



**UNIVERSIDAD
NACIONAL
DE LOJA**



Carrera de Ingeniería en
Sistemas / Computación

Facultad de Energía, las Industrias y los Recursos Naturales No
Renovables

CARRERA DE INGENIERÍA EN COMPUTACIÓN

Proyecto de Software Engineering Management

Autores: Yamilka Valeria Erazo Aleaga

Rubier Andrés Padilla Loaiza

LOJA-ECUADOR

NOVIEMBRE 2022

Índice

1	Introducción.....	1
2	Desarrollo	2
2.1	Temática	2
2.2	Objetivos	2
2.2.1	Objetivo general.....	2
2.2.2	Objetivos específicos	2
2.3	Alcance.....	2
2.4	Tecnologías (Lenguajes, metodologías, plataformas).....	4
2.4.1	Lenguajes de programación	4
2.4.2	Framework de JavaScript.....	6
2.4.3	Base de datos.....	7
2.4.4	Herramientas de desarrollo	7
2.4.5	Herramienta de producción.....	8
2.4.6	Pasarelas de pagos.....	8
2.5	Metodología, filosofía o arquitectura de software	8
2.6	Proceso en notación BPM (bpmn)	10
3	Conclusiones.....	11
4	Recomendación	12
5	Referencias Bibliográficas.....	13

1 Introducción

Actualmente, el software se encuentra inmerso dentro de las actividades diarias que desempeñan las personas de todo el mundo, ya sean profesionales o personales, volviéndose indispensable para el correcto desarrollo de un país al ser parte de su administración (transporte, luminaria, telecomunicaciones, procesos industriales, agricultura, comercio, etc.), siendo de esta manera, necesaria la aplicación de la ingeniería de software para construir programas informáticos, una disciplina en la que se abordan todas las fases del ciclo de vida del software, todo ello previo a un análisis que permita determinar el tipo de modelo, metodología, actividades y tareas necesarias a efectuar por parte del equipo de trabajo para asegurar la calidad del sistema.

Los proyectos basados en la ingeniería de software componen una serie de habilidades obtenidas a lo largo de la formación académica de un estudiante que curse una carrera referente a la informática, y al momento de emplearlas en un escenario real, se percatará de que son procesos complejos que involucran distintos profesionales, ingeniería del software, sistemas de información y procedimientos complicados, que, al finalizar, buscan obtener que el sistema cumpla con las expectativas del cliente y satisfaga sus necesidades. En el presente informe se expone la temática de un proyecto, junto a los objetivos que se desean lograr para la construcción de una aplicación web para el restaurante “Yogupop”, además del alcance, las tecnologías que se aplicaran y la metodología que guiará al mismo.

2 Desarrollo

2.1 Temática

Aplicación Web de pedidos, pagos y acceso al menú por código QR del restaurante “Yogupop”.

2.2 Objetivos

2.2.1 Objetivo general

- Implementar una aplicación web para el control de los pedidos, pagos y acceso al menú por código QR del restaurante “Yogupop”.

2.2.2 Objetivos específicos

- Analizar los requerimientos del restaurante, la administración de los productos y servicios que ofrece, las promociones que ofertan y los tipos de pago que poseen, para de esta manera, llegar a posibles soluciones que satisfagan las necesidades del cliente.
- Desarrollar el modelo de datos, la gestión de roles, las tareas y los procesos que se manejarán en la aplicación web.
- Diseñar el Front-end y codificar el Back-end buscando siempre la optimización del código, la fluidez de la aplicación, que sea intuitiva para los usuarios y verificando todo esto a través de la aplicación de pruebas en el entorno del cliente.

2.3 Alcance

El presente trabajo se desarrollará de forma teórica/práctica y culminará con el desarrollo e implementación de una aplicación web que permita administrar los pedidos, la forma de pago y el acceso al menú por código QR del restaurante “Yogupop”, para ello se tiene estimado un tiempo de siete meses, en el cual se busca llevar a cabo las siguientes actividades y tareas por cada uno de los objetivos específicos previamente establecidos:

1. Analizar los requerimientos del restaurante, la administración de los productos y servicios que ofrece, las promociones que ofertan y los tipos de pago que poseen, para de esta manera, llegar a posibles soluciones que satisfagan las necesidades del cliente.
 - Identificar tanto los requerimientos funcionales como no funcionales que el restaurante posee y necesita para el software, mediante entrevistas durante el desarrollo del proceso.
 - Conocer todo el proceso de gestión que emplea el restaurante para sus productos, servicios, promociones y tipos de pago mediante visitas de negocio en su instalación física durante el horario laboral.
 - Planificar las etapas para el desarrollo de la solución del sistema de software con base en la información obtenida.
2. Desarrollar el modelo de datos, la gestión de roles, las tareas y los procesos que se manejarán en la aplicación web.
 - Planificar los plazos de entrega de cada uno de los modelos dentro de las etapas de desarrollo.
 - Diseñar el modelo de datos, especificando los tipos de roles presentes en el sistema junto a los procesos que realizarán.
 - Verificar que los modelos estén correctos al presentarlos al cliente y obtener una retroalimentación que ayude a corregirlos, teniéndose una idea más clara de los planos a seguir en la construcción del código.
3. Diseñar el Front-end y codificar el Back-end buscando siempre la optimización del código, la fluidez de la aplicación, que sea intuitiva para los usuarios y verificando todo esto a través de la aplicación de pruebas en el entorno del cliente.

- Construir las interfaces de usuario aplicando la disciplina Human Computer Interaction (HCI), teniendo bases de diseño gráfico (tipografía, color, etc.), ingeniería de software y test de usabilidad.
- Aplicar el uso de patrones de diseño dentro del Back-end, buscando de esta manera la optimización del código.
- Plantear las pruebas unitarias, integración, del sistema y de aceptación, basándonos en los requisitos iniciales, al nivel que fuese necesario.

2.4 Tecnologías (Lenguajes, metodologías, plataformas)

2.4.1 Lenguajes de programación

Un lenguaje de programación es el medio por el cual los humanos interactúan con las computadoras, a través del uso de reglas sintácticas y semánticas que serán procesadas y darán cierta salida de datos, para ello se hace uso de algoritmos que son instrucciones con un principio y un fin y tienen sentido.

Existen varios lenguajes de programación y cada uno es utilizado en un escenario específico por las características que lo componen, en este caso el lenguaje más adecuado para la creación de una aplicación web es JavaScript.

JavaScript: JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB y Adobe Acrobat. JavaScript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo, programación funcional). [1]

En un principio cada navegador tenía su variante de este lenguaje, por lo que, se creó el estándar ECMAScript (ECMA-262), de manera que, sea posible utilizar un mismo código en varios navegadores, existiendo documentación y algunos artículos de MDN que explican a detalle los estándares actuales. A continuación, se presentan las ventajas y desventajas de este lenguaje de programación:

Ventajas:

- Es un lenguaje muy sencillo.
- Es rápido, por lo tanto, tiende a ejecutar las funciones inmediatamente.
- Cuenta con múltiples opciones de efectos visuales.
- Es soportado por los navegadores más populares y es compatible con los dispositivos más modernos, incluyendo iPhone, móviles y PS3.
- Es muy versátil, puesto que es muy útil para desarrollar páginas dinámicas y aplicaciones web. [2]

Desventajas:

- En el FrontEnd sus códigos son visibles, por lo tanto, pueden ser leídos por cualquier usuario.
- Tiende a introducir gran cantidad de fragmentos de código en los sitios web.
- Sus opciones 3D son limitadas, si se quiere utilizar este lenguaje de programación para crear un juego, deben emplearse otras herramientas.
- No es compatible en todos los navegadores de manera uniforme.
- Los usuarios tienen la opción de desactivar JavaScript desde su navegador. [2]

2.4.2 Framework de JavaScript

El uso de frameworks acelera el proceso de programar, al existir ciertas librerías creadas con anterioridad que pueden ser reutilizadas por los programadores en la construcción de su propia aplicación, así también, ayudan a organizar y apoyan al trabajo en equipo dentro de un proyecto, cada framework se encuentra basado en un lenguaje de programación, por ende, es que algunos son usados en ciertos campos por las características que poseen. Se presenta el framework a utilizar junto a un marco de desarrollo minimalista que permite estructurar una aplicación web de una manera ágil.

Node.js: Es un entorno de tiempo de ejecución de JavaScript (de ahí su terminación en .js haciendo alusión al lenguaje JavaScript). Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript. También aporta muchos beneficios y soluciona muchísimos problemas. [3]

Node.js utiliza un modelo de entrada y salida sin bloqueo controlado por eventos que lo hace ligero y eficiente (con entrada nos referimos a solicitudes y con salida a respuestas). Puede referirse a cualquier operación, desde leer o escribir archivos de cualquier tipo hasta hacer una solicitud HTTP. [3]

Express.js: A veces también llamado «Express», es un framework de backend Node.js minimalista, rápido y similar a Sinatra, que proporciona características y herramientas robustas para desarrollar aplicaciones de backend escalables. Ofrece el sistema de enrutamiento y características simplificadas para ampliar el framework con componentes y partes más potentes en función de los casos de uso de la aplicación. [4]

El framework proporciona un conjunto de herramientas para aplicaciones web, peticiones y respuestas HTTP, enrutamiento y middleware para construir y desplegar aplicaciones a gran escala y preparadas para la empresa. [4]

No hay que olvidar la construcción de las interfaces de usuario, para ello también existe una biblioteca de JavaScript, la cual es **React**, que ayuda a crear interfaces de usuario interactivas de forma sencilla. Diseñar vistas simples para cada estado de la aplicación, además, se encargará de actualizar y renderizar de manera eficiente los componentes correctos cuando los datos cambien.

2.4.3 Base de datos

MongoDB: Es una base de datos NoSQL orientada a documentos que apareció a mediados de la década de 2000. Se utiliza para almacenar volúmenes masivos de datos.

A diferencia de una base de datos relacional SQL tradicional, MongoDB no se basa en tablas y columnas. Los datos se almacenan como colecciones y documentos.

Los documentos son pares value/key que sirven como unidad básica de datos. Las colecciones contienen conjuntos de documentos y funciones. Son el equivalente a las tablas en las bases de datos relacionales clásicas. [5]

2.4.4 Herramientas de desarrollo

Dentro de las herramientas de desarrollo que se van a usar están:

Visual Studio Code: Es un editor de texto de uso libre el cual es altamente configurable, permitiendo trabajar con cualquier lenguaje de programación con la ayuda de extensiones.

Git: Es un software para la administración de control de versión de código abierto y de uso libre.

GitHub: Es un software el cual permite la colaboración con “git” y a su vez poder crear repositorios en la nube para de esta manera respaldar el trabajo realizado.

2.4.5 Herramienta de producción

La herramienta de producción será utilizada al momento de subir el proyecto a la web para que pueda ser accedido desde cualquier lugar.

Heroku: Es uno de los PaaS más utilizados en la actualidad en entornos empresariales por su fuerte enfoque en resolver el despliegue de una aplicación. En otras palabras, a Heroku solo le dices qué lenguaje de backend estás utilizando o qué base de datos vas a utilizar y te preocupas únicamente por el desarrollo de tu aplicación. [6]

2.4.6 Pasarelas de pagos

Las pasarelas de pagos son sistemas de pago electrónico que permiten la realización de pagos y transferencias entre tiendas electrónicas y entidades bancarias de manera segura. [7]

Se integrará una pasarela de pagos en este proyecto para que se puedan realizar y recibir pagos, se ha seleccionado “PayPal” para ser el medio de pago el cual se va a integrar gracias a sus características, las cuales son:

- Permite transacciones seguras.
- Es fácil de implementar en aplicaciones web.
- Ofrece un entorno de prueba
- Es una de las plataformas de pagos más conocidas de la actualidad.

2.5 Metodología, filosofía o arquitectura de software

Se hará uso de la **metodología ágil de desarrollo de software ICONIX**, que se encuentra entre RUP (Rational Unified Process) y XP (eXtreme Programming). Al derivar directamente de RUP, el diagrama de clases y casos de uso se verán presentes. La metodología sigue un ciclo de vida iterativo e incremental donde se busca obtener el sistema final a partir de los casos de uso.

ICONIX contempla 4 fases, la primera es el análisis de requisitos donde se realiza el modelo del dominio, un diagrama de clases simplificado, realizado únicamente con las clases que existen en la vida real que poseen comportamientos y atributos, los cuales, serán guardados en el sistema, y al ser refinado se obtendrá el UML. La segunda fase es el análisis y diseño preliminar, donde se trata el diagrama de casos de uso y su descripción. La tercera fase es el diseño y tratará la vista de procesos, despliegue y física, de manera que, se llegue a la última fase, que es la implementación, donde se codificará siguiendo el diagrama de clases y de actividades para determinar los métodos que existen y cómo será la mensajería entre los diferentes objetos; resulta adecuado y necesario realizar un diagrama de componentes al hacerse uno de software de terceros.

La arquitectura seleccionada fue la del **Proceso Racional Unificado (RUP)**, el cual, es un claro ejemplo de un modelo de proceso híbrido y moderno que proviene del trabajo en UML y el asociado Proceso Unificado de Desarrollo de Software. Entre las ventajas de usar RUP se encuentran el hacer uso del Modelo de Vistas de Arquitectura 4+1 de Kruchten, donde es posible describir la arquitectura del sistema de software, a través del uso de múltiples vistas, como lo son: vista lógica, vista de procesos, vista de despliegue, vista física, y por último la vista de escenarios donde se presentan los diagramas de caso de uso para ilustrar la arquitectura en su nivel más básico; cada una de ellas describe el software desde un punto de vista diferente para los grupos de interés, tales como usuarios finales, desarrolladores y diseñadores.

2.6 Proceso en notación BPM (bpmn)

3 Conclusiones

- Los proyectos basados en la ingeniería de software permiten a los estudiantes de la carrera de computación aplicar los conocimientos adquiridos en diversas asignaturas dentro de un escenario real, ayudándolos a tener un acercamiento al trabajo que deberán realizar para su titulación.
- La aplicación de una metodología ágil representa una gran ventaja al momento de que el equipo de trabajo se adapte a los nuevos requerimientos del cliente, no solo en las funcionalidades del sistema, sino también en las especificaciones de las tecnologías que desea para la creación del mismo.
- La aplicación de una metodología de software guía al grupo de desarrolladores en la creación del proyecto, siguiendo directrices y técnicas de trabajo que los ayudarán a cumplir sus objetivos, se puede mencionar en el caso de las metodologías ágiles la importancia del cliente al entregar feedback continuo de los prototipos del sistema que son entregados gracias a la iteración continua de las actividades.
- Se decidió usar la metodología ágil ICONIX, al poseer características de XP y RUP, donde el cliente cumple un papel importante en la elicitación de requerimientos, proceso que no se realiza únicamente al principio del desarrollo, sino que es continuo a lo largo de la creación del sistema, y ello se lo evidencia en la fase de análisis de requerimientos con la realización de las interfaces gráficas que tendrán correcciones continuas al ser presentadas al cliente.

4 Recomendación

- Al momento de seleccionar una metodología de desarrollo de software, existen diversas cuestiones que se tiene que tener en consideración, entre ellas, se encuentra el tamaño del proyecto, el número de involucrados en el grupo de trabajo, si los requisitos se mantendrán constantes, entre otras, para de esta manera, utilizar la que mejor se adapte al presente escenario.

5 Referencias Bibliográficas

[1] JavaScript (s.f). [en línea]. Disponible en:

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

[2] Conoce las características, ventajas y desventajas de JavaScript (s.f). [en línea]. Disponible en: <https://www.nextu.com/blog/conoce-las-ventajas-y-desventajas-de-javascript/>

[3] J. Lucas, (2019, Sep. 4). “Qué es NodeJS y para qué sirve”. [en línea]. Disponible en:

<https://openwebinars.net/blog/que-es-nodejs/>

[4] ¿Qué es Express.js? Todo lo que Debes Saber (2022). [en línea]. Disponible en:

<https://kinsta.com/es/base-de-conocimiento/que-es-express/>

[5] MongoDB: todo sobre la base de datos NoSQL orientada a documentos (2022). [en línea].

Disponible en: [https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-](https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-orientada-a-documentos#:~:text=ingenier%C3%ADa%20de%20datos,-.MongoDB%20es%20una%20base%20de%20datos%20NoSQL%20orientada%20a%20documentos,almacenar%20como%20colecciones%20y%20documentos)

[orientada-a-documentos#:~:text=ingenier%C3%ADa%20de%20datos,-](https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-orientada-a-documentos#:~:text=ingenier%C3%ADa%20de%20datos,-.MongoDB%20es%20una%20base%20de%20datos%20NoSQL%20orientada%20a%20documentos,almacenar%20como%20colecciones%20y%20documentos)

[.MongoDB%20es%20una%20base%20de%20datos%20NoSQL%20orientada%20a%20documentos,almacenar%20como%20colecciones%20y%20documentos](https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-orientada-a-documentos#:~:text=ingenier%C3%ADa%20de%20datos,-.MongoDB%20es%20una%20base%20de%20datos%20NoSQL%20orientada%20a%20documentos,almacenar%20como%20colecciones%20y%20documentos)

[6] ¿Qué es Heroku? Cómo funciona la plataforma y para qué sirve (2017). [en línea].

Disponible en: [https://platzi.com/blog/que-es-](https://platzi.com/blog/que-es-heroku/?utm_source=google&utm_medium=cpc&utm_campaign=18798607679&utm_adgroup=&utm_content=&gclid=CjwKCAjwtp2bBhAGEiwAOZZTuCZEwnswzFyww4EMXtcFYPC2l8Ve537aNA8E4s7PvUsaGv8JTVJF3RoCxwwQAvD_BwE&gclsrc=aw.ds)

[heroku/?utm_source=google&utm_medium=cpc&utm_campaign=18798607679&utm_adgroup=&utm_content=&gclid=CjwKCAjwtp2bBhAGEiwAOZZTuCZEwnswzFyww4EMXtcFYPC2l8Ve537aNA8E4s7PvUsaGv8JTVJF3RoCxwwQAvD_BwE&gclsrc=aw.ds](https://platzi.com/blog/que-es-heroku/?utm_source=google&utm_medium=cpc&utm_campaign=18798607679&utm_adgroup=&utm_content=&gclid=CjwKCAjwtp2bBhAGEiwAOZZTuCZEwnswzFyww4EMXtcFYPC2l8Ve537aNA8E4s7PvUsaGv8JTVJF3RoCxwwQAvD_BwE&gclsrc=aw.ds)

[7] D. Solis Fonseca, W. Roque Pérez y M. Morilla Faurés, (2013, Jul. 7). “PASARELA DE Pagos para la seguridad de transacciones bancarias en línea”. [en línea]. Disponible en:

<https://www.3ciencias.com/wp-content/uploads/2013/08/pasarela-de-pagos.pdf>