

**ESCUELA POLITÉCNICA NACIONAL**  
**ESCUELA DE FORMACIÓN DE TECNÓLOGOS**



**FUNDAMENTOS DE DESARROLLO DE SOFTWARE**  
**(TDSR324)**

**ASIGNATURA:** Fundamentos Desarrollo de Software

**PROFESOR:** Ing. Juan Pablo Zaldumbide

**PERÍODO ACADÉMICO:** 2020-B

**Informe Técnico**  
**Jamafud Web App**  
**Grupo 12**

**TÍTULO:** PROYECTO FINAL

## Introducción

### 1. Objetivo General

- Desarrollar una aplicación web que genere un presupuesto y una lista de ingredientes en base a un menú semanal escogido por el usuario.

### 2. Objetivos Específicos

- Disminuir la cantidad de dinero semanal invertido en comida comprada de manera esporádica.
- Disminuir la cantidad de veces que el usuario sale de casa.
- Disminuir la posibilidad de contagio de COVID-19 al evitar que los usuarios frecuenten varias veces las tiendas y supermercados locales.
- Facilitar la vida al usuario, presentar una solución que ayudará a planear un menú alimenticio saludable, de una manera rápida y divertida.

### 3. Alcance

Una problemática contemporánea a la cual el mundo se enfrenta dentro del contexto de la pandemia es el uso irresponsable de recursos alimenticios y las aglomeraciones en lugares de compras de víveres de primera necesidad. En Quito, existe el recurrente problema de concentración de personas que no obedecen a las medidas de distanciamiento. Las personas salen varias veces de casa debido a que no cuentan con una planificación semanal o mensual de sus compras, además no cuentan con un presupuesto establecido para adquirir alimentos. “Un sondeo realizado por Ipsos durante la emergencia sanitaria muestra que en el 58% de los 802 encuestados, la frecuencia de compra de alimentos creció, los encuestados además indicaron que están realizando más compras en tiendas de barrio, más que los supermercados” [1]. Además, los gastos que realizan los ciudadanos han aumentado considerablemente, en marzo de 2020 el promedio de gasto semanal en los hogares creció en marzo un 30%” [1]. Una manera de frenar este comportamiento y colaborar de manera eficiente es optimizar la planificación alimenticia y presupuestaria de los quiteños mediante la aplicación Jamafud, la cual permitirá crear un menú semanal y un presupuesto para que el usuario tenga que salir una sola vez a comprar alimentos frescos y también gastar lo planificado. Con esto también se pretende disminuir la posibilidad de contagio de COVID-19.

## Metodología

### 4. Metodología de Desarrollo SCRUM

El desarrollo del presente proyecto se lo ha implementado bajo *Scrum*, una metodología ágil que permite evidenciar el desarrollo del proyecto a través de *Sprints* o iteraciones que se planifican para completar cada objetivo de forma exitosa. Las razones por las cuales se usa esta metodología en el siguiente trabajo debido

a que esta metodología “permite adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto y solapa las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o en cascada” [2], también debido a que el equipo de desarrollo de este proyecto es pequeño y tiene el objetivo de presentar avances semanales para dar forma poco a poco a la aplicación web Jamafud los cuales son calificadas por el docente.

## Roles

Los roles para utilizar son los de producto *owner*, *scrum master* y *development team*. Los roles tienen una participación indispensable en el proyecto, ya que definen las responsabilidades que cada miembro tiene que cumplir para asegurar que la información brindada sea correcta y que la comunicación con el cliente sea eficiente para la mejora del proyecto. Es vital que cada persona encargada cumpla con las responsabilidades para el éxito de cada iteración.

### *Product Owner*

Es el propietario del software, toma las decisiones, conoce el negocio y la visión del producto. Tiene la responsabilidad de incrementar el valor del producto y mantener el control de los entregables. Este rol está representado por Kevin Morales propietario de “Jamafud”, quien otorga la información sobre cómo se lleva a cabo el modelo de negocio de la aplicación y la idea general del proyecto.

### *Scrum Master*

Es la persona encargada de que se sigan las prácticas y valores descritos en la metodología *Scrum*, guiando y orientando al equipo de trabajo hacia el éxito durante el desarrollo del proyecto. Este rol lo representa la Ing. Ivonne Maldonado quien supervisa al equipo de desarrollo para que cada fase sea implementada adecuadamente.

### *Development Team*

Son los profesionales que entregan un incremento potencial del producto al terminar cada *Sprint*. Solo los integrantes del *development team* realizan la creación de un incremento a partir del *Product Backlog*. Este rol es representando por todos los integrantes del proyecto.

La **TABLA I** presenta la manera en la que está conformado el equipo de trabajo en el presente proyecto de titulación.

**TABLA I** Roles del Proyecto

<b>Rol</b>	<b>Integrante</b>
<i>Product Owner</i>	Kevin Morales
<i>Scrum Master</i>	Ing. Juan Pablo Zaldumbide
<i>Development Team</i>	Carlos Quel
	Alejandro Armas
	Kevin Morales

## Artefactos

Para el desarrollo del presente proyecto se van a usar los artefactos que plantea la metodología SCRUM los cuales son herramientas o elementos que garantizan el funcionamiento, transparencias y registro de la información clave dentro del proceso *de desarrollo*. Con esto se pretende entregar calidad en el producto.

## Recopilación de Requerimientos

El levantamiento o recopilación de requerimientos hace referencia a la identificación y documentación de requerimientos de un proyecto, los cuales provienen de los usuarios, cliente y/o interesados (*stakeholders*).

El equipo de desarrollo se reúne con el propietario del producto de software con el objetivo de entender el problema a resolver, identificar y definir los requerimientos del proyecto, que se encuentran detallados en el Manual Técnico – Sección Recopilación de Requerimientos.

## Historias de Usuario

Es una explicación general de un requerimiento o función del proyecto, escrita desde la perspectiva del cliente o usuario final, con el propósito de articular el trabajo a realizar por el *development team* como valor específico para el producto final. Estas historias permitirán dividir las tareas y también diferenciar la prioridad y la importancia que le da el producto *owner*.

En las Historias de Usuario se establecen todas las funcionalidades a implementar en la aplicación web. La **TABLA II** presenta un ejemplo de modelo de Historias de Usuario que se han utilizado.

Las Historias de Usuario restantes se ubican en el Manual Técnico – Sección Historias de Usuario.

**TABLA II:** Historia de usuario Nro. 7: Asignar cantidad de productos

Historia de Usuario	
<b>Identificador:</b> HU03	<b>Usuario:</b> Cliente
<b>Nombre Historia:</b> : Ingresar Cantidad de Personas	
<b>Prioridad en Negocio (Alta/Medio/Baja):</b> Media	<b>Riesgo en Desarrollo (Alta/Medio/Baja):</b> Alta
<b>Iteración asignada:</b> 3	
<b>Responsable (es):</b> Kevin Morales	
<b>Descripción:</b> El usuario puede ingresar la cantidad de personas que van a comer los platillos seleccionados.	
<b>Observación:</b> El propósito de ingresar la cantidad de usuarios es la de calcular la cantidad de ingredientes que se va a necesitar para elaborar dichos platillos y su respectivo costo.	

### ***Product Backlog***

Es una lista ordenada que avanza a medida que crece el producto y es la única fuente de requisitos para realizar cualquier cambio. Esta lista prioriza los requisitos y representa las expectativas del cliente con respecto a las entregas del producto final. El *Product Backlog* definido para el desarrollo de este proyecto se detalla en el Manual Técnico – Sección *Product Backlog*.

### ***Sprint Backlog***

Es una lista con los elementos funcionales y las diferentes tareas a ser desarrolladas día a día en cada *sprint*. Cada *sprint* cumplido es un entregable e incremento dentro del proyecto. El *Sprint Backlog* definido para el desarrollo de este proyecto se detalla en el Manual Técnico – Sección *Sprint Backlog*.

## **5. Diseño de interfaces**

Establecidos los requerimientos se inicia la construcción del diseño de las interfaces (*mockups*) de la Aplicación Móvil usando la aplicación FIGMA.

### **Figma**

Es una herramienta que permite crear prototipos interactivos de plataformas digitales de manera rápida y sencilla, donde los desarrolladores trabajan de forma colaborativa mediante un grupo. Además, permite cargar imágenes de todo tipo y agregar la funcionalidad que se desee, de manera que el usuario tiene una idea de las interacciones de la implementación final. Otra ventaja de esta herramienta es que permite establecer colores según el manual de marca para usarlas en todo el diseño de la aplicación.

Para el prototipo de la Aplicación Web se han establecido 12 pantallas, bajo el diseño y formato que se ilustra en la **Fig. 1**. El diseño completo definido para el desarrollo de este proyecto se encuentra en Manual Técnico – Sección Diseño de Interfaces.



**Fig. 1:** Prototipo de interfaz

## 6. Diseño de la arquitectura

La elección del patrón de arquitectura permite dar una solución eficaz a un determinado problema planteado. A continuación, se describe el modelo arquitectónico que se utilizó en el desarrollo de este proyecto.

### Patrón arquitectónico Modelo Vista Controlador (MVC)

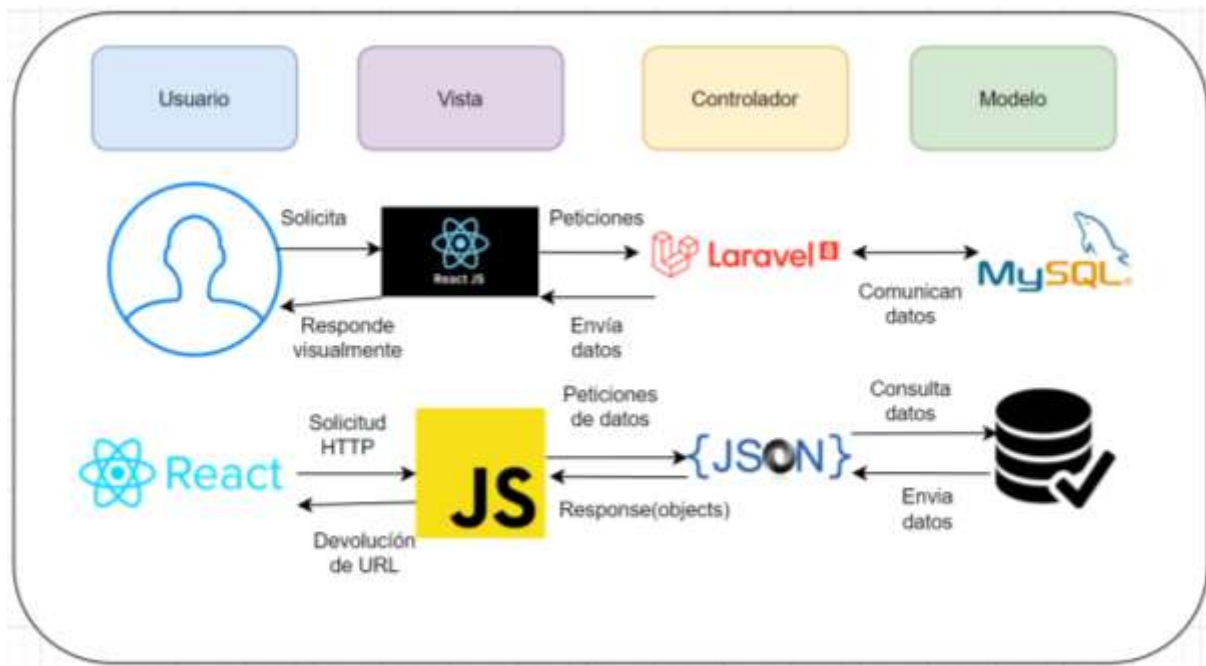
El patrón MVC separa la interfaz de usuario, la lógica de control y los datos en tres componentes de una aplicación en los siguientes componentes:

**Modelo:** contiene los datos que maneja y usa la aplicación, la lógica del negocio y la persistencia.

**Vista:** conocida como la interfaz de usuario, es aquella que muestra la información que se envía al cliente y los métodos de interacción que tiene con este.

**Controlador:** es el intermediario entre el Modelo y la Vista, controla y administra el flujo de información entre ambos. Además, transforma los datos según las necesidades de cada uno.

La **Fig. 2** muestra el flujo del MVC del proyecto, en función a las herramientas utilizadas, permitiendo que el proyecto este organizado y con un desarrollo mejor estructurado.



**Fig. 2:** Patrón arquitectónico de la Aplicación Jamafud

## 7. Herramientas de desarrollo

Las herramientas de desarrollo que se usan son acordes a la arquitectura definida para la Aplicación Web. La **TABLA III** muestra las herramientas indispensables para avanzar en el desarrollo del proyecto.

**TABLA III:** Herramientas para el desarrollo de la Aplicación Móvil

Herramienta		Justificación
<b>Laravel</b>	Marco de trabajo de código abierto para el desarrollo aplicaciones y servicios web con PHP 8. Sus bases se centran en desarrollar código PHP de forma simple, evitando el "código espagueti". El uso de <i>Eloquent</i> los que permite trabajar con base de datos relacionales y el manejo de objetos directamente sin el uso de consultas SQL.	Su implementación permite generar una API de manera sencilla, obteniendo los datos del lado del servidor y presentarlos al cliente de forma eficiente. También el manejo de los datos de las bases SQL a objetos de tipo JSON mediante el uso de recursos API.
<b>MySQL</b>	Es un sistema de gestión de bases de datos relacionales, de código abierto y que cuenta con una versión comercial gestionada por Oracle. Usa la arquitectura en el modelo Cliente – Servidor.	Es fácil de usar, además de permitir el almacenamiento y la consulta de los datos de forma simple a través de relaciones entre tablas.
<b>Composer</b>	Es un gestor de dependencias de PHP que permite descargar, declarar y mantener actualizados los paquetes de software para un proyecto que se basado en PHP.	Es la herramienta necesaria para mantener actualizado ciertas dependencias necesarias en el <i>back-end</i> y utilizadas por la herramienta principal Laravel
<b>Postman</b>	Herramienta que se utiliza para realizar pruebas a una API REST, además permite monitorizarlas, escribir pruebas automatizadas, etc.	Su uso permite dos partes importantes, la primera es hacer <i>testings</i> a los <i>endpoints</i> como el de autenticación y de ingreso de datos a la API y la segunda es ver como llegan o se reciben los datos al realizar una petición al API.
<b>XAMPP</b>	Es una distribución que contiene varios softwares libres como lo son el servidor web Apache, el sistema de administración de BBDD MariaDB y	Su uso permite levantar un servidor local que ayuda a conectar herramientas como la base de datos y la API, además de permitir mantener levantada la API para la conexión al <i>front-end</i> y manejar los datos.

	MySQL y los lenguajes de programación Pearl y PHP.	
<b>NodeJS</b>	Es un entorno de tiempo real con todas las dependencias necesarias para ejecutar el código escrito en JavaScript y el resultado únicamente se lo puede visualizar en un navegador como si fuera una aplicación independiente.	Su uso permite instalar los paquetes necesarios para React y obtener librerías específicas para el desarrollo de <i>front-end</i> .

## Librerías

La **TABLA IV** muestra un conjunto de bibliotecas a utilizar que complementan el desarrollo de la Aplicación Móvil para el presente proyecto de titulación.

**TABLA IV:** Librerías usadas en el desarrollo de la Aplicación Móvil

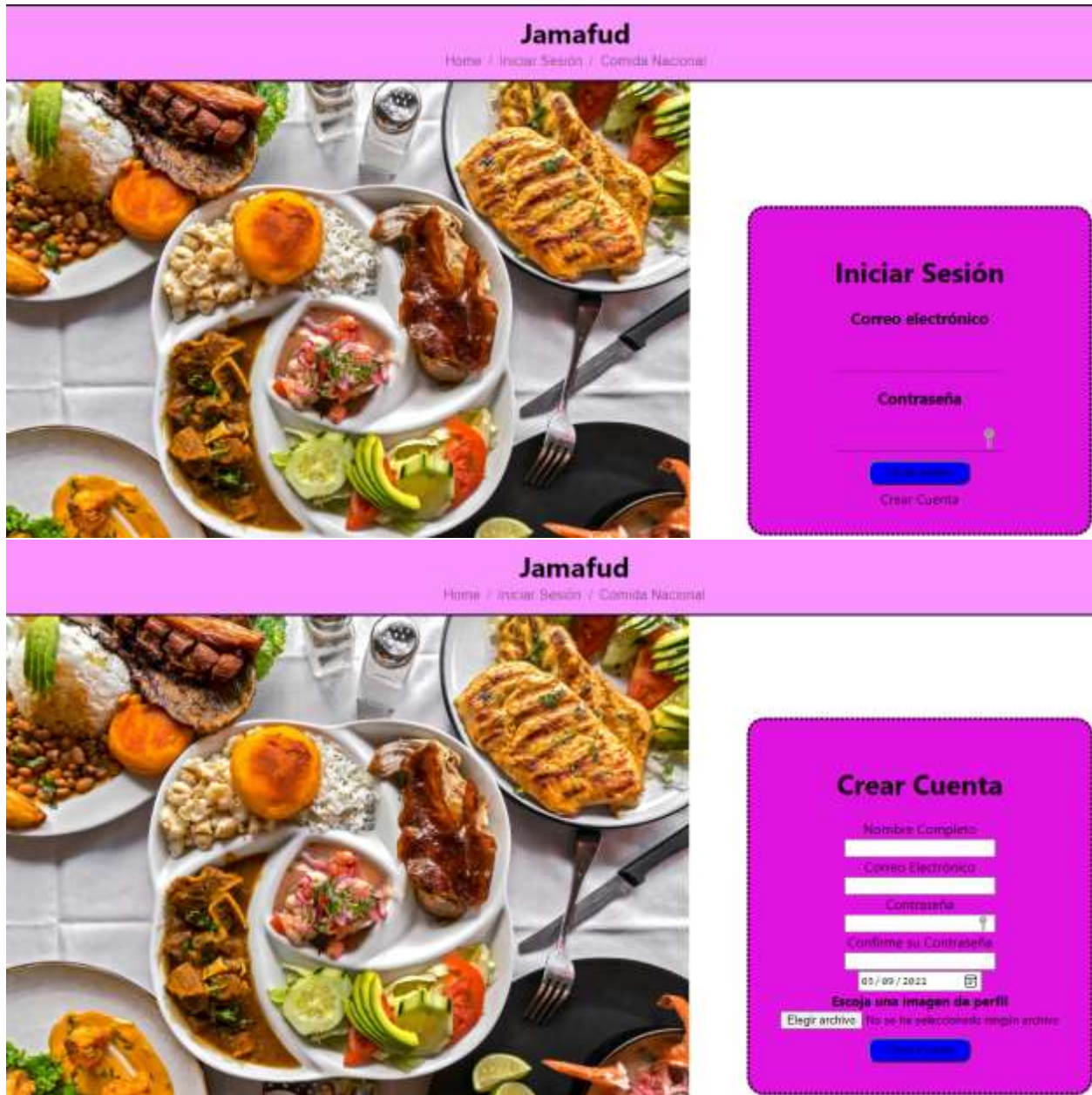
Librería		Justificación
<b>React</b>	Es una biblioteca de JS que permite crear interfaces de usuario de forma sencilla y con un alto nivel de interacción, se basa en componentes los cuales se actualizan y se renderizan al cambiar los datos.	Es la biblioteca en la que se basa el proyecto por ser rápido y ligero, permitiendo usar componentes web en el tema desarrollo móvil.
<b>AntD</b>	Es una librería de componentes para React, los cuales facilitarán la implementación del Front End. Esta librería contiene componentes para formularios que prácticamente requieren sólo un poco de CSS ajustado al branding del producto.	Los componentes de Ant Design pueden ser importados y complementados de manera rápida y eficiente. Esto disminuye las líneas de código y permite que los programadores se enfoquen en crear las funcionalidades más necesarias.

## 8. Resultados y discusión

Como resultados hemos obtenido grandes avances con algo de diseño un poco diferente que funciona bastante bien, con algunos puntos por completar y solucionar. Tomará un poco más de tiempo tenerlo listo por lo que se hace la presentación en gráficos un poco inconclusos.

Inicio de sesión y registro listos, es la parte más fundamental de la aplicación web.





## 9. Sprint 0. Configuración del ambiente de desarrollo

Tareas:

- Diseño y creación de la base de datos.
- Creación del proyecto Next JS
- Creación del proyecto en Laravel.
- Usuarios

