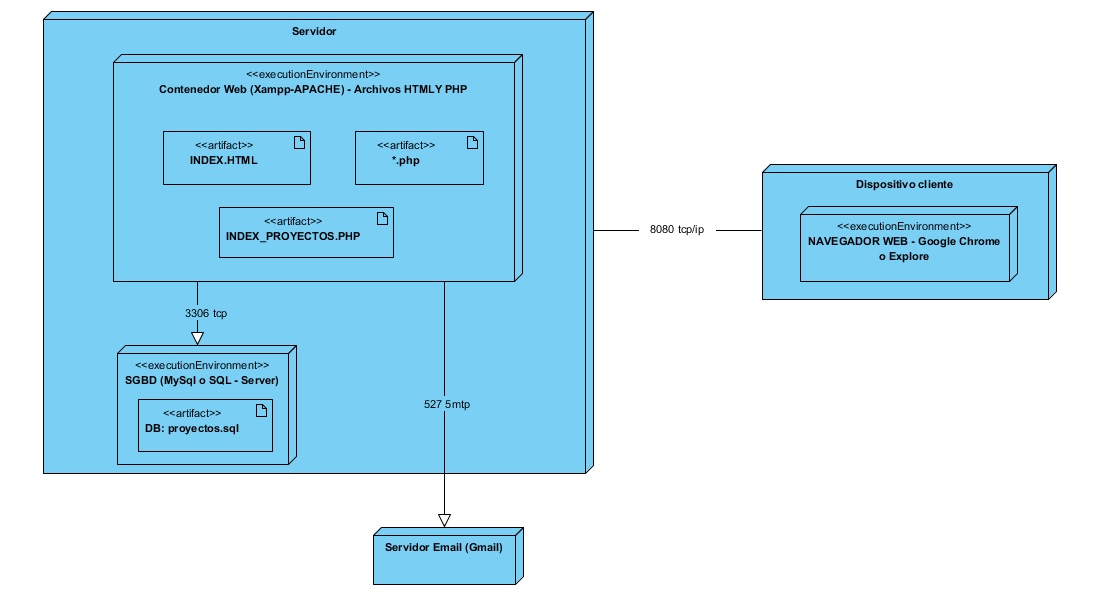
****

**MAPA DE PROCESOS**

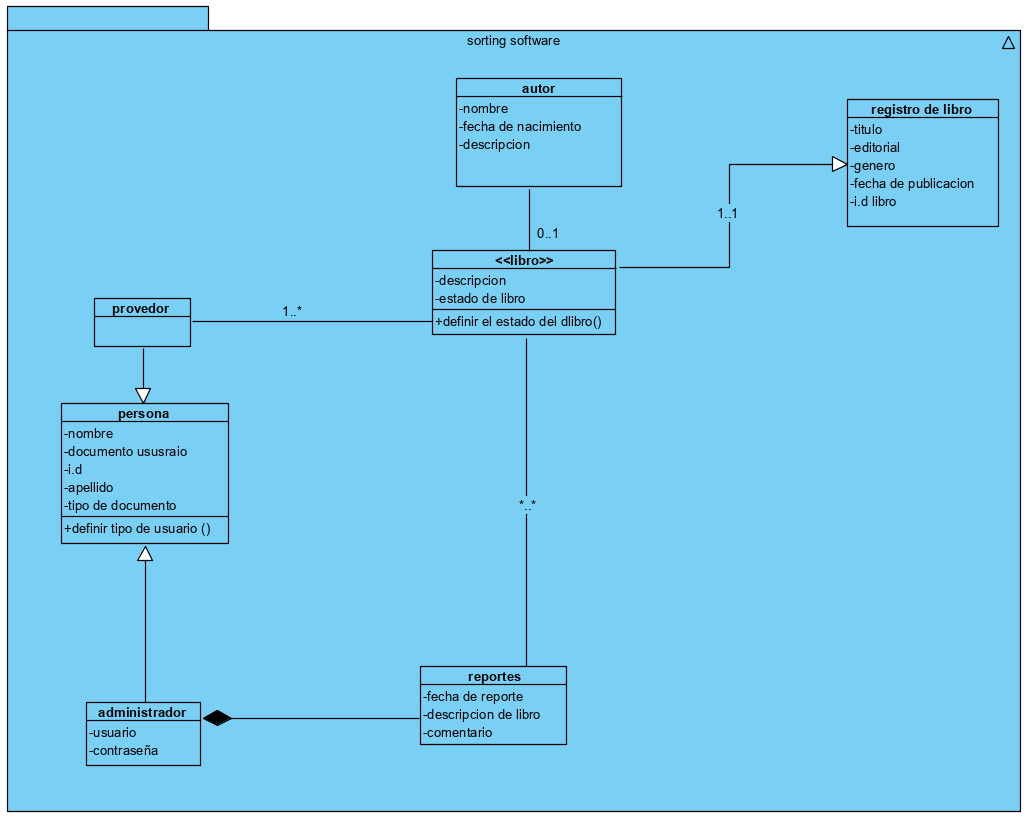
****

**DIAGRAMAS DE DESPLIEGUE**

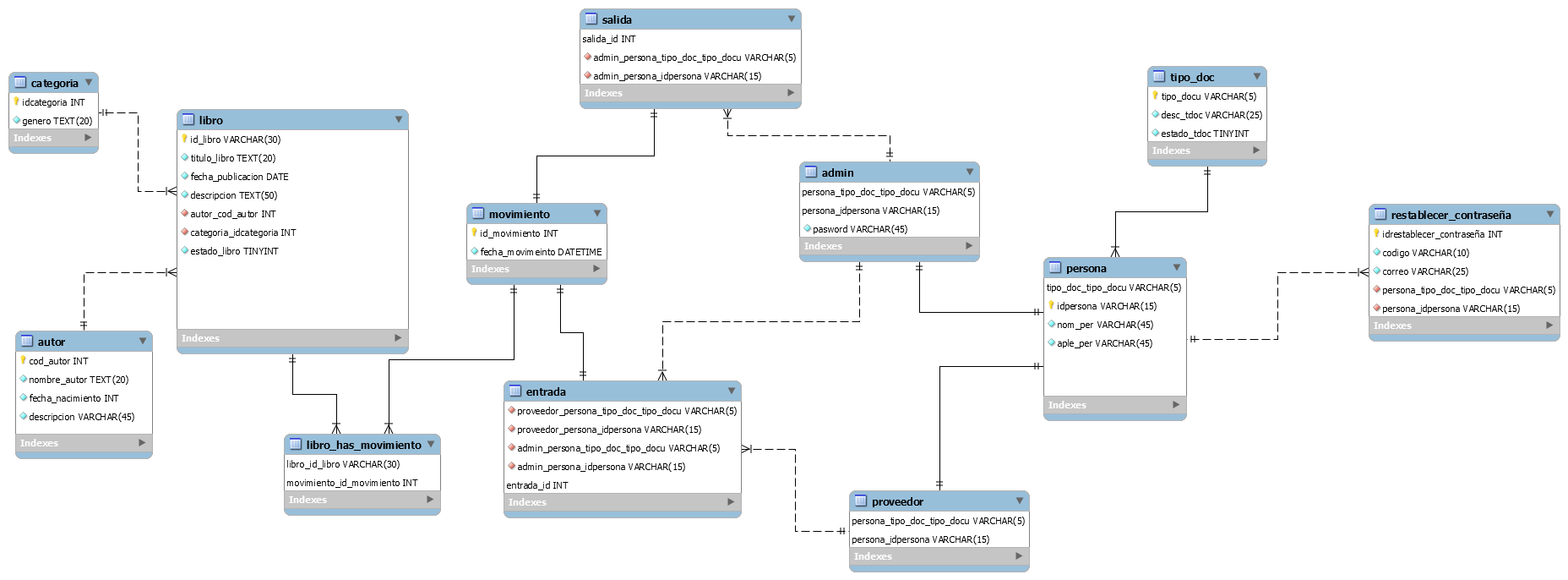
****

****

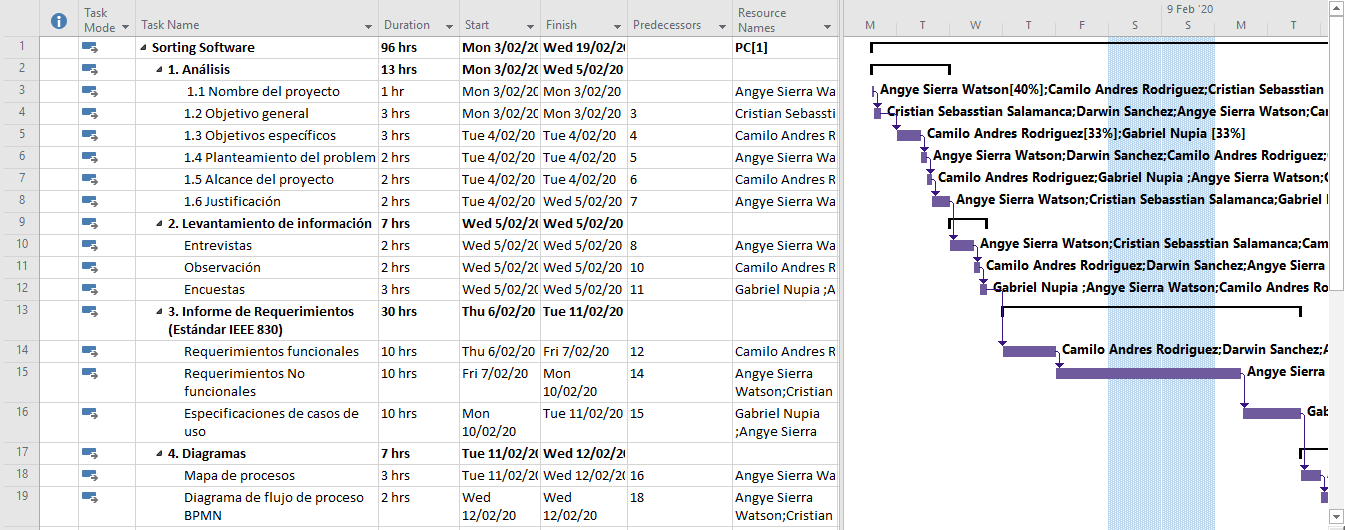
**DIAGRAMAS DE CLASES**

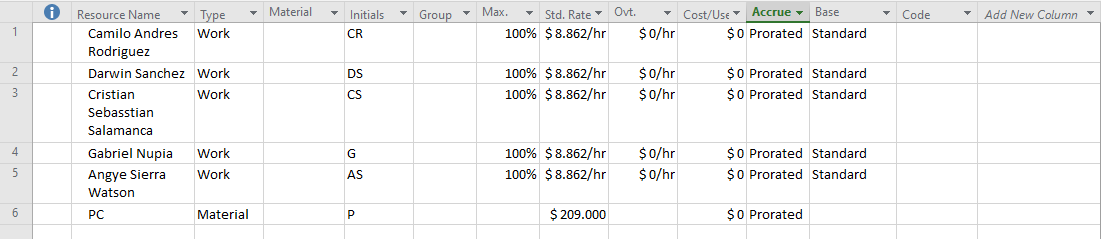
****

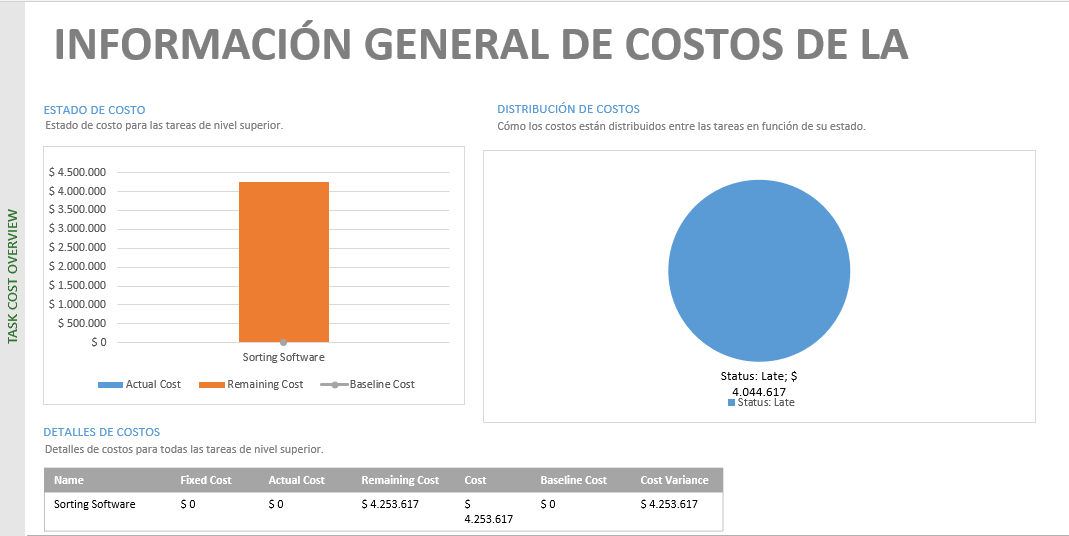
**MODELO ENTIDAD RELACIÓN**

****

**PROJECT**

****

****

****

**ELABORACION DEL SOFTWARE**

Para la elaboración de nuestro software necesitamos un hosting y servidor, donde se detallen las capacidades generales y específicas donde se alojará la aplicación WEB.

**HOSTING**

Tradicionalmente cuando se habla de Hosting se refiere al servicio que permite que un sitio web permanezca en línea los 365 días del año, sin embargo, el Hosting no solamente sirve para hospedar un sitio web, sino que sirve para hospedar cualquier tipo de información, incluido aplicaciones, sistemas de correos, archivos, bases de datos etc. En pocas palabras si se trata de confiar la información en un proveedor externo, cualquiera que ella sea, se requiere este servicio.

El hosting que vamos a utilizar es :

**HOSTGATOR COLOMBIA**

Características:

• 1 nombre para tu web Gratis

. • Sitio más que suficiente para crear un blog o una página web.

• Cetrificado de seguridad SSL totalmente gratis.

• Transferencia de datos sin límites.

• 3 cuentas de email.

• Creador de sitios web profesionales.

• Servidores SSD de alto rendimiento

• Migración gratuita

• Garantía de 30 días o la devolución de tu dinero.

• Backups semanales automáticos y totalmente gratis

**PRECIO $7.150 COP/MES\***

**SERVIDOR WEB**

El servidor web (también llamado webserver en inglés) es el software que se encarga de despachar el contenido de un sitio web al usuario. Este proceso de despacho, que a simple vista parece muy simple, es en realidad más complejo de lo que parece, pues toda la «magia» de un webserver ocurre fuera de quien está navegando por un sitio web.

Existen multitud de servidores web, y entre los más conocidos podemos encontrar por ejemplo a Apache, Nginx, LiteSpeed o IIS.

**Servidor Windows**

Intel Xeon E3-1270 v6

4 núcleos, 350 ghz

16 gigas de RAM

64 GB

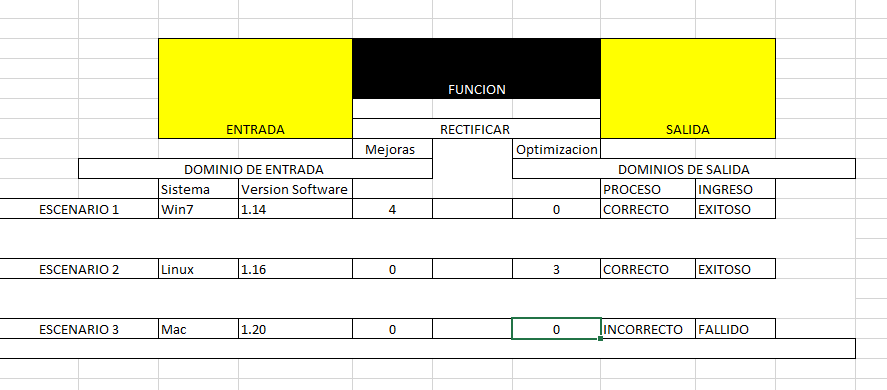
**IMPLEMENTACIÓN DE PRUEBAS**

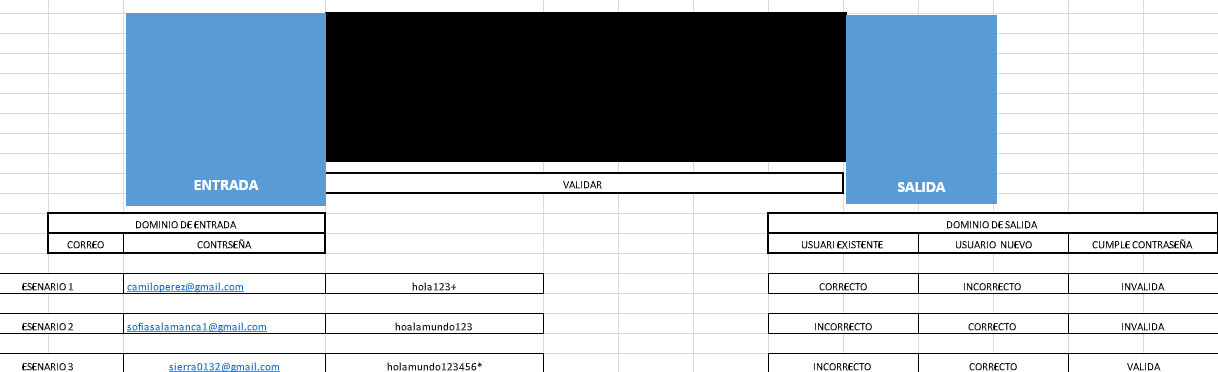
En este capítulo se harán pruebas de caja negra y caja blanca, en el caso de cajas negras se analizarán las entradas y salidas verificando que se ejecuten correctamente y tengan el resultado esperado; en el caso de caja blanca se encargara de verificar el código al momento de ejecutarse funcione correctamente sin generar ningún tipo de error.

**ESTANDARES DE PROGRAMACIÓN**

Uno de los aspectos más importantes al iniciar un proyecto informático es el desarrollo de normas y políticas que permiten la mejor comprensión de los documentos, código de programación de las bases de datos entre otros recursos. Si se trata de interpretar los bloques de código de uno o varios proyectos al momento de dar mantenimiento así directamente se perderá tiempo y al momento de realizar cambios los realizara de una manera errónea por lo cual el documento dará gran ayuda para lograr entender la codificación del sistema y administración de las entradas y salidas que se generen dentro del software.

**PRUEBAS CAJA NEGRAS DEL SOFTWARE**

****

****

**CONCLUSIÓN**

**BIBLIOGRAFIAS**

<https://blog.infranetworking.com/servidor-web/#Que_es_un_servidor_web>

<https://disenowebakus.net/javascript.php>

<https://www.cursosgis.com/caracteristicas-basicas-del-lenguaje-de-programacion-javascript-variables/>

<https://developer.mozilla.org/es/docs/Web/JavaScript>