

**FACULTAD DE INGENIERIA**

**ESCUELA DE COMPUTACIÓN**

**Asignatura: Diseño y Programación de Software Multiplataforma**

**CICLO ACADEMICO: II - 2024**

Título:

Fase I del Proyecto de Cátedra

Docente:

Ing. Karens Medrano

Presentado por:

Apellidos, Nombres	Carné
Aguilar Linares, José Carlos	AL210841
Campos Monterroza, Carlos Fernando	CM210153
Cruz Reyes, Christopher Amílcar	CR211136
Rivera Mendoza, Mónica Lissette	RM210852
Velasco Flores, Luis Pablo	VF202313

Grupo Teórico: 01T

Campus Soyapango, 31 de agosto de 2024

# Índice

Introducción .....	1
Perfil del Proyecto .....	2
Nombre del Proyecto .....	2
Problema a Resolver .....	2
Objetivo General.....	2
Metas .....	2
Resultados Esperados .....	2
Metodología.....	2
Área Geográfica y Beneficiarios del Proyecto.....	3
Factores Críticos de Éxito.....	3
Carta de Presentación .....	4
Presentación del diseño UX/UI .....	5
Paleta de Colores: .....	5
Iconografía: .....	5
Tipografía: .....	7
Imágenes:.....	7
Navegación Web / Móvil: .....	7
Diseño de Pantallas:.....	8
Mockups: .....	8
Explicación de la lógica de solución .....	9
Diagrama del Diseño de Arquitectura .....	11
Lógica de Negocio: .....	11
Bases de Datos: .....	12
Comunicación entre Componentes:.....	12
Seguridad: .....	13
Gestión de Sesiones y Estado:.....	14
Gestión de Configuración: .....	14
Herramientas de Integración Continua y Despliegue Continuo: .....	14
Herramientas a utilizar durante el desarrollo .....	16
Visual Code: .....	16

Figma .....	16
Trello .....	16
Android Studio .....	17
Presupuesto del costo de aplicación .....	18
Cronograma del trabajo .....	19
Fuentes de Consulta .....	23

## Introducción

Actualmente, los negocios se enfrentan al desafío de adaptarse a las crecientes demandas de sus clientes, quienes buscan una experiencia más accesible y personalizada. Con el auge de la tecnología móvil, desarrollar una aplicación para estos dispositivos se ha convertido en una estrategia clave para asegurar el éxito en las operaciones comerciales, donde el negocio pueda ofrecer sus productos, y los clientes puedan consumir estos productos.

Este proyecto está centrado en el diseño y desarrollo de una aplicación móvil para la gestión de un cine, que permita a sus usuarios: consultar cartelera y comprar entradas para las funciones de su preferencia. Además, proporcionará a los administradores del cine, herramientas para gestionar operaciones diarias como la programación de películas en sus diferentes salas.

El proyecto proporcionará una solución tecnológica innovadora que facilite la experiencia de sus usuarios, ya sea empleados del cine o sus clientes, contribuyendo así a brindar un servicio más ágil y satisfactorio para ambas partes.

# Perfil del Proyecto

## **Nombre del Proyecto**

Aplicación multiplataforma para la reserva, compra y gestión de los boletos a las funciones disponibles de un cine.

## **Problema a Resolver**

En la presente era digital, los negocios deben adaptarse a las tecnologías existentes para no quedar atrás. Cualquier empresa que se considere formal debe tener un sitio web moderno de calidad, con la mejor experiencia e interfaz de usuario disponible. Las franquicias de cines no se quedan atrás, ya que los clientes esperan poder visualizar la información de las películas que se están proyectando en el presente, agendar los horarios según su preferencia, y ordenar en línea. Si el cine en cuestión no tiene un sitio web, o bien, la página no cumple sus expectativas, entonces su competencia tendrá mejor ventaja. Por el contrario, si el sitio web es llamativo, de fácil uso e intuitivo, el usuario se verá más motivado a elegir ese cine.

## **Objetivo General**

Crear un sistema multiplataforma para un cine ficticio que permita la gestión de la cartelera disponible y la compra de boletos por parte del usuario.

## **Metas**

- Desplegar una aplicación para los dispositivos Android que permita a los clientes del cine comprar boletos para la película de su elección, según las disponibilidades de horarios e idiomas que haya en cada sucursal, reservando los asientos que ellos gusten.
- Permitirles a los administradores añadir películas a la cartelera, asociarlas a salas del cine y a horarios disponibles, de acuerdo con cada sucursal del cine.

## **Resultados Esperados**

- Mayor accesibilidad al realizar la compra de boletos para funciones de cine
- Obtener información de relevancia más fácilmente como películas en cartelera u ofertas.
- Crear una aplicación sencilla e intuitiva al momento de hacer uso de ella.

## **Metodología**

Para desarrollar una página web y una aplicación móvil para la reserva, compra y gestión de boletos de cine utilizando SCRUM, el proyecto se inicia con la definición de objetivos y alcance, seguido por la creación de un Product Backlog con historias de usuario prioritarias, como autenticación, visualización de funciones, selección de asientos, procesamiento de pagos, gestión de boletos y notificaciones. El equipo, compuesto por Product Owner, Scrum Master y desarrolladores, planifica y ejecuta

sprints cortos (2-4 semanas), realizando reuniones semanales (Daily Stand-ups), revisiones de sprint (Sprint Review) y retrospectivas (Sprint Retrospective) para mostrar el progreso, recibir feedback y mejorar continuamente hasta completar el proyecto.

### **Área Geográfica y Beneficiarios del Proyecto**

La cadena de distribuciones multimedia posee dos sucursales ubicadas en el departamento de San Salvador, municipio de San Salvador.

Los trabajadores, clientes y personal administrativo del cine se verían beneficiados con la implementación del sistema de gestión y venta de boletos, ya que se podrá llevar un recuento de las compras de boletos realizadas, los asientos que ya han sido reservados por cada sala, se podrá conocer que películas tienen más demanda, se llevara un mejor control de las funciones a programar y de su saturación de personas, entre otros. De manera que se agilizara y se facilite el proceso de compra/venta de boletos del establecimiento.

### **Factores Críticos de Éxito**

Posibles riesgos:

1. El cine no proporciona y /o tiene en primera instancia la información solicitada para el desarrollo de los mockups

Solución: Dado que la información solicitada (imágenes de películas en cartelera, información los horarios de películas, diferentes tipos de boletos (vip, 3d, d-box), entre otros.) no es vital para comenzar el desarrollo de la página web, se esperará a que la empresa recolecte la información y se la otorgue al equipo de desarrollo antes del lanzamiento de la página web

2. A lo largo del desarrollo del sitio web puede haber fallas técnicas en el equipo utilizado.

Solución: Realizar revisiones periódicas a los equipos utilizados por los desarrolladores, para asegurarse de que estos estén en el estado óptimo para su utilización para tareas de alto rendimiento.

3. Los programadores no están completamente capacitados en el lenguaje de programación escogido para desarrollar el proyecto

Solución: Se planea primeramente llegar a un acuerdo sobre desarrollar la página web en un lenguaje con el que el equipo de desarrollo este familiarizado, en dado caso no se llegue a este acuerdo, se realizaran consultas mediante la documentación y material audiovisual con el propósito de fortalecer y aprender más sobre el lenguaje en específico.

4. A la hora de planificar los entregables no se toma en cuenta que podría haber retrasos en el desarrollo del proyecto

Solución: Se planificará cronograma de manera que se dejen plazos largos para aquellas actividades que requieren más trabajo o sean más tediosas para los desarrolladores

5. El listado de todos los requisitos brindados, no son cubiertos y/o aplicados al 100% en su entrega final en los apartados de funcionalidad y diseño

Solución: Dado que existe la posibilidad de no cubrir todos los requisitos otorgados por el cliente, el scrum master verificará con especial atención que todos estos verdaderamente se cumplan, mediante el control y monitoreo continuo de las actividades efectuadas por los desarrolladores del proyecto, dado por seguro que, estas vayan de acuerdo con lo establecido.

### **Carta de Presentación**

Ninguna.





## Presentación del diseño UX/UI

### Paleta de Colores:

Para darle una buena imagen a las interfaces del proyecto, se utilizará la siguiente paleta de colores:

Color	HEX	RGB	Uso
	#820f0f	rgb(130, 15, 15)	Primario
	#b50c0c	rgb(181, 12, 12)	Titulos primario
	#383232	rgb(56, 50, 50)	Fondo oscuro
	#d9d9d9	rgb(217, 217, 217)	Fondo claro
	#7a7a7a	rgb(122, 122, 122)	Color secundario
	#d23a3a	rgb(210, 58, 58)	Botón primario
	#dd4040	rgb(221, 64, 64)	Boton eliminar

### Iconografía:

Icono	Uso
	Logo de marca sobre la cual se realiza el proyecto
	Icono que contendrá el botón de eliminar
	Icono que contendrá el botón para editar u modificar
	Icono para el menú desplegado



			Icono con el botón para el menú desplegable
			Icono que indica que es la ventana para elegir la película
			Icono para indicar que el usuario está en la selección de sala
			Icono que indica que el usuario está en la ventana para realizar el pago de las entradas
			Icono que indica el input para la forma de pago
			Icono que acompaña al nombre de la sucursal deseada
			Icono que indica la fecha y hora de la función seleccionada
			Icono que indica el audio de la película seleccionada

### **Tipografía:**

Tipo de Letra	Tamaño	Estilo de fuente	Uso
Anton	48px	Bold	Header 1
Anton	36px	Bold	Header 2
PT Serif	24px	Regular	Parrafo
Body	22px	Regular	Título película

### **Imágenes:**

Las imágenes que tendrá el sitio web son los carteles de las películas:

- Para la sección de cartelera tendrán las siguientes dimensiones: 172 x 217
- Para la sección de ofertas tendremos imágenes con 213px de altura
- En la sección para ver el horario de función, tendrán las dimensiones: 152 x 266
- En la sección de comprar entrada tendrán las dimensiones de: 173 x 243

### **Navegación Web / Móvil:**

Para este proyecto presentaremos un modelo de navegación jerárquico, ya que el software contará con dos roles, uno de usuario y otro de administrados, los cuales tendrán vistas que no compartirán con el otro rol especificado. Entre las vistas que tendrá cada rol serían:

Rol de usuario:

- Inicio
- Cartelera
- Dulcería
- Registrarse
- Iniciar sesión
- Comprar entradas

Rol de administrador:

- Añadir Película
- Añadir Sala
- Modificar función
- Modificar Película
- Modificar Sala
- Modificar función

## Diseño de Pantallas:

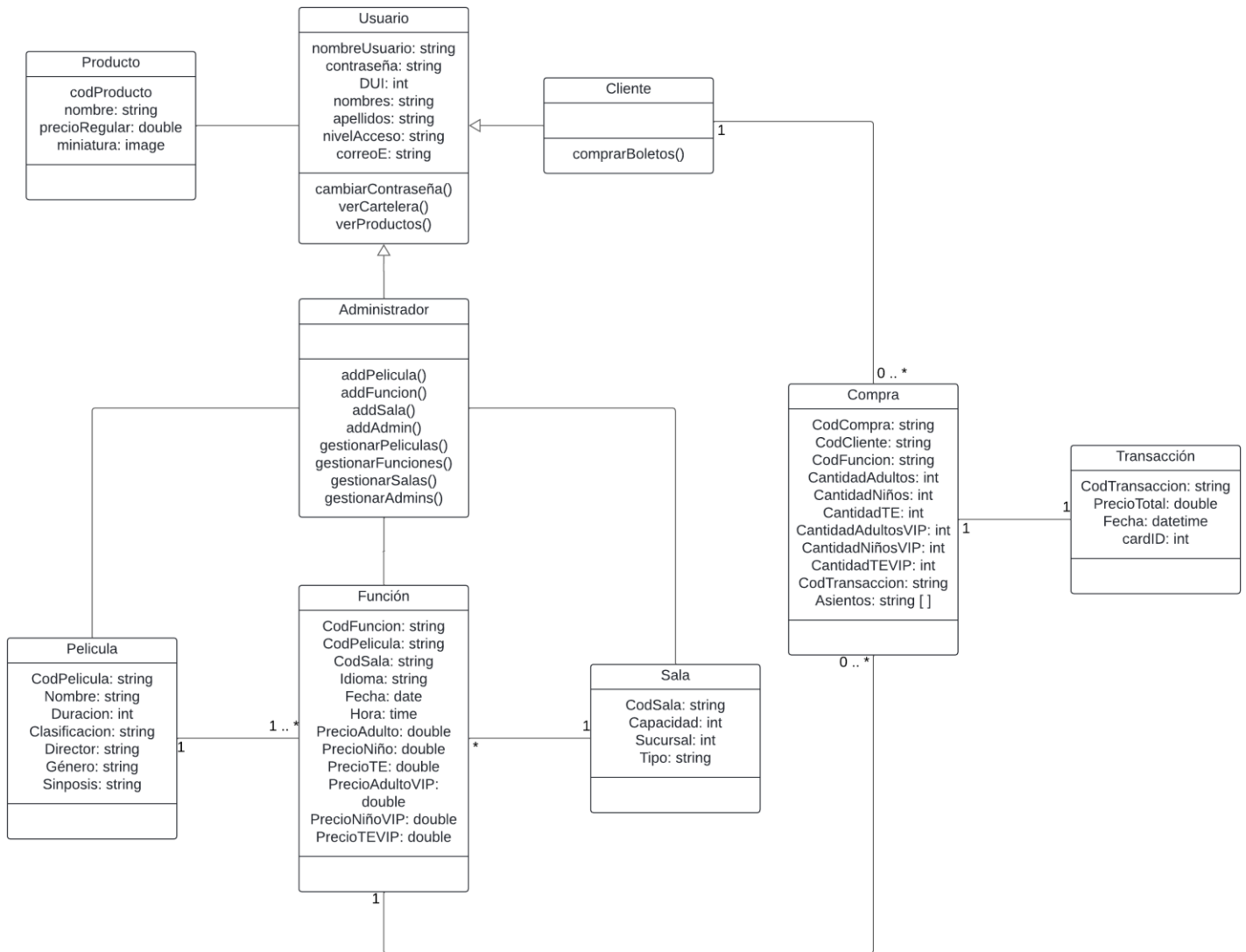


Para el diseño de la interfaz principal tenemos que en la parte superior la barra de navegación en la cual contiene el logo de la empresa junto con un botón que desplegará el menú de navegación. En la parte inferior podemos encontrar un footer en el cual contiene la información y manera de contactar con la empresa, además de las redes sociales de la misma.

## Mockups:

<https://www.figma.com/design/8Kqi38FRhuuhb65F0PPwoj/Cine?node-id=0-1&t=c1Jg2QtYDbdC2W37-1>

## Explicación de la lógica de solución



En el diagrama de clases se aprecia a grandes rasgos el funcionamiento de la aplicación. La cartelera del cine se podrá visualizar tan pronto se abra la app, al igual que el menú con todos los productos que se venden en la dulcería; pero para realizar otras acciones el usuario deberá iniciar sesión con su contraseña (la cual podrá cambiar en caso de que la olvide). Dependiendo de su nivel de acceso visualizará la interfaz de administrador o la de cliente regular.

Los administradores podrán añadir al sistema:

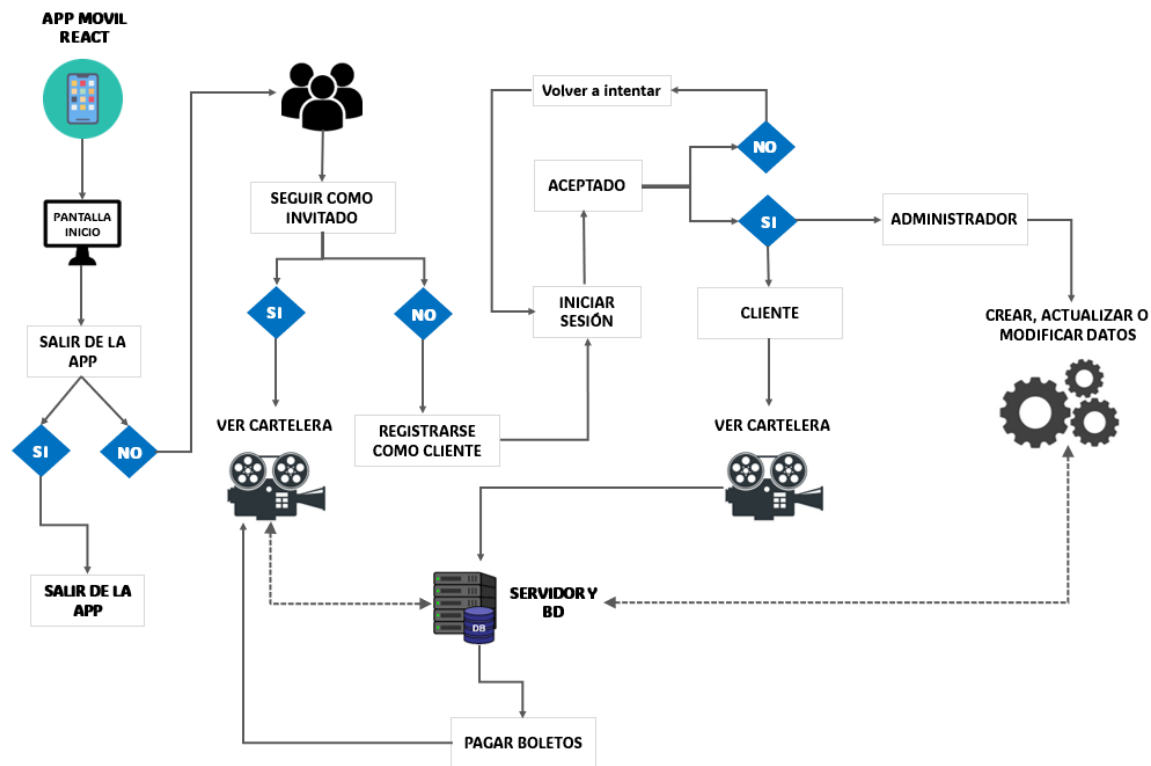
- Películas, con la información respectiva como nombre, duración, clasificación, género, director y sinopsis.

- b) Salas, con su capacidad de asientos, sucursal del cine en que se encuentra, tipo de pantalla (normal, 3D, etc).
- c) Funciones, en las que se deberá detallar en que sala se estará proyectando una película determinada, con la información respectiva de si será doblada o subtitulada, la fecha y hora de la función, y el precio de las entradas para adultos, niños y 3° edad (si es que aplica).
- d) También podrán añadir nuevos administradores. Es decir, para evitar que una persona externa se auto añada como administrador, solo un administrador puede añadir a otro administrador.

De igual forma podrán realizar editar y eliminar lo que hayan añadido.

Por otra parte, los clientes tienen la capacidad de comprar boletos según las películas disponibles en la cartelera, es decir, aquellas que tienen funciones asociadas. Elegirán la cantidad de entradas a comprar, ya sea para adultos, 3° edad o para niños (o las modalidades VIP cuando aplique), así como los asientos en los que verán la película. Por último, harán el pago usando una tarjeta de crédito, descontándoles el precio total. La transacción les generará un comprobante del pago para facilitar su entrada al cine.

# Diagrama del Diseño de Arquitectura



## Lógica de Negocio:

- Servidores:

Durante la creación de la aplicación, se optará por usar **XAMPP** en el desarrollo del backend que dará soporte al sitio. XAMPP es “un servidor web local multiplataforma que permite la creación y prueba de páginas web u otros elementos de programación” (Jesús, 2022). La ventaja de usar XAMPP es que integra todo lo necesario para el desarrollo de un sitio web, como bases de datos y soporte de PHP; y al trabajar de forma local, no es necesario que la computadora esté conectada a Internet (Jesús, 2022).

- **Lenguajes de Programación:**

El lenguaje de programación a emplearse en el desarrollo frontend será **JavaScript**, haciendo uso de su biblioteca **React** y el framework especializado para crear interfaces para aplicaciones móviles, **React Native**. React Native, es recomendado cuando un proyecto no requiere interfaces complicadas, cuando se desea desarrollar para una sola plataforma (en nuestro caso Android), y más importante, cuando el presupuesto es limitado. (Budziński, 2024). Para el desarrollo backend,

se utilizará **PHP**, específicamente su framework **Laravel**, para generar y responder las diferentes peticiones que la aplicación realice a la base de datos del sistema.

- Frameworks Backend:

**Laravel** es el framework que se usará para la creación del proyecto. De acuerdo con Axarnet:

Laravel crea un entorno de trabajo y proporciona herramientas a los desarrolladores para ayudarles a desarrollar en PHP sus aplicaciones web. Lo que se busca con Laravel es construir aplicaciones sólidas y estables, que sean fáciles de desarrollar y la utilización de parte del código preprogramada, para que pueda aprovecharse y reutilizarse, evitando así la reescritura del código en la misma aplicación (2023).

Este framework cuenta con una extensa documentación, además de librerías y paquetes disponibles. Es fácil de utilizar y brinda seguridad, ya que ofrece “una defensa muy alta frente a las amenazas más comunes, como el malware o las inyecciones de SQL maliciosas” (Axarnet, 2023).

### **Bases de Datos:**

- Sistema de Gestión de Bases de Datos:

Para este propósito se usará **MySQL**, el cual “es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto” (Robledano, 2023), aunque también posee una versión de paga manejada por la empresa Oracle. MySQL posee una arquitectura de cliente y servidor, completa compatibilidad con SQL, la capacidad de crear vistas personalizadas, entre otras características.

- Modelos de Datos:

Uno de los motivos de usar MySQL es utilizar el modelo **Entidad-Relación** dentro de la Base de Datos de la página. Este enfoque consiste en almacenar los atributos de las distintas entidades en tablas relacionadas entre sí por llaves primarias y secundarias.

### **Comunicación entre Componentes:**

- API:

Se desarrollará una aplicación en Laravel, para comunicar la base de datos con la aplicación. Esta API se encargará de dar respuesta a las peticiones que los usuarios realicen en las interfaces de la aplicación móvil.

- Protocolos de Comunicación

Como protocolo de comunicación se usará **HTTP/HTTPS**. Por una parte, HTTP “es el método principal por el cual los datos de las páginas web se transfieren a través de una red”. Sin HTTP, “la World Wide Web (WWW) no existirá tal como la conocemos”. No obstante, la información que se transmite por HTTP no es privada y es susceptible a ser interceptada por atacantes externos (Harnish, 2021).

HTTPS viene a ser la opción “segura” de HTTP, y ya que la página manejará datos personales de los clientes, por ejemplo, tarjetas de crédito, la ciberseguridad no se puede subestimar.

HTTPS utiliza el certificado de un proveedor externo para asegurar, y proteger, una conexión y así verificar que el sitio sea legítimo. Este certificado de seguridad se conoce como certificado SSL [...] Esto crea una conexión segura y cifrada entre un navegador y un servidor, y protege la capa de comunicación entre ambos. Este certificado cifra una conexión con un nivel de protección designado en el momento de la compra del certificado SSL. (Harnish, 2021)

### **Seguridad:**

- Autenticación y Autorización:

La aplicación será accesible para cualquier persona que la instale, obteniendo funciones básicas como consultar cartelera, entre otras. Sin embargo, para realizar funciones como la compra de boletos o la gestión de la cartelera, la aplicación requerirá inicio de sesión para identificar el tipo de usuario y mostrar las funciones ofrecidas según el tipo de usuario autenticado.

- Seguridad de Datos:

Laravel es robusto en cuanto a seguridad se refiere, ya que tiene “una defensa muy alta frente a las amenazas más comunes, como el malware o las inyecciones de SQL maliciosas” (Axarnet, 2023).

El tipo de encriptación a ocupar es **encriptación SHA256**. Esta es una función de tipo HASH, ocupadas ampliamente en la tecnología blockchain. Estas funcionan gracias a procesos matemáticos y lógicos, con los que “se busca obtener una cadena de caracteres de longitud fija y única para los datos introducidos, a la vez que se hace prácticamente imposible realizar el proceso contrario” (Academy, 2023): la creación de hashes es un proceso de un solo sentido (Academy, 2023).

“SHA-256 es uno de los algoritmos de hashing más utilizados debido a su equilibrio entre seguridad y costo computacional de generación, ya que es un método muy eficiente con una alta resistencia a la colisión”, que siempre genera una cadena de 64 letras y dígitos, codificada como 256 bits, o sea, 32 bytes (Learn, 2022).



### **Gestión de Sesiones y Estado:**

- Cookies y Tokens de Sesión:

Se utilizarán para almacenar las preferencias de los usuarios y personalizar la experiencia del usuario cada vez que utiliza la aplicación, por ejemplo, las cookies pueden almacenar temporalmente información de las compras hasta que el usuario complete la compra. Las cookies pueden usarse para rastrear el comportamiento del usuario dentro de la aplicación, como las películas vistas o las secciones más visitadas. Estos datos pueden luego analizarse para mejorar la experiencia del usuario o para personalizar ofertas.

- Gestión del Estado del Cliente:

Consideramos que la aplicación tiene un flujo de datos simple, por lo que no se necesitará la gestión del estado del cliente. El uso de `useState` y `useContext` en React es suficiente y más sencillo de implementar para cumplir los propósitos del proyecto.

### **Gestión de Configuración:**

- Configuración del Entorno:

La configuración de las variables de entorno se realizará mediante **archivos .env**. Estos son archivos de texto plano que almacenan las variables de entorno (como contraseñas de bases de datos u otra información sensible) usando pares de "llave=valor". Este tipo de archivos deben ser excluidos de los repositorios como GitHub, y facilitan el trabajo en equipo (Kogan, 2022).

- Automatización de Implementación

La automatización de implementación se hace mediante contenedores, los cuales son entornos de ejecución ligeros que proporcionan los recursos y componentes necesarios para que una aplicación funcione, sin llegar a ser tan pesados como una máquina virtual (DataScientest, 2023).

Para el proyecto, se ocupará **Docker**, que es la plataforma de contenedores más utilizada: es de código abierto, segura y con una gran comunidad de respaldo.

### **Herramientas de Integración Continua y Despliegue Continuo:**

- Automatización de despliegue:

La integración continua se refiere a la integración de mejoras en el código de un proyecto una vez han sido validadas, con el propósito de detectar errores lo antes posible; este proceso suele ocurrir varias veces al día (Sentry, 2022).

Para realizar la integración continua, se usará **Jenkins**, el cual es un servidor open source, multiplataforma y de acceso libre que facilita el testeo de los proyectos: cada commit realizado es verificado mediante la creación de un build, y si este falla, se le notifica al equipo, pero si funciona exitosamente, se despliega en el servidor de testeo (Sentry, 2022).

- Controlador de Versiones:

Como controlador de versiones se usará **Git**, y en particular, **GitHub**, que funciona como un repositorio donde los usuarios pueden trabajar en conjunto mediante la creación de ramas (branches) y la realización de commits, de manera que se pueda hacer un seguimiento del código y de como se ha ido expandiendo en todo momento.

# Herramientas a utilizar durante el desarrollo

## **Visual Code:**

Visual Code (también conocido como VsCode), es un editor de código multiplataforma desarrollado por Microsoft, fácilmente adaptable a cualquier lenguaje de programación (De Zúñiga, 2024). No es una IDE estrictamente hablando, ya que no permite compilar el código, pero añadiendo extensiones puede asemejarse bastante a una (De Zúñiga, 2024).

Comúnmente es empleado para la programación Frontend, pero también se adapta a lenguajes como PHP y Java. También se integra fácilmente con los repositorios Git. Es un editor con una amplia biblioteca de extensiones que permiten dar formato al código, depurar, hacer pruebas, entre muchas otras funciones (De Zúñiga, 2024).

Para el desarrollo del proyecto, Visual Code será el editor de texto principal con el cual se creará el sitio web.

## **Figma**

Figma es una plataforma que sirve para diseñar interfaces de páginas web, aunque también permite crear presentaciones y publicaciones para redes sociales. Es una herramienta online que permite el trabajo colaborativo y en tiempo real (Blandino, 2023), una característica que la hace muy adecuada para el trabajo remoto que muchas veces está asociado al desarrollo de software. Existe una versión básica gratis, y también hay suscripciones para acceder a todas las funcionalidades (Blandino, 2023).

Figma facilita la creación de los mockups, por lo que se ha usado para ese propósito en este proyecto.

## **Trello**

Trello es un software que permite la organización de actividades de un proyecto, mediante tarjetas, etiquetas, checklists y elementos similares:

El objetivo principal de Trello puede resumirse en una única tarea: proveer una herramienta para la organización de la información. Esta plataforma está optimizada para hacer más sencillas las tareas de los equipos de trabajo. Mediante una interfaz interactiva, los usuarios pueden acceder a información ordenada sobre un proyecto, un plan de trabajo o una meta a cumplir. (Terrerros, 2023)

Trello se convierte en una ayuda visual para revisar el avance del desarrollo de la página web que se está diseñando, por lo que en un tablero de Trello se ha colocado

el resumen de todas las actividades que son necesarias para dar cumplimiento a los objetivos, junto con el Product Backlog del sistema.

### **Android Studio**

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Ofrece un dispositivo virtual de Android que se utilizará para ejecutar y probar la aplicación web a desarrollar.

## Presupuesto del costo de aplicación

Presupuesto						
	Quincena 1	Quincena 2	Semana 5	Cantidad	Costo Unitario	Costo Total
<b>Mano de obra</b>						
Director del proyecto	80	80	40	200	\$7.00	\$1400.00
Diseñador	80	80	40	200	\$5.00	\$1,000.00
Desarrollador Back-End	80	80	40	200	\$6.00	\$1,200.00
Programador	80	80	40	200	\$5.00	\$1,000.00
Desarrollador de Software Full Stack	80	80	40	200	\$8.00	\$1,600.00
<b>Hardware</b>						
Depreciación por el uso informatico de computadoras	320	320	160	800	\$0.35	\$280.00
<b>Software</b>						
Software de uso libre	0	0	0		0	0
<b>Servicios</b>						
Energia electrica (kw/H)	230.4	230.4	144	604.8	\$0.15	\$90.72
Internet (plan)	1	1	1	3	\$30.00	\$120.00
					<b>Subtotal</b>	\$6,660.70
					15% imprevistos	\$999.105
					25% ganancias	\$1,665.175
					13% IVA	\$865.90
					<b>TOTAL</b>	\$8,459.08

## Cronograma del trabajo

Semanas							
Primera Fase			Segunda Fase				
Del 12 al 17 de agosto de 2024	Del 19 al 24 de agosto de 2024	Del 26 al 31 de agosto de 2024	Del 2 al 7 de septiembre de 2024	Del 9 al 14 de septiembre de 2024	Del 16 al 21 de septiembre de 2024	Del 23 al 28 de septiembre de 2024	Del 30 de septiembre al 5 de octubre de 2024
Realización del backlog							
Definición de vistas mediante la creación de mockups							
Selección de tipografías, íconos, paletas de colores, etc.							
Creación del diagrama UML de clases							
Selección de todos los componentes de software							
	Determinar otras herramientas a usar en el proyecto						

	Elaboración del presupuesto						
Elaboración del cronograma							
Creación del tablero en Trello							
	Creación del repositorio en GitHub						
	Selección de licencia creative commons						
			Automatización de despliegue con Jenkins				
			Automatización de implementación con Docker				
			Creación de todas las interfaces				
			Diseño y creación de la BD				

				Desarrollo del login y registro de clientes			
				Desarrollo del restablecimiento de contraseña			
				Creación de la ventana principal			
				Desarrollo del registro de nuevos administradores			
					Desarrollo de procedimientos para añadir nuevas películas		
					Desarrollo de procedimientos para añadir nuevas salas		
					Desarrollo de procedimientos para añadir nuevas funciones		



					Modificación de todos los registros añadidos		
						Visualización de cartelera y funciones proyectadas	
						Compra de boletos por cantidad y tipo de persona	
						Selección de asientos para la función	
						Cálculo automatizado del precio total	
							Desarrollo de la plataforma de pago
							Generación del comprobante de compra

## Fuentes de Consulta

Academy, B. (2023, 3 agosto). ¿Qué es un hash? *Bit2Me Academy*.

<https://academy.bit2me.com/que-es-hash/>

Axarnet. (2023, 23 febrero). *Laravel el Framework de PHP - Qué es y para qué*

*sirve*. <https://axarnet.es/blog/que-es-laravel>

Blandino, G. (2023, 4 enero). *Figma: qué es y cómo funciona* |. El Blog de

Pixartprinting. <https://www.pixartprinting.es/blog/figma-que-es/>

Budziński, M. (2024). *What Is React Native? Complex Guide for 2024*. Netguru.

<https://www.netguru.com/glossary/react-native>

DataScientest. (2023, 30 octubre). *Docker: ¿qué es y cómo se usa?* Formación En

Ciencia de Datos | DataScientest.com. <https://datascientest.com/es/docker-todo-que-saber>

De Zúñiga, F. G. (2024, 21 febrero). *¿Qué es Visual Studio Code y cuáles son sus*

*ventajas?* Blog de arsys.es. <https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas>

Harnish, B. (2021, 2 febrero). *Qué es HTTPS: la guía definitiva para entender*

*cómo funciona*. Semrush Blog. <https://es.semrush.com/blog/que-es-https/#header2>

Jesús. (2022, 18 noviembre). *Conoce qué es Xampp y por qué deberías usarlo en*

*tus proyectos*. Tutoriales Dongee. <https://www.dongee.com/tutoriales/que-es-xampp/>

Kogan, D. (2022, 11 diciembre). Why a .env? - Dev Genius. *Medium*.

<https://blog.devgenius.io/why-a-env-7b4a79ba689>

Learn, B. (2022, 2 junio). *¿Qué es el algoritmo SHA-256?* Bybit Learn.

<https://learn.bybit.com/es/blockchain/que-es-el-algoritmo-sha-256/>

Robledano, A. (2023, 13 abril). Qué es MySQL: Características y ventajas.

*OpenWebinars.net*. <https://openwebinars.net/blog/que-es-mysql/>

Sentrio. (2022, 23 marzo). *Introducción a Jenkins: ¿qué es, para qué sirve y cómo*

*funciona?* Sentrio. <https://sentrio.io/blog/que-es-jenkins/>

Terreros, D. (2023, 20 enero). *Qué es Trello, para qué sirve y cómo funciona.*

HubSpot. <https://blog.hubspot.es/marketing/que-es-trello>