



**INSTITUTO
TECNOLOGICO
DE CULIACÁN**

INSTITUTO TECNOLÓGICO DE CULIACÁN

INFORME TÉCNICO DE RESIDENCIA PROFESIONAL

Ing. en Sistemas Computacionales

Semestre 10°

Alumno: Jesus Alejandro Sandoval Lopez

No. de control 09170894

ASESORES

INTERNO: DR. RICARDO RAFAEL QUINTERO MEZA

EXTERNO: LIC. JUAN CARLOS GALLARDO INZUNZA

EMPRESA: Proyectos Oftalmológicos, S. A. de C. V.

Culiacán, Sinaloa a 15 de Enero de 2015

Índice

Capitulo	Pag.
1. INTRODUCCIÓN	3
2. JUSTIFICACIÓN	4
3. OBJETIVOS	5
3.1 Objetivos generales	5
3.2 Objetivos específicos	5
4. CARACTERIZACIÓN DEL ÁREA EN QUE SE PARTICIPÓ	6
5. PROBLEMAS A RESOLVER	7
6. ALCANCES Y LIMITACIONES	8
6.1 Alcances	8
6.2 Limitaciones	8
7. FUNDAMENTO TEÓRICO	9
7.1 Software Gestor de Recursos humanos	9
7.2 Arquitectura de Software	11
7.2.1 Cliente - servidor	11
7.2.2 Arquitectura en capas	12
7.3 Software de Gestión de Bases de Datos	13
7.3.1 MySQL	14
7.4 Tecnologías web usadas en este proyecto	15
7.4.1 PHP	15
7.4.2 Laravel - Framework PHP para desarrollo web	16
7.4.3 HTML5	16
7.4.4 Twitter Bootstrap	17
7.4.5 Javascript	17
7.4.6 CSS	18
8. PROCEDIMIENTOS Y DESCRIPCIÓN DE LAS ACTIVIDADES REALIZADAS	18
8.1 Análisis de requisitos	18
8.2 Requisitos funcionales	20
8.3 Requisitos de Calidad	22
8.4 Casos de Uso	22
9. RESULTADOS, PLANOS, GRÁFICAS, PROTOTIPOS Y PROGRAMAS	31
9.1 Diagrama de la base de datos del sistema	31
9.2 Diseños de la interfaz	33
9.3 Programación de la aplicación	40
9.4 Capturas de pantalla del sistema en funcionamiento	42
10. CONCLUSIONES	46
11. BIBLIOGRAFÍA	47

Índice de Imágenes

Figura	Pag.
1. Organigrama de fundación Markoptic	6
2. Estructura básica de la arquitectura cliente - servidor	12
3. Arquitectura en Capas Modelo Vista Controlador	13
4. Diagrama de casos de uso	23
5. Diagrama de la base de datos	31
6. login y registro.....	33
7. Editar CVU	33
8. Dirección	34
9. Teléfonos	34
10. Escuelas	35
11. Títulos	35
12. Cursos	36
13. Documentos	36
14. Especialidades	37
15. Idiomas	37
16. Proyectos anteriores	38
17. Publicaciones	38
18. Tecnologías	39
19. Trabajos	39
20. Ejemplo de Modelo	40
21. Ejemplo de Controlador	41
22. Ejemplo de una vista	42
23. Ventana de login	43
24. Formulario de registro	43
25. Edición de curriculum	44
26. Pantallas varias de llenado del Curriculum	44

1. Introducción

En el presente documento se realiza una descripción del proyecto **“Diseño y programación de un Sistema para el registro y manejo de Currículos de investigadores adscritos a Fundación Markoptic”** y todo su contenido que se ha recabado conforme marcha su elaboración.

Este proyecto fue elaborado por mí como estudiante del INSTITUTO TECNOLÓGICO DE CULIACÁN para hacer un sistema web que ayudará a la fundación a llevar sus funciones administrativas referentes a al manejo de Recursos Humanos de forma más sencilla.

Este sistema web facilitará el trabajo del departamento de recursos humanos que lleva un control de las personas que laboran en los distintos proyectos de la fundación y del centro de investigación, lo cual se hace de forma manual, archivando los currículos en físico y se los consulta uno a uno cada que se necesita asignar personal para los proyectos, así como también permitirá que las personas que deseen adscribirse a la fundación o al centro de investigación enviar su currículum.

Además el sistema está diseñado para poder mejorarlo o expandirlo, añadiendo nuevos módulos y de esta forma extender sus funcionalidades en un futuro, como puede ser agregar un módulo que permita el seguimiento de los distintos proyectos de la fundación y su centro de investigación.

Cualquier persona que esté adscrita a la fundación o al centro de investigación podría interactuar con el sistema, pero las opciones que tendrá disponibles en el sistema dependerá de su perfil, por ejemplo, si es administrador además de las opciones normales tendrá otras que le permitirán ver y editar los datos de los demás usuarios.

Para la creación del sistema se utilizará una arquitectura de tres capas, es un estilo de programación por el medio del cual se pretende separar la capa de presentación, la capa del dominio o negocio y la capa de persistencia de datos.

- Capa de presentación: Interfaz gráfica de usuario por medio del cual el usuario interactúa con el sistema.
- Capa del dominio: Lógica del sistema, en la cual se encuentran implementados todos los requerimientos hechos por el cliente.
- Capa de persistencia: Se encuentra implementado el soporte de datos (Base de Datos) para el sistema.

El sistema está basado en web. La parte del cliente se desarrolló en JQuery (librería de JavaScript multiplataforma), y la parte del servidor se desarrollo en PHP haciendo uso del framework Laravel y MySQL para la persistencia, siguiendo la arquitectura cliente-servidor.

2. Justificación

Fundación Markoptic, al igual que cualquier organización, cuenta con un departamento encargado de la gestión de los recursos humanos (RRHH), el cual se encarga de gestionar las personas que laboran tanto en la fundación como en el centro de investigación de la misma fundación, asigna al personal adecuado para los distintos puestos que existen en la fundación y además se encarga de asignar el personal de acuerdo a su perfil para los distintos proyectos del centro de investigación.

Llevar un control manual de todas las tareas que debe realizar el departamento de RRHH puede llegar a ser algo difícil y tedioso para el administrador. Ya que tiene que recibir y organizar los CV(curriculum vitae) de las personas que quieran formar parte de la organización, además de analizarlos para saber a qué área o proyecto asignar a esa persona, todo esto conlleva el riesgo de extravío de los documentos y conlleva una gran carga de trabajo y de tiempo.

Por estas razones se requiere la creación de un sistema que automatice el registro y administración de los CV de las personas que quieran integrarse a la organización y con esto facilitar el trabajo del administrador del departamento de RRHH, el sistema debe ser fácil de usar y debe contar con una arquitectura que permite ampliarlo y mantenerlo fácilmente, además debe ser lo más eficiente y eficaz posible.

3. Objetivos

3.1 Objetivo General:

Desarrollar un sistema de software de gestión de recursos humanos basado en web para fundación Markoptic especializado en el registro y manejo de los curriculums de los investigadores y demás personal adscrito y que desea suscribirse a fundación Markoptic y su centro de investigación.

El sistema debe ser desarrollado usando tecnologías HTML5, javascript y php, con una arquitectura que se adapte a los cambios que en un futuro puedan presentarse. Además esta arquitectura debe facilitar el mantenimiento por otros programadores que en un futuro se involucren en el proyecto.

3.2 Objetivos Específicos:

- Definir la información requerida en los curriculums.
- Creación de la base de datos donde se almacenarán los curriculums.
- Creación del sistema de registro de los curriculums.
- Creación de un modulo de busqueda para el administrador donde pueda visualizar los curriculums registrados en el sistema.
- El sistema debe poder ser accesible a traves de internet

- Diseñar el sistema tomando en cuenta la posibilidad de añadir otras características en un futuro.

4. Características del área en que participó

El proyecto de **“Diseño y programación de un Sistema para el registro y manejo de Currículos de investigadores adscritos a Fundación Markoptic”** fue solicitado al departamento de desarrollo tecnológico de Fundación Markoptic y surgió como una necesidad del departamento de Gestión y Loc. de Recursos, debido a que la fundación es de reciente creación son muy pocas las personas que forman parte de este departamento (figura 1).



Figura 1. Organigrama de fundación Markoptic.

5. Problemas a resolver

Los sistemas de software de Gestión Recursos Humanos son una de las herramientas más utilizadas por las organizaciones y empresas que cuentan con una gran cantidad de personal laboral. Pero se debe tener en cuenta que las necesidades de cada organización son distintas y que es muy común que las soluciones de software que ya existen en el mercado no se ajusten a las necesidades de las empresas y estas vean en la necesidad de crear una solución de software que cumpla con sus requerimientos.

Este es el problema que busca resolver este proyecto, la problemática que se suscita durante el manejo y registro de los curriculums de las personas que quieren formar parte de Markoptic, por lo general se entregan los curriculums en forma física, después estos se archivan para su posterior lectura y análisis, para luego asignarle a esa persona su puesto dentro de la fundación o un proyecto dentro del centro de investigación.

Algo parecido sucede cuando hay un nuevo proyecto dentro del centro de investigación y se quiere asignar personal ya adscrito a Markoptic para asignarlo a dicho proyecto, pues se tiene que hacer una revisión de todos los curriculums ya archivados y buscar los perfiles que mejor se adecuen para llevar dicho proyecto.

Llevar a cabo todas estas tareas de forma manual requiere de mucho trabajo y tiempo, es por eso que se solicitó la creación de un sistema de software que permita hacer todo esto de forma más rápida, eliminando los curriculum físicos y a su vez permitiendo buscar en ellos la información necesaria para designar a la persona adecuada a los proyectos dependiendo del perfil registrado en su currículum.

6. Alcance y Limitaciones

6.1 Alcance

Crear un sistema de software, que permita el registro de los curriculums de las personas que estén adscritas a fundación Markoptic y a su centro de investigación, así como el de las personas que quieran formar parte de esta, el sistema deberá almacenar dichos currículos y llevará un control de ellos, permitirá a los usuarios añadir, editar o borrar su información personal, además de dar la posibilidad a los administradores llevar un control completo del perfil de cada miembro del personal con el que cuentan, además permitirá de buscar la información pertinente en los distintos currículos almacenados para designar de forma correcta al personal más adecuado a los proyectos.

6.2 Limitaciones

La principal limitante del problema es que debido a que fundación Markoptic es de reciente creación, no existen bases sobre las cuales trabajar para crear el sistema, todo debería hacerse desde cero y se deberá tener en cuenta que el trabajo que se realice tendrá alguna repercusión en los distintos proyectos de software que se hagan para la fundación en el futuro, todos estos puntos deberán ser contemplados a la hora de elegir la arquitectura del sistema y al crear la estructura de la base de datos donde se almacena la información.

Otro punto que se debe tener en cuenta es también los recursos económicos con los que cuenta la fundación, ya que se tienen que decidir si se comprara un servidor o se rentará un servicio de hosting con algún proveedor para hacer el despliegue del sistema.

7. Fundamento Teórico

En fechas recientes se han vuelto indispensable el uso de las nuevas tecnologías ya que gracias a ellas podemos hacer infinidad de cosas relacionadas con la informática y en otros campos para los cuales sirven de igual manera, así como herramientas para el desarrollo, cada una de ellas atendiendo una plataforma en específico y para la cual sea necesitada.

Con este tipo de herramientas se facilita el trabajo además de mejorar el rendimiento, de ahí la idea de crear sistemas de software, con esto se busca simplificar las distintas tareas que realizan los diferentes departamentos y sobre todo mejorar su desempeño.

En este capítulo se documentará todos los términos usados en la elaboración de este proyecto de residencias, así como las bases teóricas sobre las que se sustenta el mismo. Se describen lenguajes, patrones, estilos arquitectónicos, etc.

7.1 Software Gestor de Recursos humanos

Se denomina recursos humanos (RRHH) al trabajo que aporta el conjunto de los empleados o colaboradores de una organización. Es muy frecuente llamar así al sistema o proceso de gestión que se ocupa de seleccionar, contratar, formar, emplear y retener al personal de la organización.

Los sistemas de administración de recursos humanos o sistemas de información de recursos humanos, forman una interfaz entre la gestión de recursos humanos y la tecnología de información. Esto combina los recursos humanos y en particular la actividad de administrarlos con los medios puestos a su disposición por la informática, y se refieren en particular a las actividades de planificación y tratamiento de datos para integrarlos en un único sistema de gestión.

Los sistemas de gestión de recursos humanos pueden tener distintos enfoques facilitan el manejo de dichos recursos, algunos ejemplos de estos enfoques son:

- Nómina:

Este enfoque se utiliza para automatizar la gestión del sueldo reuniendo los datos del trabajador sobre su trabajo, asistencia, calculando las deducciones e impuestos, la cotización social ha generando periódicamente la orden de pago.

- Gestión de las prestaciones de trabajo

Un sistema de gestión de recursos humanos enfocado a las prestaciones de trabajo evalúa la información de tiempo/trabajo de cada empleado. Al analizar estos datos se puede distribuir de mejor forma el trabajo.

- Gestión de recursos y humanos

Los sistemas enfocados a la gestión de recursos y humanos incluyen menos particularidades legales y están más enfocados a gestionar los recursos disponibles. Registra datos como dirección, selección, formación y desarrollo, capacidades, dirección de habilidades y otras actividades relacionadas. También se administran los datos personales (edad, dirección, familia...), las competencias y títulos, las formaciones seguidas, los niveles de salario, el registro de los datos del currículum vitae entre otros.

- Productividad

Los sistemas enfocados a evaluar la productividad de cada empleado, llevan una cuantificación del tiempo y las actividades que generan resultados positivos dentro de la organización. Se miden las actividades de cada uno de los empleados y se separan las actividades "Productivas" (Aquellas que están directamente relacionadas con la empresa) de las "Personales" (Aquellas que no tienen relación directa con la empresa). De esta manera se cuantifica el costo "real" (horas hombre) que le toma a cada uno de ellos realizar sus labores, así como las herramientas que utilizan para alcanzar los objetivos.

Además de poder tener algún enfoque específico, también existen sistemas que pueden tener más de un solo enfoque, separan cada parte en módulos y cada módulo se encarga de manejar un enfoque distinto de la gestión de RRHH.

7.2 Arquitectura de Software

La arquitectura de software es, a grandes rasgos una vista del sistema que incluye los componentes principales del mismo, como se comportan esos componentes según los percibe el resto del sistema, y la forma en que los componentes interactúan y se coordinan para alcanzar la misión del sistema (Pressman, 2002).

Este punto es uno de los más importantes al construir una aplicación, pues de ello depende su mantenibilidad y flexibilidad al crecimiento.

La arquitectura elegida para el sistema es una mezcla entre la arquitectura de Cliente - Servidor y la arquitectura en Capas.

7.2.1 Cliente - servidor

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema, esta es una parte muy importante dentro de nuestro sistema, ya que será el servidor donde se mantendrá la información referente a los currículos, manteniéndola disponible para que los “clientes” puedan tener acceso a ella.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa (figura 2).

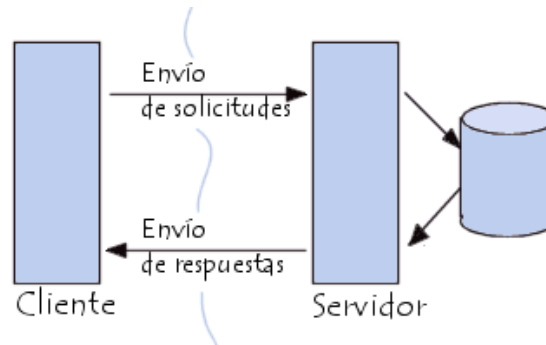


Figura 2. Estructura básica de la arquitectura cliente - servidor

7.2.2 Arquitectura en capas

Garlan y Shaw definen el estilo en capas como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y recibe servicios que le brinda la inferior.

En términos más simples, se dividen las funcionalidades y responsabilidades del sistema en distintas capas y cada una se encarga de atender una responsabilidad específica comunicándose solo con su capa adyacente, pero sin conocer más allá de las capas superior e inferior.

El estilo en capas es uno de los más comunes y existen una gran cantidad de variantes de esta arquitectura, pero la más común es la que separa el sistema en una capa de presentación, una capa de reglas de negocios y una capa de servicios de datos, se le conoce como patrón Modelo Vista Controlador.

En el patrón Modelo-Vista-Controlador(MVC) se separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes (Reynoso, 2004):

- **Modelo:** En él se encapsulan las reglas de negocio, el comportamiento de los datos y la manera en que las entidades se encargan de suplir los requerimientos de la aplicación.
- **Vista:** Es la interfaz en la que el usuario final puede ver representados los datos de la aplicación.
- **Controlador:** Es el intermediario entre el usuario y la aplicación, y entre el modelo y la vista. Es el controlador el encargado de servirle la vista al usuario y de comunicar la vista con el modelo.

Esta arquitectura se está volviendo muy popular en los desarrollos web. Actualmente muchos de los frameworks de programación web están basados en esta arquitectura (Figura 3).

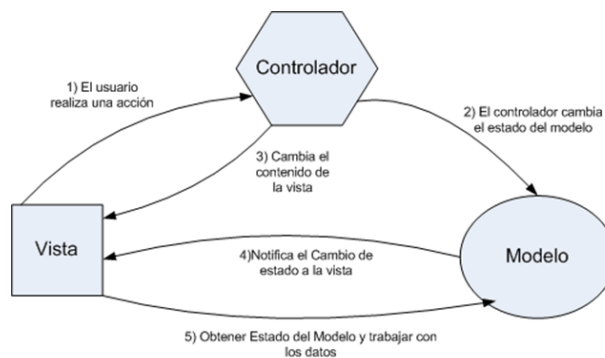


Figura 3. Arquitectura en Capas Modelo Vista Controlador

7.3 Software de Gestión de Bases de Datos

Los softwares de gestión de bases de datos (en inglés database management system, abreviado DBMS) son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos, el usuario y las aplicaciones que la utilizan. El propósito general de los sistemas de gestión de bases de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización.

7.3.1 MySQL

Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

MySQL surgió alrededor de la década del 90, Michael Windenis comenzó a usar mSQL para conectar tablas usando sus propias rutinas de bajo nivel (ISAM). Tras unas primeras pruebas, llegó a la conclusión de que mSQL no era lo bastante flexible ni rápido para lo que necesitaba, por lo que tuvo que desarrollar nuevas funciones. Esto resultó en una interfaz SQL a su base de datos, totalmente compatible a mSQL.

Las principales características de MySQL son:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

7.4 Tecnologías web usadas en este proyecto

Se quiere que la aplicación web utilice las tecnologías en desarrollo web más novedosas y funcione de forma similar sin importar desde que dispositivo se visualice el contenido, para poder cumplir con estos objetivos se hará uso de las tecnologías para desarrollo web que se listan a continuación.

7.4.1 PHP

PHP fue creado en 1994 por Rasmus Lerdorff para tener un conteo de cuantas personas visitaban su página. Fue creciendo hasta llegar a ser utilizado en situaciones más profesionales.

PHP es un lenguaje interpretado embebido en HTML lo que significa que puedes escribir código PHP dentro de etiquetas HTML. Por ser un lenguaje interpretado significa que no genera un código objeto, sino que se va interpretando en tiempo de ejecución en instrucciones legibles para la computadora.

Otra característica importante es que es un lenguaje del lado del servidor. Para ejecutarlo es necesario contar con un servidor web y hacerle peticiones al mismo. El servidor por excelencia de PHP es el servidor Apache. Sin embargo también es posible ejecutarlo en otros servidores tales como IIS de Microsoft.

Que el procesamiento se haga del lado del servidor reduce la carga de la máquina cliente y además proporciona mecanismos más seguros para realizar operaciones delicadas.

Con todas sus características, PHP se convierte en un lenguaje de programación web que permite crear sitios dinámicos y realizar procesos complejos de las reglas de negocios. Su sintaxis es muy simple pero a la vez es un lenguaje poderoso ideal para este proyecto de residencias(Ullman, 2007).

7.4.2 Laravel - Framework PHP para desarrollo web

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

Laravel tiene como objetivo ser un framework que permita el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Intenta aprovechar lo mejor de otros frameworks y aprovechar las características de las últimas versiones de PHP (Dayle Rees , 2013).

7.4.3 HTML5

HTML es el acrónimo en inglés de HyperText Markup Language (en español se traduce como lenguaje de marcado de hipertexto). HTML 5 es el nombre que se usa para referirse a la quinta revisión del lenguaje HTML.

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas <div> y , pero tienen un significado semántico, como por ejemplo <nav> (bloque de navegación del sitio web) y <footer>. Otros elementos proporcionan nuevas funcionalidades a través de una interfaz estandarizada, como los elementos <audio> y <video>. Mejora el elemento <canvas>, capaz de renderizar elementos 3D en los navegadores más importantes (Firefox, Chrome, Opera, Safari e Internet Explorer).

7.4.4 Twitter Bootstrap

Twitter Bootstrap es un framework o conjunto de herramientas de software libre para diseño de sitios y aplicaciones web de forma responsiva. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

El framework proporciona clases css y código javascript para definir el layout de la página, crear componentes que respondan a eventos y estilizar los elementos html más habituales.

Los principios en los que se basa el framework son:

- **Responsive Design:** Consiste en que la página trata de “hacer lo correcto” al ser visualizada independientemente del dispositivo y tamaño de la pantalla.
- **Cross Browser:** Trata de ser compatible con la mayoría de navegadores.

7.4.5 Javascript

JavaScript es un lenguaje de programación utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página web. Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario. JavaScript es un lenguaje con muchas posibilidades, es orientado a objetos, con funciones, estructuras de datos complejas, etc (Gandy, 2005).

JavaScript no permite dos de las características clásicas de los lenguajes orientados a objetos (herencia y polimorfismo), pero permite la creación y manipulación de objetos sencillos, y la definición de métodos y propiedades para dichos objetos.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas

escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems.

7.4.6 CSS

Las hojas de estilo en cascada (CSS por sus siglas en inglés) hacen referencia a un lenguaje de hojas de estilos usado para describir la presentación semántica (el aspecto y formato) de un documento escrito en lenguaje de marcas. Su aplicación más común es dar estilo a páginas webs escritas en lenguaje HTML y XHTML (García, 2009).

CSS, es una tecnología que nos permite crear páginas web de una manera más creativa, gracias a este tenemos más control sobre el aspecto final de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores, efectos, animaciones, etc.

8. Procedimientos y descripción de las actividades realizadas

Las actividades realizadas durante el periodo de residencias profesionales van desde el comienzo del ciclo de vida del software iniciando con el análisis de los requisitos, hasta el diseño e implementación del mismo. Estas actividades pueden clasificarse en el análisis, diseño, implementación y pruebas del software.

8.1 Análisis de requisitos

“Sistema Integral para Administración CV”

El proyecto consiste en diseñar, evaluar y transformar el registro de usuarios del sistema integral para administración de CV (estudiantes, Doctores etc.) El sistema ayudará al administrador de CV a llevar un control más preciso y organizado a cerca de las

especialidades y ramas de cada registro de CV así como mantener la información actualizada.

Los requerimientos de usuario son los siguientes:

1. Existe una persona que se encarga de administrar los CV.
2. Cualquier persona que se presente a laborar en Markoptic (fundación o centro de investigación) deberá registrar su CV.
3. El administrador puede conocer la información que viene en el CV de cada participante.
4. El administrador solo requiere que los que registren su CV, registre solo lo que se le pide en la pantalla de la página de la empresa.
5. Existen diferentes tipos de CV (Licenciatura ya sea terminada o en proceso de prácticas profesionales/servicio social, maestría, doctorado).
6. El administrador verifica el registro de CV todos los días.
7. El administrador requiere de retroalimentación para los que registren su CV.
8. El registro y administración de los CV debe ser en línea.
9. Los CV deben almacenarse en el sistema.
10. El administrador requiere de clasificación de los CV por áreas y especialidades (por ejemplo medicina, física, electrónica etc.)
11. Los usuarios del sistema deberán identificarse para tener acceso al mismo.
12. El administrador debe poder editar o eliminar la información de los usuarios
13. La información que los CV deben contener es la siguiente:
 - Nombre completo
 - Edad
 - Fecha de nacimiento
 - Domicilio
 - RFC
 - Sexo
 - Teléfono
 - Nombre de la preparatoria en la que estudió

- Nombre de la universidad en la que estudia o estudió
- Su licenciatura si cuenta con una
- Maestría (si cuenta con una)
- Doctorado (si cuenta con uno)
- Líneas de investigación o de aplicación del conocimiento
- Mencione los cursos o capacitaciones adquiridos
- Últimos 3 Trabajos anteriores (Nombre del trabajo, puesto que desempeñó, tiempo que duró trabajando ahí, o si aún trabaja ahí, jefe directo o ultimo jefe)
- ¿Ha trabajado en algún proyecto? ¿Nombre del proyecto y una breve descripción?
- ¿Ha publicado alguna tesis, artículo, Libro, Etc.?
- Mencione los nombres de dichos trabajos y una breve descripción
- Idiomas que domina, porcentaje certificado EJEMPLO INGLÉS: (toefl)
- Equipamiento o maquinaria científica y/o de desarrollo tecnológico que domina
- Software, programas que domina
- Disponibilidad de horarios
- ¿Cómo se enteró de la empresa?
- ¿Esta dispuesto a viajar?
- Adjuntar archivos físicos en pdf (por ejemplo rfc, acta de nacimiento, etc.)

8.2 Requisitos funcionales

Después de analizar los requisitos especificados por el usuario se realizó un proceso de modificación, filtrado y reacomodo de los requisitos y de este modo obtener los requisitos funcionales del sistema a desarrollar.

1. El sistema debe permitir a cualquier persona registrarte.

2. El sistema debe contar con distintos perfiles (administrador, residente, investigador, prestadores de servicio social, voluntarios, etc.)
3. El sistema debe permitir al administrador agregar o eliminar perfiles.
4. El sistema debe permitir a los usuarios registrados llenar un CV (currículo vitae) con los siguientes campos:
 - Nombre completo
 - Edad
 - Sexo
 - Fecha de nacimiento
 - Domicilio
 - Teléfonos
 - RFC
 - CURP
 - Nombre de las escuelas donde estudia o a estudiado (desde preparatoria)
 - Liste los títulos con los que cuenta
 - Mencione sus Líneas de investigación, de aplicación del conocimiento y especialidades
 - Mencione los cursos o capacitaciones adquiridos
 - Últimos 3 Trabajos anteriores (Nombre del trabajo, puesto que desempeñó, tiempo que duró trabajando ahí, o si aún trabaja ahí, jefe directo o ultimo jefe)
 - ¿Nombre de los proyectos en los que ha participado y una breve descripción?
 - Mencione los nombres de los trabajos que ha publicado y una breve descripción de ellos
 - Idiomas que domina, porcentaje certificado EJEMPLO INGLES: (toefl)
 - Nombre el Software, equipamiento o maquinaria científica y/o de desarrollo tecnológico que domina
 - Disponibilidad de horarios

- Esta dispuesto a viajar
5. El sistema debe requerir a los usuarios identificarse con usuario y contraseña para poder acceder a este y poder realizar las tareas correspondientes a su perfil.
 6. El sistema debe permitir a cualquiera que esté registrado llenar su CV.
 7. El sistema debe mostrar al administrador los nuevos usuarios registrados.
 8. El sistema debe permitir al administrador poder ver la información de los usuario registrados.
 9. El sistema debe permitir filtrar los CV al administrador de acuerdo a diversos criterios de búsqueda (por perfil, especialidad, por área de aplicación de conocimiento, por carrera, por nombre, etc.)
 10. El sistema debe permitir al administrador poder indicarle a los demás usuarios cambios en la información que registraron para mejorar su CV.
 11. El sistema debe permitir al usuario editar su información personal.
 12. El sistema debe permitir al administrador editar la información de cualquiera que esté registrado.
 13. El sistema debe indicar al usuario el nivel de completitud del registro de su CV.
 14. El sistema debe permitir al usuario adjuntar distintos archivos en pdf a su CVU.

8.3 Requisitos de Calidad

1. El sistema debe asegurar la integridad de la información
2. El sistema debe estar en línea las 24 horas
3. El sistema debe ser fácil de mantener
4. El sistema debe ser fácil de extender y mejorar
5. El sistema debe contar con una interfaz amigable
6. El tiempo de respuesta del sistema deberá ser corto

8.4 Casos de Uso

Un caso de uso es una descripción de algún proceso detallando cada uno de los pasos o las actividades que deberán realizarse para llevar a cabo dicho proceso de forma satisfactoria.

Son una herramienta muy útil para expresar en detalle los pasos a seguir para cumplir con un requerimiento.

Los actores involucrados y su semántica se describen a continuación:

Administrador: Es el encargado de administrar de administrar los curriculums que están en el sistema, puede ver el contenido de cualquier curriculum, buscar información en la base de datos de curriculums y editar la información de estos .

Personal: Son toda aquella persona que quiera formar parte de fundación Markoptic y de su centro de investigación, pueden tener distintos perfiles (investigador, residente, voluntario, servicio social, etc.) y puede crear un currículum dentro del sistema, a su vez puede editar o borrar información de su propio curriculum.

A continuación se muestra el diagrama de casos de usos identificados en el análisis de requerimientos (Figura 4)

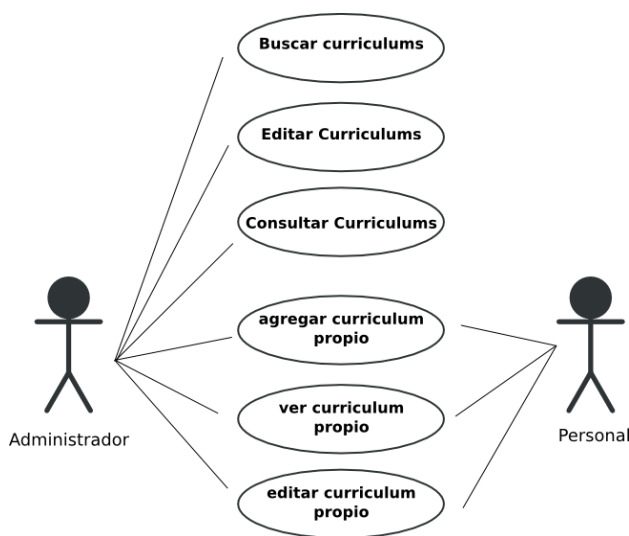


Figura 4. Diagrama de casos de uso

Caso de uso UC01: Agregar Currículum propio

Nombre del caso de uso: Agregar Currículum propio

Actores Principales: Administrador y Personal

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: Se agrega la información del currículum

Flujo Básico:

1. El usuario selecciona la opción de editar información del currículum
2. Se muestra una serie de opciones donde se indica la información a rellenar.
3. El usuario elige una opción para rellenar y se muestra el formulario donde ingresara la información referente a la opción seleccionada.
4. El usuario llena la información en un formulario.
5. El usuario guarda la información del formulario.
6. El usuario regresa a las opciones de edición del curriculum y elige la siguiente opción que desea completar.
7. El usuario continúa repitiendo los pasos anteriores hasta que haya terminado de llenar toda la información requerida del curriculum

Flujo alternativo

1. Datos no válidos

El sistema informa del error y vuelve a solicitar los datos

El usuario vuelve a introducir los datos solicitados

3. Formulario no completado

El usuario puede dejar algún formulario sin llenar completamente y volver más tarde a llenarlo después de guardarlo

4. El usuario cancela la operación

El sistema aborta el proceso de llenado de la información

Caso de uso UC02: Ver Currículum propio

Nombre del caso de uso: Ver Currículum propio

Actores Principales: Administrador y Personal

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: Ninguna

Flujo Básico:

1. El usuario accede a la visualización de la información de su curriculum.
2. El sistema muestra muestra la información disponible del curriculum del usuario.

Flujo alternativo

4. El usuario cancela la operación

El sistema aborta el proceso de llenado de la información

Caso de uso UC03: Editar Currículum propio

Nombre del caso de uso: Editar Currículum propio

Actores Principales: Administrador y Personal

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: Se edita la información ya almacenada del currículum

Flujo Básico:

1. El usuario ingresa a la seccion de edicion de información del currículum
2. Se muestra una serie de opciones donde le indica la información a rellenar.
3. El usuario selecciona que opción quiere editar y se muestra la información ya guardada referente a esa opción.
4. El usuario selecciona editar información.
5. El sistema muestra un formulario con la información almacenada de la opción elegida.
6. El usuario edita la información que desee.
7. El usuario guarda los cambios hechos a la información.

Flujo alternativo

1. Datos no válidos

El sistema informa del error

El usuario vuelve a modificar los datos solicitados.

2. Formulario no completado

El usuario puede dejar algún formulario sin llenar completamente y volver más tarde a llenarlo después de guardarlo

3. El usuario cancela la operación

El sistema aborta el proceso de edición de la información

Caso de uso UC04: Ver Currículums

Nombre del caso de uso: Ver Currículums

Actores Principales: Administrador

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: Ninguna

Flujo Básico:

1. El administrador ingresa a la sección de visualización de los curriculums.
2. El administrador selecciona el curriculum que desea ver.
3. El sistema muestra toda la información disponible del curriculum seleccionado.
4. Para finalizar el usuario regresa la sección de visualización de los curriculums.

Flujo alternativo

1. El usuario cancela la operación

El sistema aborta el proceso de llenado de la información.

Caso de uso UC05: Buscar Currículums

Nombre del caso de uso: Buscar Currículums

Actores Principales: Administrador

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: Ninguna

Flujo Básico:

1. El administrador accede a la sección de visualización de los curriculums.
2. El administrador ingresa el término a buscar en los curriculums.
3. El sistema muestra todos los curriculums disponibles que contengan el término de búsqueda.
4. El administrador podrá seleccionar alguno de los curriculums mostrados por el resultado de la búsqueda para poder ver toda la información disponible de ese curriculum.

Flujo alternativo

1. El término de búsqueda no se encuentra

El sistema informa que el termino de busqueda no existe en ningún curriculum y le indica al usuario que ingrese uno distinto.

2. El usuario cancela la operación

El sistema aborta el proceso de busqueda de curriculums.

Caso de uso UC06: Editar Currículums

Nombre del caso de uso: Editar Currículums

Actores Principales: Administrador

Stake holders: Ninguno

Precondiciones:

Estar autenticado en el sistema y tener permisos para ejecutar este caso de uso

Postcondiciones: El sistema guarda los cambios hechos al curriculum seleccionado

Flujo Básico:

1. El administrador ingresa a la sección de visualización de los curriculums.
2. El administrador selecciona el curriculum que desea editar.
3. El sistema muestra las opciones de llenado del curriculum seleccionado.
4. El administrador selecciona la opción donde se encuentra la información a editar y selecciona la información que desea modificar.
5. El sistema muestra el formulario de edición con la información seleccionada .

6. EL administrador realiza los cambios a la información.

7. El usuario guarda los cambios realizados

Flujo alternativo

1. Datos no válidos

El sistema informa del error

El administrador vuelve a modificar los datos deseados.

2. Formulario no completado

El administrador puede dejar algún formulario sin llenar completamente y volver mas tarde a llenarlo después de guardarlo

1. El usuario cancela la operación

El sistema aborta el proceso de edición de curriculums.

9. Resultados, Planos, Gráficas, Prototipos y Programas

9.1 Diagrama de la base de datos del sistema

A continuación se muestra el diagrama de la estructura de la base de datos(figura 5) y se hace una descripción de las relaciones y el funcionamiento de cada uno de los componentes que la conforman.

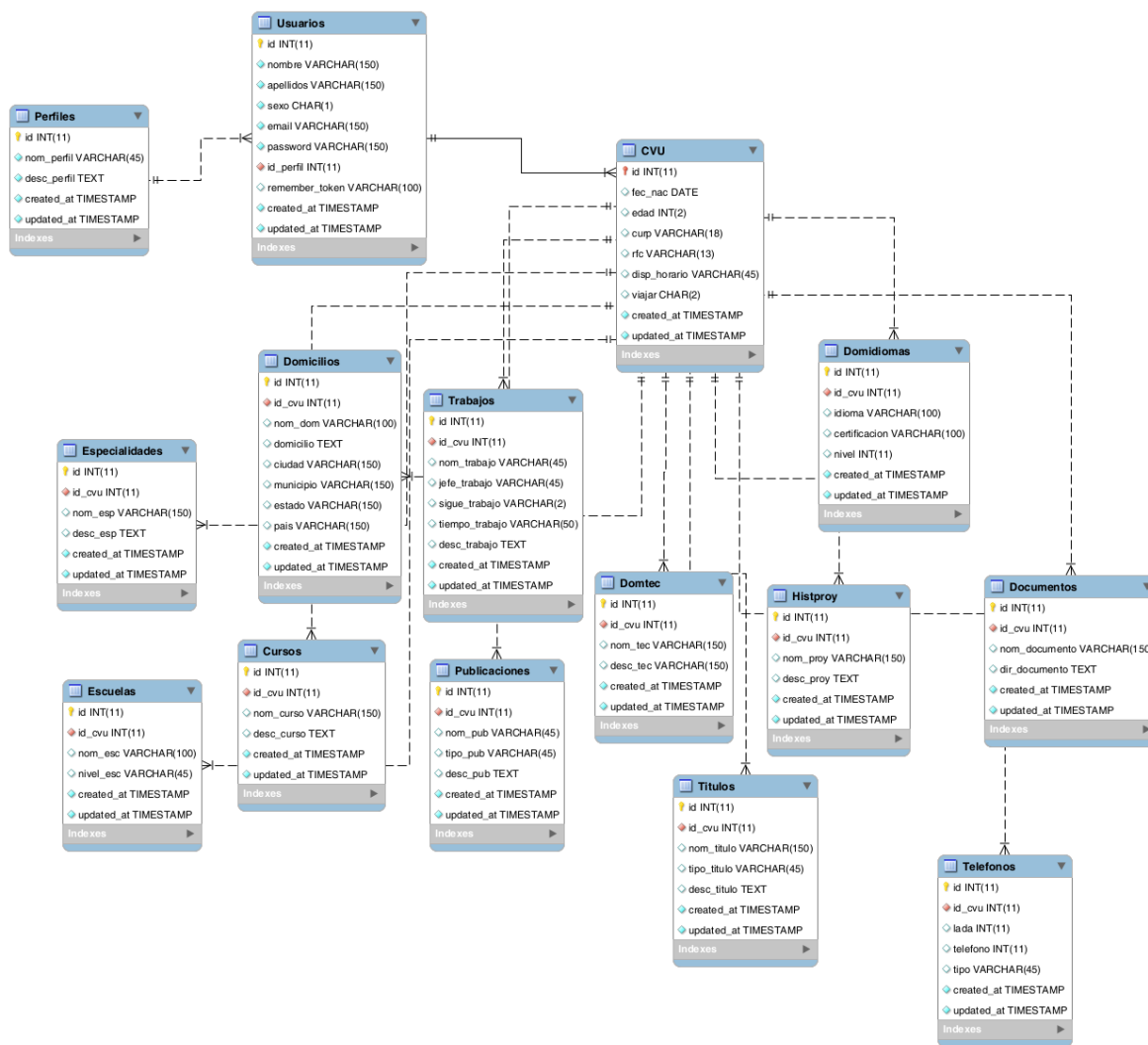


Figura 5. Diagrama de la base de datos

En el diagrama de la base de datos del sistema se puede ver las distintas tablas que componen la estructura de la base de datos, existen 3 tablas principales:

- **Perfiles:** Es donde se almacenan los distintos perfiles a los que puede pertenecer un usuario del sistema, y tiene una relación de 1 a muchos con la tabla de usuarios (un usuario puede tener solo un perfil, pero un perfil puede tener varios usuarios).
- **Usuarios:** En esta tabla se almacena la información básica del usuario (nombre, sexo, etc.), así como la información referente al inicio de sesión en el sistema, tiene una relación de muchos a uno con la tabla perfiles (muchos usuarios pueden tener un mismo perfil) y una relación de uno a uno con la tabla CVU (un usuario solo puede tener un CVU).
- **CVU (currículum vitae universal):** En esta tabla se almacena toda la información básica del currículum del usuario además de que es la columna principal a la hora de formar el curriculum pues tiene varias relaciones con distintas tablas dentro de la base de datos y que juntas conforman el currículum completo del usuario, tiene una relación de uno a uno con la tabla usuarios (un cvu solo puede tener un usuario y viceversa), y tiene una relación de uno a muchos con las distintas tablas que terminan de componer el curriculum (domicilios, teléfonos, títulos, escuelas, etc.), esto es para permitir al usuario tener más de un domicilio, teléfono, títulos, escuelas, etc.

9.2 Diseños de la interfaz

A continuación se muestran una serie de imágenes donde se muestran los diseños que se planearon para la interfaz del sistema de gestión de curriculums, se busca que mediante un buen diseño de la interfaz la aplicación se fácil de entender y de usar.

Iniciar Sesión **Registrarme**

Iniciar sesion

[Olvide mi contraseña](#)

Integrate con Nosotros

Registrarme

Figura 6. login y registro

Juan Perez
Administrador

CVU ▼

Visualisar

Editar

Mensajes

Busqueda

Editar CVU

Selecciona la opción a editar de tu CVU

 Informacion personal <small>Agrega o edita la informacion personal de tu CVU</small>	 Domicilios <small>Agrega o edita un domicilio donde contactarte, puedes agregar varios en caso de tener mas de uno.</small>	 Escuelas <small>Agregue o edite las escuelas donde a estudiado</small>	 Titulos <small>Agregue o edite los titulos que tiene acreditados</small>
 Especialidades, Areas de investigacion y de aplicacion del conocimiento <small>Mencione sus especialidades, areas de investigacion y de aplicacion de conocimiento y una breve descripcion</small>	 Cursos y Capacitaciones <small>Mencione los cursos y capacitaciones que a tomado y una breve descripcion de ellos</small>	 Publicaciones <small>Mencione las publicaciones que tenga junto con el tipo y una breve descripcion</small>	 Tecnologias que domina <small>Mencione las tecnologias que domina (equipo de investigacion, software, dispositivos, etc.) y una breve descripcion de cada uno</small>
 Idiomas <small>Mencione los idiomas que domina y su nivel certificado</small>	 Proyectos Anteriores <small>Mencione los distintos proyectos en los que ha trabajado anteriormente</small>	 Trabajos <small>Mencione los trabajos en los que labora o laboro anteriormente</small>	 Adjuntar Archivos <small>Aqui podria subir y gestionar los documentos que se le soliciten</small>

El estado de tu CVU es: 20% completado

Figura 7. Editar CVU




<p> Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Domicilios en esta seccion puedes agregar, editar o eliminar sus domicilios</p> <p>3  Salir</p>
	<p>Domicilio <input type="text" value="Av. Fulano de tal Calle del olvido No. 123"/></p> <p>Ciudad <input type="text"/></p> <p>Municipio <input type="text"/></p> <p>Estado <input type="text"/></p> <p>Pais <input type="text"/></p> <p> <input type="button" value="Guardar"/> <input type="button" value="Eliminar"/> </p> <p></p>
<p>El estado de tu CVU es: <input type="text" value=" "/> 20% completado</p>	

Figura 8.dirección




<p> Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Telefonos en esta seccion puedes agregar, editar o eliminar sus numeros telefonicos de contacto</p> <p>3  Salir</p>
	<p>Lada <input type="text"/> Telefono <input type="text"/> Tipo <input type="text" value="Celular"/></p> <p> <input type="button" value="Guardar"/> <input type="button" value="Eliminar"/> </p> <p></p>
<p>El estado de tu CVU es: <input type="text" value=" "/> 20% completado</p>	

Figura 9. Teléfonos

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<div>Logo</div> <div> Escuelas en esta seccion puedes agregar, editar o eliminar las escuelas donde a estudiado </div> <div>3</div> <div> Salir </div>
	<div> Nombre de la escuela <input type="text"/> </div> <div> <input type="button" value="Guardar"/> <input type="button" value="Eliminar"/> </div> <div>+</div>
El estado de tu CVU es: <div><div></div></div> 20% completado	

Figura 10.Escuelas

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<div>Logo</div> <div> Títulos en esta seccion puedes agregar, editar o eliminar los titulos con los que cuente </div> <div>3</div> <div> Salir </div>
	<div> Tipo <input type="text" value="Licenciatura"/> Titulo <input type="text"/> </div> <div> Descripcion <input type="text"/> </div> <div> <input type="button" value="Guardar"/> <input type="button" value="Eliminar"/> </div> <div>+</div>
El estado de tu CVU es: <div><div></div></div> 20% completado	

Figura 11.Títulos

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Cursos en esta seccion puedes agregar, editar o eliminar infoamcion sobre los cursos que a recibido</p>		<p>3</p>	<p>Salir</p>
	<p>Curso <input type="text"/></p> <p>Descripcion <input type="text"/></p> <p> <input type="button" value="Guardar"/> <input type="button" value="Eliminar"/> </p> <p style="text-align: center;">+</p>			
<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>				

Figura 12.Cursos


<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Documentos en esta seccion puedes subir o eliminar los diferentes documentos que se le soliciten</p>		<p>3</p>	<p>Salir</p>
	<div style="display: flex; align-items: center;"> <div style="border: 2px dashed black; padding: 10px; text-align: center; margin-right: 20px;"> <p>+</p> <p>Agregar un archivo</p> </div> <div style="text-align: center;"> <p></p> <p>acta de nacimiento.pdf</p> </div> </div>			
<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>				

Figura 13.Documentos

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Especialidades</p> <p>en esta seccion puedes agregar, editar o eliminar sus especialidades y areas del conocimiento</p> <p>3 </p> <p>Salir</p>
	<p>Especialidad <input type="text"/></p> <p>Descripcion <input type="text"/></p> <p> Guardar Eliminar</p>
	<p>+</p>
	<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>

Figura 14. Especialidades

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Idiomas</p> <p>en esta seccion puedes agregar, editar o eliminar informacion sobre los idiomas que domina (nivel certificado)</p> <p>3 </p> <p>Salir</p>
	<p>Idioma <input type="text"/> Nivel <input type="text"/></p> <p>Certificado <input type="text"/></p> <p> Guardar Eliminar</p>
	<p>+</p>
	<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>

Figura 15. Idiomas

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Proyectos anteriores en esta seccion puedes agregar, editar o eliminar informacion sobre proyectos anteriores donde a trabajado</p>		<p>3</p>	<p>Salir</p>
	<p>Proyecto</p> <p>Descripcion</p> <p>Guardar Eliminar</p>			
	<p>+</p>			
	<p>El estado de tu CVU es: 20% completado</p>			

Figura 16. Proyectos anteriores

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Publicaciones en esta seccion puedes agregar, editar o eliminar infoamcion sobre las publicaciones que ha hecho o a participado</p>		<p>3</p>	<p>Salir</p>
	<p>Nombre de la pub.</p> <p>Descripcion</p> <p>Guardar Eliminar</p>			
	<p>+</p>			
	<p>El estado de tu CVU es: 20% completado</p>			

Figura 17. Publicaciones

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Trabajos</p> <p>en esta seccion puedes agregar, editar o eliminar informacion sobre los ultimos trabajos que ha tenido</p> <p>3 </p> <p>Salir</p>
	<p>Trabajo</p> <p><input type="text"/></p> <p>Nombre del jefe</p> <p><input type="text"/></p> <p>¿Continua trabajando? <input type="checkbox"/> Tiempo trabajado <input type="text"/></p> <p>Descripcion</p> <p><input type="text"/></p> <p> Guardar Eliminar</p> <p></p>
<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>	

Figura 18.Tecnologías

<p>Juan Perez Administrador</p> <p>CVU ▼</p> <p>Visualisar</p> <p>Editar</p> <p>Mensajes</p> <p>Busqueda</p>	<p>Logo</p> <p>Tecnologías que domina</p> <p>en esta seccion puedes agregar, editar o eliminar informacion sobre las tecnologias que domina(maquinaria, software,etc.)</p> <p>3 </p> <p>Salir</p>
	<p>Tecnologia <input type="text"/></p> <p>Descripcion</p> <p><input type="text"/></p> <p> Guardar Eliminar</p> <p></p>
<p>El estado de tu CVU es: <input type="text"/> 20% completado</p>	

Figura 19. Trabajos

9.3 Programación de la aplicación

La programación del sistema se realizó utilizando el lenguaje de desarrollo web PHP, haciendo uso del framework de aplicaciones web Laravel, el cual facilita el desarrollo de la aplicación y además permite que sea facil de añadir funcionalidades extras al sistema en un futuro.

El framework Laravel esta basado en el modelo arquitectónico de capas “MVC” (modelo, vista, controlador), donde cada una de las capas se le asigna un papel dentro del sistema para hacer funcionar a este.

Modelo: El modelo se encarga de decirle al sistema donde y como debe guardar la informacion y tambien como debe accederla, en laravel los modelos de las tablas que componen la base de datos se guardan en la carpeta models, cuando se crea el archivo del modelo de una tabla se le especifican qué campos contiene esa tabla, en que orden se llenan y si son accesibles o no.

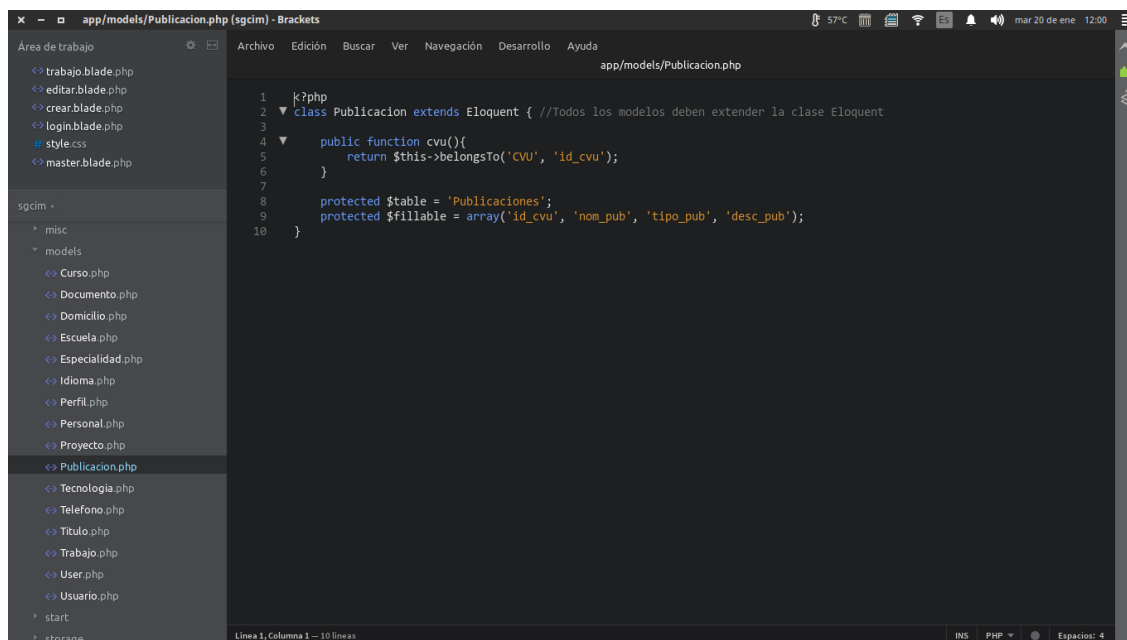
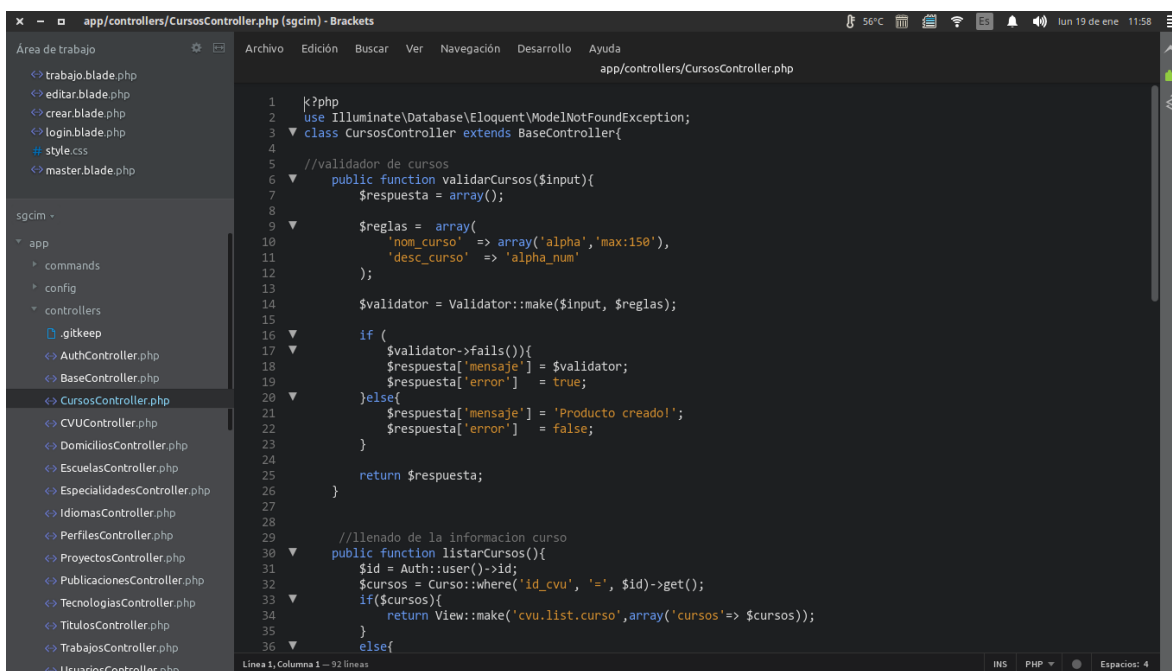


Figura 20. Ejemplo de Modelo

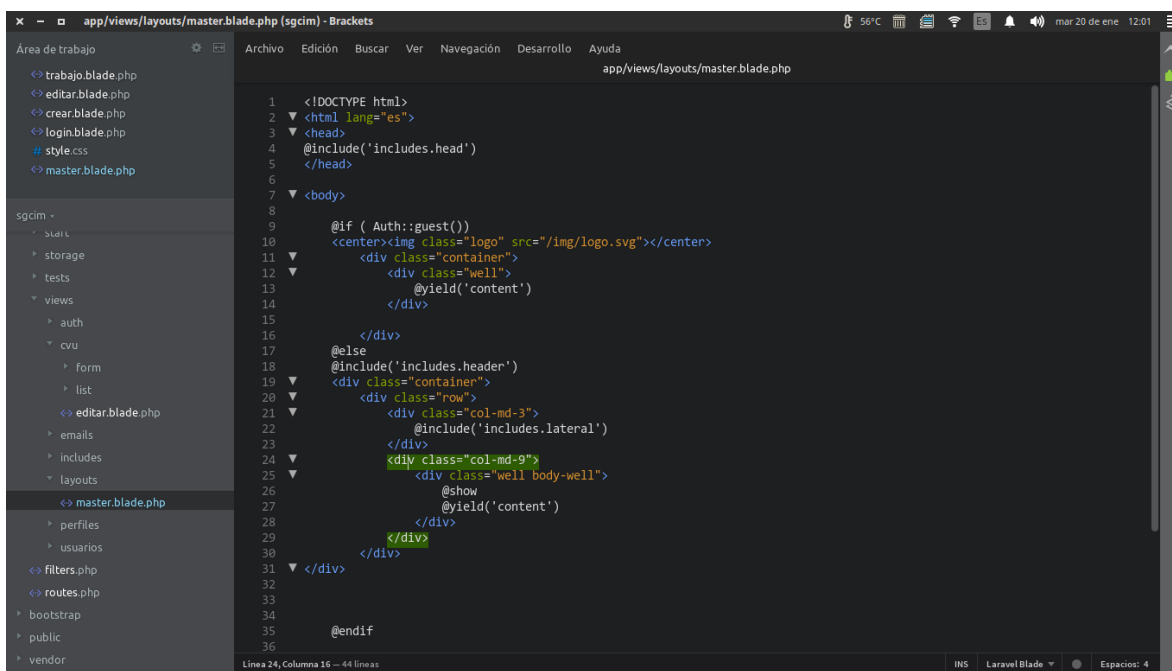
Controlador: Los controladores se encargan de la parte conocida como Modelo de negocio, osea hacen que el programa cumpla las funciones para las que fue creado, procesan la información, realizan tareas, automatizar algún proceso en el sistema, etc. Estos se guardan dentro de la carpeta controllers, al crear un controlador nuevo para el sistema se debe extender el controlador base que se encuentra por default en el framework y desde ahí se crean las clases necesarias para que el controlador realice la función deseada.



```
1  <?php
2  use Illuminate\Database\Eloquent\ModelNotFoundException;
3  class CursosController extends BaseController{
4
5      //validador de cursos
6      public function validarCursos($input){
7          $respuesta = array();
8
9          $reglas = array(
10             'nom_curso' => array('alpha','max:150'),
11             'desc_curso' => 'alpha_num'
12         );
13
14         $validator = Validator::make($input, $reglas);
15
16         if (
17             $validator->fails()){
18             $respuesta['mensaje'] = $validator;
19             $respuesta['error'] = true;
20         }else{
21             $respuesta['mensaje'] = 'Producto creado!';
22             $respuesta['error'] = false;
23         }
24
25         return $respuesta;
26     }
27
28
29     //llenado de la informacion curso
30     public function listarCursos(){
31         $id = Auth::user()->id;
32         $cursos = Curso::where('id_cvu', '=', $id)->get();
33         if($cursos){
34             return View::make('cvu.list.curso',array('cursos'=> $cursos));
35         }
36     }
37 }
```

Figura 21. Ejemplo de Controlador

Vistas: Las vistas le indican al sistema de qué forma debe mostrar la información que está almacenada, además permiten que el usuario interactúe con el sistema, esta es la capa con la que el usuario toma contacto de forma directa. En laravel las vistas se guardan en la carpeta views y estas estan programadas en codigo HTML y con extensiones del lenguaje de plantillas Blade del propio framework.



```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 @include('includes.head')
5 </head>
6
7 <body>
8
9 @if ( Auth::guest() )
10 <center></center>
11 <div class="container">
12 <div class="well">
13 @yield('content')
14 </div>
15 </div>
16
17 @else
18 @include('includes.header')
19 <div class="container">
20 <div class="row">
21 <div class="col-md-3">
22 @include('includes.lateral')
23 </div>
24 <div class="col-md-9">
25 <div class="well body-well">
26 @show
27 @yield('content')
28 </div>
29 </div>
30 </div>
31 </div>
32
33
34 @endif
```

Figura 22. Ejemplo de una vista

9.4 Capturas de pantalla del sistema en funcionamiento

A continuación se muestran algunas capturas de pantalla del sistema en funcionamiento una vez finalizada su programación, se pueden apreciar algunos cambios en el diseño de los formularios pues se decidió que estos cambios serán más amigables para el usuario.

Cabe aclarar que en las capturas se verán algunos textos de relleno de espacio y que algunas partes del sistema no están terminadas, pero estas no forman parte del proyecto, se están trabajando en los textos que reemplazarán a los textos de relleno y las partes no terminadas son de las ampliaciones que se tiene pensado hacer al sistema en un futuro.

Figura 23. Ventana de login

The screenshot shows a web browser window with the address bar displaying 'proyecto.dev/login'. The page features the Fundación Markoptic logo at the top. Below the logo, there is a login section titled 'Iniciar Sesión' with a link '¿Olvide mi contraseña?'. The login form includes an email field with 'alexandrpunk@gmail.com', a password field with four dots, a 'Recordar mis datos' checkbox, and an 'Iniciar Sesión' button. To the right of the login form, there is a paragraph of placeholder text and a green button labeled 'Crear un nuevo usuario (Integrate)'. At the bottom of the browser window, the address bar shows 'proyecto.dev/usuarios/nuevo'.

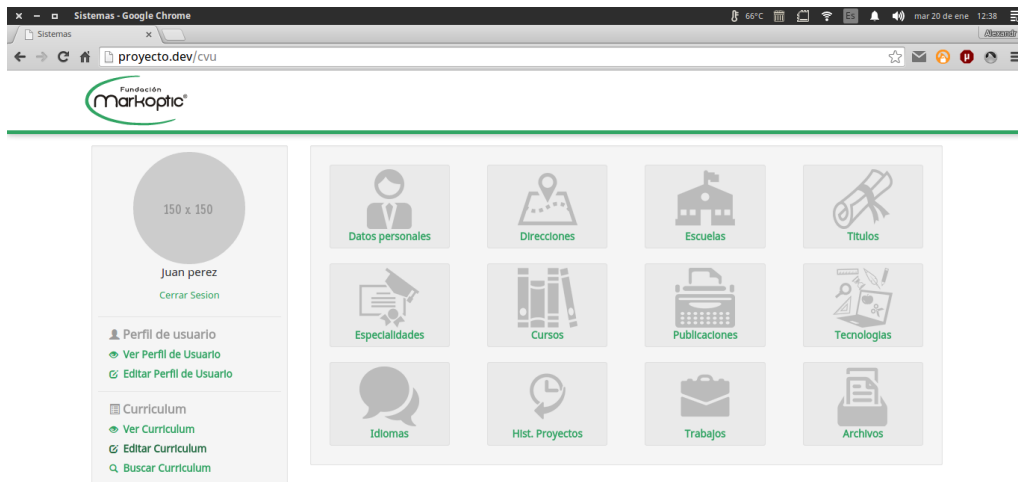
esta es la ventana de donde se muestra el formulario de inicio de sesion y el enlace al formulario de registro.

Figura 24. Formulario de registro

The screenshot shows a web browser window with the address bar displaying 'proyecto.dev/usuarios/nuevo'. The page features the Fundación Markoptic logo at the top. Below the logo, there is a registration form titled 'volver' with the subtitle 'Nombre del usuario'. The form includes fields for 'Nombre del usuario' (with placeholder 'Ingresa tu(s) nombre(s)'), 'apellidos del usuario' (with placeholder 'Ingresa tus apellidos'), 'Sexo' (with radio buttons for 'Hombre' and 'mujer'), 'Correo Electronico' (with placeholder 'Correo Electronico'), 'Contraseña' (with placeholder 'Contraseña'), and a 'Perfil' dropdown menu with 'residente' selected. At the bottom of the form, there are two buttons: 'Guardar' and 'Resetear'.

el formulario de registro es donde se pide la información básica necesaria para registrarse dentro del sistema.

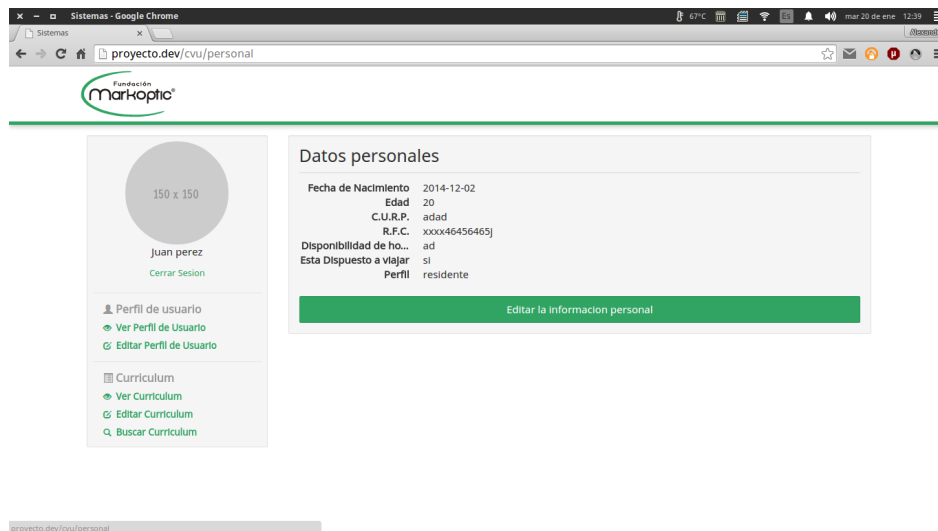
Figura 25. Edición de curriculum



en esta pantalla es donde se nos muestran las opciones a llenar de nuestro curriculum.

Figura 26. Pantallas varias de llenado del Curriculum

A continuación se muestran algunas capturas de las opciones a llenar y de los formularios donde el usuario deposita su información personal.





150 x 150

Juan perez
 Cerrar Sesión

Perfil de usuario
 Ver Perfil de Usuario
 Editar Perfil de Usuario

Curriculum
 Ver Curriculum
 Editar Curriculum
 Buscar Curriculum

Ingrese sus datos personales

Fecha de Nacimiento: 2014-12-02

Edad: 20

¿Estaría dispuesto a viajar?: Si

CURP: adaa...
 RFC: XXXX46456465J

Disponibilidad de Horario: ad



150 x 150

Juan perez
 Cerrar Sesión

Perfil de usuario
 Ver Perfil de Usuario
 Editar Perfil de Usuario

Curriculum
 Ver Curriculum
 Editar Curriculum
 Buscar Curriculum

Domicilios

Casa de guasave
 av. adolfo lopez mateos, col del bosque, #462
 guasave, guasave, sinaloa
 mexico

Editar Borrar

oficina de los mochis
 av rio lerma #5678 altos, col morelos
 los mochis, ahome, sinaloa
 mexico

Editar Borrar

agregar un domicilio



150 x 150

Juan perez
 Cerrar Sesión

Perfil de usuario
 Ver Perfil de Usuario
 Editar Perfil de Usuario

Curriculum
 Ver Curriculum
 Editar Curriculum
 Buscar Curriculum

Ingrese los datos de su domicilio

Nombre del Domicilio: oficina de los mochis

Domicilio: av rio lerma #5678 altos, col morelos

Ciudad: los mochis Municipio: ahome

Estado: sinaloa Pais: mexico

10. Conclusiones

Tras concluir mi proyecto puedo darme cuenta de la importancia que tiene la ingeniería de software en el desarrollo de aplicaciones de calidad. Desde el comienzo de mi proyecto hasta su culminación hice uso de todos los conocimientos adquiridos durante mi curso universitario.

Espero que todo el conocimiento aplicado en este proyecto cumpla con los requisitos de Fundación Markoptic y que además sirva para sentar las bases para los futuros proyectos de software que surjan dentro de esta.

Este sistema gestor de Curriculums es un ejemplo de los beneficios que brindan el uso de patrones, un estilo arquitectónico definido y demás buenas prácticas de ingeniería, pues el producto final es de fácil mantenimiento y está preparado para absorber cambios y crecer conforme crezcan las necesidades.

El tiempo de desarrollo del proyecto fue bastante largo y eso se debió a que todo se comenzó desde cero, pues no solo fue la creación del proyecto en sí, sino también fue la creación de las bases para los futuros proyectos que se desarrollen en Markoptic.

Debido a esto, he podido utilizar todos mis conocimientos como ingeniero en sistemas en la realización de este proyecto de residencias, por lo cual ha sido una valiosa experiencia que me dejara conocimiento muy valioso para mi futura vida laboral.

Bibliografía

Gandy, E. (2005). JavaScript, creating dynamic web pages.LEXDEN PUBLISHING.

García, M. (2009). Curso CSS.

Garlan, D. (1996). Style-based refinement for software architecture. THE ACM GUIDE TO COMPUTING LITERATURE.

Lorna Mitchell, D. S. (2011). PHP MASTER, WRITE CUTTING-EDGE CODE. SITEPOINT.

Niederst, J. (2002). HTML Pocket Reference.O'REILLY.

Pressman, R. S. (2002). Ingeniería de software, un enfoque práctico. MC GRAW HILL.

Reynoso, C. B. (2004). Introducción a la arquitectura de software.UNIVERSIDAD DE BUENOS AIRES.

Robbins, J. N. (2007). Learning Web Design, a begginers guide. O'REILLY.

Ullman, L. (2007). PHP for the web Third Edition. O'REILLY.

Dayle Rees , Antonio Laguna. (2013). Laravel: Code Bright (ES)