



**UNIVERSIDAD DON BOSCO**

**FACULTAD DE INGENIERÍA**

**ASIGNATURA:**

DISEÑO Y PROGRAMACIÓN DE SOFTWARE MULTIPLATAFORMA

**GRUPO:**

02T

**ACTIVIDAD:**

PRIMERA FASE PROYECTO.

**DOCENTE:**

ING. ALEXANDER ALBERTO SIGUENZA CAMPOS

**INTEGRANTES:**

**NOMBRE**

**CARNÉ**

ALEJANDRA DEL CARMEN PONCE LÓPEZ

PL210665

DAVID ELEAZAR GUZMÁN GUARDADO

GG171680

MARÍA JOSÉ TORRES MOLINA

TM210066

LEÓN MARROQUÍN, RONALD VLADIMIR

LM192213

OSWALDO ENRIQUE HERNÁNDEZ TORRES

HT192210

CARLOS ALBERTO AMAYA CÁRCAMO

AC220139

## Índice

Introducción.....	3
Perfil de proyecto .....	4
Nombre .....	4
Problema por resolver.....	4
Objetivo general .....	5
Metas .....	5
Resultados esperados .....	6
Metodología .....	8
Área geográfica y beneficiarios del proyecto.....	9
Posibles Fuentes de Financiamiento.....	10
Factores de éxito .....	11
Carta de presentación del proyecto .....	12
Presentación del diseño UX/UI.....	12
Lógica y diagramas UML .....	17
Diagrama gráfico del diseño de la arquitectura del proyecto de software .....	24
Herramientas .....	25
Cronograma de trabajo .....	30
Presupuesto .....	30

## Introducción

Mostrar un emocionante proyecto de programación diseñado y centrado en el desarrollo de una aplicación móvil y web para facilitar el proceso de pedidos y entregas de alimentos en restaurantes. La aplicación, basada en la versátil plataforma React Native, ofrece una experiencia mejorada tanto para los clientes que desean realizar pedidos como para los propietarios de restaurantes que desean administrar de manera óptima sus operaciones.

La aplicación se organiza en torno a diversas vistas y funcionalidades diseñadas para simplificar la interacción de los usuarios con los restaurantes. Los clientes pueden explorar restaurantes (en caso de ser múltiples) o el restaurante en caso de ser uno solo, ver el respectivo menú, personalizar órdenes y realizar un seguimiento de sus pedidos.

Por otro lado, siendo uno o múltiples restaurantes estos podrán gestionar sus menús, supervisar pedidos y actualizar la información de su establecimiento de manera sencilla. Se detalla las diferentes vistas y funciones disponibles en la aplicación, lo que proporciona una comprensión completa de su alcance y utilidad.

Para brindar una visión más completa del proyecto, se estiman los costos asociados con el desarrollo, pruebas, alojamiento, marketing y mantenimiento continuo. Esto nos brinda una vista de la planificación financiera que usaremos para la ejecución del proyecto.

Por último, el proyecto se presenta como una oportunidad para demostrar el potencial de React Native en el mundo del desarrollo móvil; nosotros usaremos un cronograma de actividades que abarca desde la definición y diseño iniciales hasta la implementación, pruebas y lanzamiento, dando así una guía clara para el desarrollo del proyecto.

## Perfil de proyecto

### Nombre

HappyPedidos

### Problema por resolver

Este emocionante proyecto se centra en la creación de una innovadora aplicación móvil para la gestión de pedidos y entregas de comida en restaurantes. Utilizando la versátil plataforma React Native, nos proponemos desarrollar una solución integral que transformará la forma en que los usuarios interactúan con sus restaurantes favoritos. La aplicación permitirá a los usuarios no sólo realizar pedidos de sus platillos preferidos de manera rápida y sencilla, sino también acceder a información detallada sobre los menús, promociones y tiempos de entrega estimados.

Desde la perspectiva de los restaurantes, esta aplicación ofrecerá herramientas avanzadas para gestionar los pedidos de manera eficiente y mantener su oferta gastronómica actualizada. Los restaurantes podrán recibir y procesar pedidos en tiempo real, ajustar su menú según la disponibilidad de ingredientes, y ofrecer un servicio personalizado a sus clientes. Además, la plataforma incluirá funciones de análisis y reportes que ayudarán a los restaurantes a comprender mejor las preferencias de sus clientes y a optimizar sus operaciones diarias.

Un aspecto clave de esta aplicación es la diferenciación de roles y funcionalidades específicas para cada tipo de usuario. Los clientes tendrán una interfaz amigable y fácil de usar para explorar menús, realizar pedidos y seguir el estado de sus entregas. Por otro lado, los administradores de los restaurantes contarán con un panel de control completo para gestionar los pedidos, actualizar la información del menú y comunicarse con los clientes de manera eficiente.

Este proyecto demostrará claramente el inmenso potencial de React Native en el desarrollo de aplicaciones móviles. Gracias a su capacidad para crear aplicaciones multiplataforma con un solo código base, React Native permitirá una implementación rápida y eficiente, garantizando un rendimiento óptimo tanto en dispositivos iOS como Android. La flexibilidad y potencia de esta plataforma se reflejarán en una experiencia de usuario fluida y en la robustez de las herramientas de gestión para los restaurantes. En definitiva, este proyecto simplificará los pedidos y entregas de comida y elevará el nivel de servicio y satisfacción para usuarios y restaurantes.

## Objetivo general

Crear una nueva y eficiente aplicación móvil, utilizando la plataforma React native, para mejorar la gestión de pedidos y entrega de comida en restaurantes. Esta solución facilitará a los usuarios realizar pedidos, acceder a menús de promociones y estimar tiempos de entrega por medio del estado del pedido realizado, de igual forma proporcionará a los restaurantes herramientas para la gestión de pedidos en tiempo real, actualización de menús y análisis de datos.

El sitio se centrará en proporcionar una experiencia de usuario personalizada y de alta calidad, separando ciertas funciones y responsabilidades de los clientes y los gerentes de restaurantes, para mejorar el rendimiento y la satisfacción del cliente en el sector de los restaurantes con entrega a domicilio.

## Metas

- Desarrollo de la Aplicación Móvil:
  - Meta: Completar el desarrollo de la aplicación móvil en un plazo de 4 meses.
  - Verificable mediante: El seguimiento del cronograma de desarrollo y la finalización de cada fase del proyecto (diseño, desarrollo, pruebas y lanzamiento).
- Desempeño y Optimización:
  - Meta: Garantizar que la aplicación mantenga un tiempo de respuesta inferior a 2 segundos para todas las operaciones críticas.
  - Verificable mediante: Pruebas de rendimiento periódicas y monitoreo continuo de la aplicación.
- Satisfacción del Usuario:
  - Meta: Alcanzar una satisfacción del usuario de 4.5 sobre 5 en el tiempo de uso durante la fase beta de la aplicación.
  - Verificable mediante: Encuestas de satisfacción a los usuarios beta y la recopilación de comentarios detallado.
- Compatibilidad Multiplataforma:
  - Meta: Asegurar la compatibilidad y funcionamiento óptimo de la aplicación en al menos el 99.99% de los dispositivos iOS y Android disponibles en el mercado actual.

- Verificable mediante: Pruebas en una variedad de dispositivos y sistemas operativos, y la obtención de resultados positivos en más del 99.99% de los casos.

## **Resultados esperados**

### Ventajas y beneficios

- Experiencia de usuario mejorada
  - Los usuarios podrán realizar pedidos de comida de forma rápida y sencilla a través de una interfaz intuitiva y amigable.
  - Acceso a información detallada sobre menús, promociones y tiempos estimados de entrega, mejorando la satisfacción del cliente.
- Eficiencia operativa para los restaurantes
  - Herramientas avanzadas para que los restaurantes gestionen pedidos en tiempo real, ajusten menús según la disponibilidad de ingredientes y brinden un servicio personalizado.
  - Servicios de análisis y generación de informes que permitirán a los restaurantes comprender mejor las preferencias de los clientes y mejorar sus operaciones diarias.
- Multiplataforma
  - Gracias a React Native, la aplicación se utilizará de forma rápida y eficiente tanto en dispositivos iOS como Android, lo que garantiza un buen rendimiento.
- Comunicación Eficiente
  - Los gerentes de restaurantes tendrán un panel de control completo para gestionar pedidos, actualizar la información del menú y comunicarse con los clientes de manera más eficiente.

- Innovación
  - La separación de roles y funciones específicas para cada tipo de usuario (clientes y responsables de restaurante) permitirá una experiencia personalizada y eficiente para todos los usuarios de la aplicación.

#### Principales resultados esperados

- Incremento en la Satisfacción del Cliente
  - Mejora significativa en la experiencia de pedido y entrega de comida, lo que resultará en una mayor satisfacción y fidelización de los usuarios.
- Aumento en la Eficiencia de Gestión para Restaurantes
  - Restaurantes podrán gestionar pedidos y menús de manera más eficiente, reduciendo errores y optimizando tiempos de respuesta.
- Datos y Análisis para la Toma de Decisiones
  - Los restaurantes tendrán acceso a datos y análisis detallados sobre las preferencias de los clientes, permitiéndoles tomar decisiones informadas para mejorar sus servicios y ofertas.
- Mayor Alcance y Accesibilidad
  - La aplicación estará disponible tanto en iOS como en Android, ampliando el alcance y accesibilidad a una mayor base de usuarios.
- Reducción de Costos Operativos
  - Al optimizar la gestión de pedidos y la actualización de menús, los restaurantes podrán reducir costos operativos y mejorar la eficiencia general.
- Posicionamiento Competitivo
  - Los restaurantes que utilicen esta aplicación estarán mejor posicionados en el mercado al ofrecer una experiencia de usuario superior y un servicio más eficiente.

## Metodología

- Investigación y Análisis

Objetivo: Identificar competidores directos e indirectos, estudiar las necesidades y preferencias de los usuarios potenciales, evaluar las tendencias del mercado en el ámbito de las entregas a domicilio y compras en línea.

Actividades: Reunir y documentar los requisitos funcionales y no funcionales del proyecto, crear casos de uso detallados para todas las funcionalidades clave.

- Planificación del Proyecto

Objetivo: Diseñar la arquitectura de la aplicación frontend, backend, base de datos y seleccionar las tecnologías adecuadas como podrían ser React para frontend, Node.js para backend, y MongoDB para la base de datos.

Actividades: Definir el alcance del proyecto, crear un cronograma con hitos y fechas de entrega, asignar roles y responsabilidades dentro del equipo de desarrollo.

- Diseño de la Aplicación

Objetivo: Asegurar una navegación intuitiva con diseño UX y una experiencia de usuario fluida, implementar principios de diseño responsivo para garantizar la compatibilidad con diferentes dispositivos.

Actividades: Diseñar interfaz de Usuario (UI), crear wireframes y prototipos, desarrollar maquetas de alta fidelidad y realizar pruebas de usabilidad con usuarios reales.

- Desarrollo

Objetivo: Desarrollo del frontend y backend implementando las funcionalidades de los requisitos definidos, realizar pruebas unitarias y de integración continuamente.

Actividades: Configuración del Entorno de Desarrollo, repositorios de control de versiones (GitHub/GitLab), establecer un entorno de desarrollo local con todas las dependencias necesarias. Diseñar el esquema de la base de datos, implementar mecanismos de seguridad y copias de seguridad.

- Pruebas

Objetivo: Verificar que cada funcionalidad de la aplicación cumple con los requisitos especificados, realizar pruebas de aceptación con usuarios para validar el funcionamiento.

Actividades: Pruebas de rendimiento y seguridad bajo diferentes cargas de trabajo, implementar y verificar las medidas de seguridad para proteger los datos de los usuarios.

- Implementación y Despliegue

Objetivo: Configurar servidores de producción y sistemas de monitoreo, migrar la base de datos y los datos iniciales necesarios.



Actividades: Desplegar la aplicación en el entorno de producción, realizar pruebas finales para asegurar que todo funciona correctamente post despliegue.

- Mantenimiento y Mejora Continua

Objetivo: Planificar y desarrollar nuevas funcionalidades según las necesidades del mercado y el feedback de los usuarios, mantener la aplicación actualizada con las últimas tecnologías y estándares de seguridad.

Actividades: Monitorear el rendimiento de la aplicación y solucionar problemas en tiempo real, proveer soporte al cliente y gestionar feedback de los usuarios.

- Documentación y Presentación

Objetivo: Preparar una presentación visualmente atractiva que resuma los logros del proyecto, incluir el enlace al repositorio de git y cualquier otra documentación relevante.

Actividades: Documentar el código y los procesos de desarrollo, crear manuales de usuario y guías de administración.

Esta metodología asegura un enfoque estructurado y eficiente para el desarrollo del proyecto "HappyPedidos", facilitando la colaboración dentro del equipo y garantizando la entrega de un producto de alta calidad.

## **Área geográfica y beneficiarios del proyecto**

El proyecto se desarrollará en la ciudad de Ciudad de San Salvador, una de las metrópolis más grandes y dinámicas de América Latina. Con una población de aproximadamente 8 millones de habitantes, es el centro económico, político y cultural de El Salvador. La ciudad cuenta con una gran diversidad de opciones gastronómicas que van desde puestos de comida callejera hasta restaurantes de alta cocina.

Geográficamente, San Salvador está en el centro de El Salvador, a 2,240 metros sobre el nivel del mar. La ciudad se divide en 19 municipios, cada uno con identidad única. Entre las delegaciones relevantes para este proyecto están Soyapango, Apopa, Ilopango, Mejicanos, Tonacatepeque, conocidas por su alta concentración de restaurantes y comercios gastronómicos.

El clima de la ciudad es templado con variaciones estacionales moderadas, lo que permite la operación continua de actividades comerciales a lo largo del año. La infraestructura tecnológica y de telecomunicaciones de San Salvador es robusta, con una alta penetración de smartphones y acceso a internet, factores que facilitan la adopción de soluciones móviles como la aplicación que se desarrollará.

Se pretende beneficiar dos sectores en específicos:

- Usuarios finales
  - **Descripción:** Personas que residen en San Salvador y sus alrededores, así como visitantes y turistas que buscan opciones convenientes para ordenar comida.
  - **Beneficios:** Acceso a una plataforma intuitiva que les permite explorar menús, realizar pedidos y recibir sus alimentos de manera eficiente. Los usuarios podrán disfrutar de una experiencia de usuario mejorada y personalizada, con opciones de seguimiento de pedidos en tiempo real y notificaciones sobre ofertas y promociones.
- Repartidores
  - **Descripción:** Personas que se dedican al reparto de comida a domicilio, ya sea empleados directos de los restaurantes o trabajadores independientes.
  - **Beneficios:** Acceso a una plataforma que les permite gestionar sus entregas de manera más organizada, con rutas optimizadas y comunicaciones claras con los clientes y los restaurantes. Esto se traduce en un aumento potencial de ingresos y una mejora en sus condiciones laborales.

## Posibles Fuentes de Financiamiento

Para el desarrollo de nuestra plataforma “Happy Pedidos”, hemos asegurado un financiamiento clave que ha sido crucial para la implementación y crecimiento de nuestra propuesta. Hasta la fecha, hemos obtenido un total de \$12,800 en financiamiento, lo cual ha permitido avanzar significativamente en las etapas iniciales del proyecto.

- Financiamiento Principal:
  - **Inversión de TechWave Solutions:** TechWave Solutions, una empresa emergente de tecnología reconocida por su apoyo a proyectos innovadores ha decidido invertir \$7,500 en “Happy Pedidos”. Este financiamiento ha sido esencial para el desarrollo de la plataforma, cubriendo costos de diseño, programación y adquisición de herramientas tecnológicas clave.
- Fondos Complementarios:
  - **Fondo de Innovación Juvenil (FIJ):** Hemos recibido \$3,300 a través del Fondo de Innovación Juvenil, destinados a campañas de marketing y la implementación de un programa piloto en áreas seleccionadas.

- Apoyo Adicional:
  - Bancaria Local: Adicionalmente, un crédito de \$1,500 de una entidad bancaria local ha sido utilizado para la compra de equipos y el refinanciamiento de gastos operativos iniciales.
  - Subvención de Fundación Innovar: La Fundación Innovar ha otorgado \$1,500 en subvención para la capacitación del equipo y el desarrollo de nuevas funcionalidades en la plataforma.

## Factores de éxito

Los factores críticos de éxito para la aplicación HappyPedidos, son los siguientes:

- **Experiencia de Usuario (UX):** Una interfaz intuitiva y fácil de usar tanto para clientes como para administradores es fundamental para garantizar la adopción y el uso continuo de la aplicación.
- **Tiempo de Entrega:** Cumplir con el plazo de 4 meses es crucial para lanzar la aplicación al mercado y obtener una ventaja competitiva.
- **Calidad del Software:** La aplicación debe ser estable, segura y libre de errores para garantizar la satisfacción del cliente.
- **Integración con Sistemas Externos:** La integración con sistemas de pago, mapas y otros servicios externos debe ser fluida y confiable.
- **Escalabilidad:** La aplicación debe manejar un aumento significativo en el número de usuarios y pedidos.

## Riesgos Significativos y Mitigación

- Riesgo: Problemas de integración con sistemas externos.
  - Mitigación: Realizar pruebas de integración a lo largo del desarrollo, establecer puntos de contacto claros con los proveedores externos y tener un plan de contingencia en caso de problemas.
- Riesgo: Retrasos en el desarrollo.
  - Mitigación: Utilizar metodologías ágiles como Scrum para realizar entregas incrementales y ajustar el alcance del proyecto si es necesario. Monitorear de cerca el progreso y realizar ajustes en el plan si se detectan desviaciones.
- Riesgo: Problemas de seguridad.

- Mitigación: Implementar medidas de seguridad robustas desde el inicio del proyecto, realizar pruebas de penetración y cumplir con las regulaciones de seguridad de datos aplicables.
- Riesgo: Baja calidad del software.
- Mitigación: Establecer un proceso de control de calidad riguroso, realizar pruebas unitarias, de integración y de aceptación, y utilizar herramientas de automatización de pruebas.

## Carta de presentación del proyecto

### Presentación del diseño UX/UI

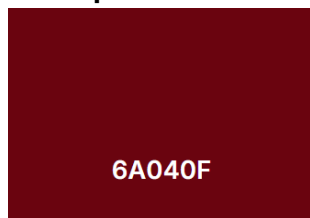
#### Paleta de colores

Para asegurar una experiencia agradable hacia el usuario, se ha seleccionado una paleta de colores:



*Figura 1. Paleta de colores*

- **Color primario:** Utilizado para los fondos y elementos principales.



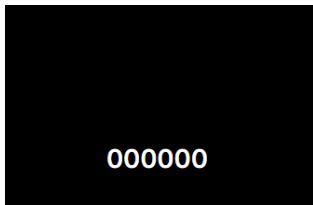
- **Color secundario:** Utilizado para los botones primarios de acción.



- **Color Terciario:** Utilizado para botones secundarios y áreas de menor énfasis.



- **Color de Texto:** Utilizado para todos los textos principales y secundarios.



## Iconografía

Para el presente proyecto, se utilizará la librería de iconos de React Native Vector Icons por su fácil implementación sobre React Native y su diseño fresco y amigable. React Native Vector Icons es una librería utilizada en el desarrollo de aplicaciones móviles con React Native que proporciona un conjunto de íconos personalizables. La librería incluye íconos populares de fuentes como FontAwesome, Material Icons, Ionicons, y muchas otras, permitiendo agregar íconos escalables y personalizables fácilmente en aplicaciones. La fuente de la iconografía es FontAwesome 5.











































AntDesign						
						
stepforward	stepbackward	forward	backward	caretright	caretleft	caredown
						
caretup	rightcircle	leftcircle	upcircle	downcircle	rightcircleo	leftcircleo
						
upcircleo	downcircleo	verticleleft	verticleright	back	retweet	shrink
						
arrowsalt	doubleright	doubleleft	arrowdown	arrowup	arrowright	arrowleft
						
down	up	right	left	minussquareo	minuscircle	minuscircleo
						
minus	pluscircleo	pluscircle	plus	infocircle	infocircleo	info

Figura 2. Iconos hechos con React

Iconos principales para la aplicación:

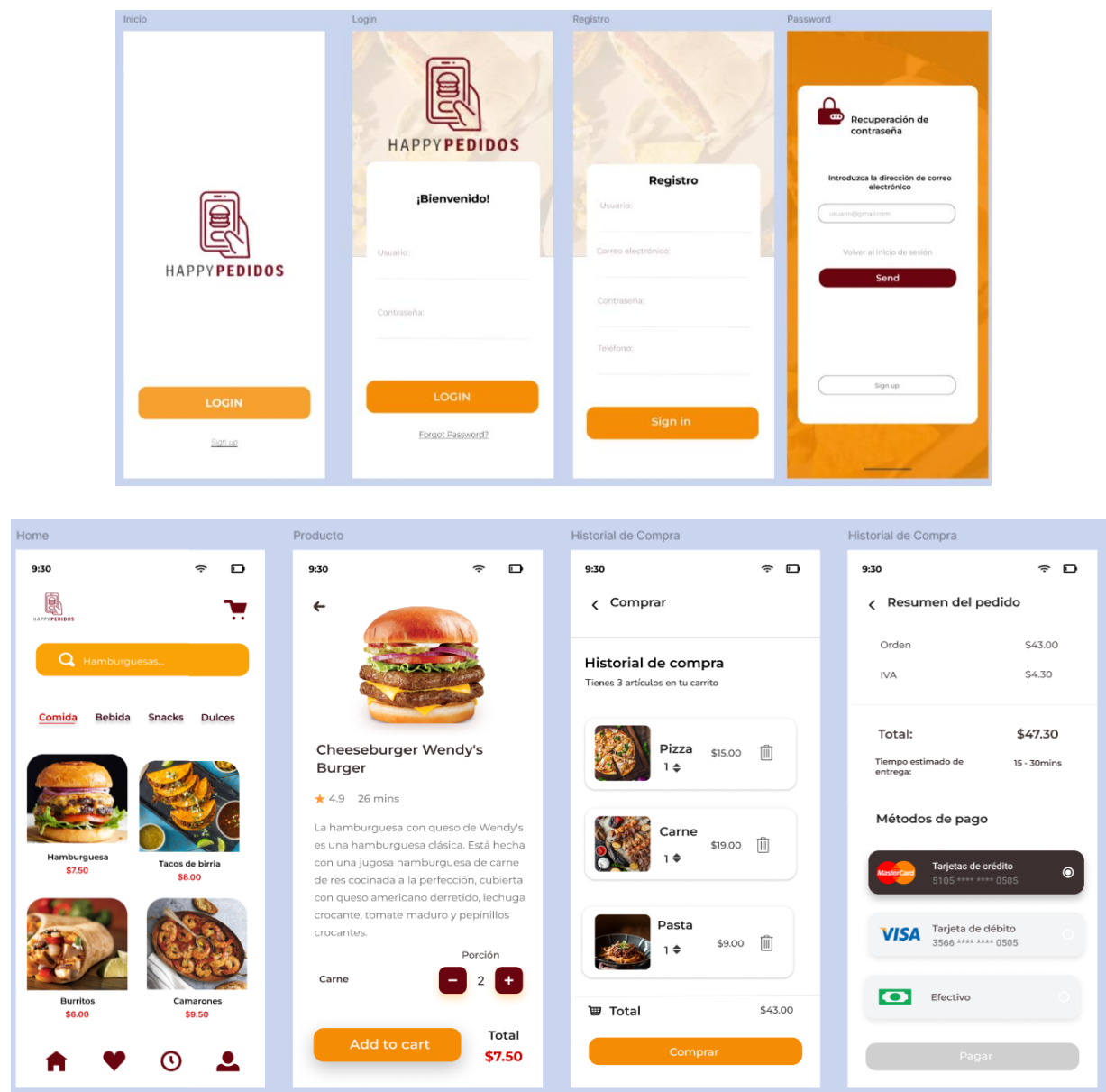
- Iconos de Navegación:
  - ✓ Inicio: Icono de casa para volver a la pantalla principal.
  - ✓ Explorar: Icono de lupa para buscar restaurantes, platos o categorías de alimentos.
  - ✓ Perfil: Icono de usuario para acceder a la configuración del perfil y métodos de pago.
  - ✓ Favoritos: Icono de corazón para acceder a la lista de restaurantes y platos favoritos.
- Iconos de Acción:
  - ✓ Añadir al Carrito: Icono de carrito de compras para agregar artículos al pedido.
  - ✓ Eliminar del Carrito: Icono de "X" o papelera para eliminar artículos del carrito.
  - ✓ Pagar: Icono de tarjeta de crédito o billetera para proceder con el pago.
- Iconos de Estado:
  - ✓ Preparando: Icono de chef o utensilios de cocina para indicar que el pedido está siendo preparado.
  - ✓ En Camino: Icono de motocicleta o coche para mostrar que el pedido está en camino.
  - ✓ Entregado: Icono de caja con una marca de verificación para indicar que el pedido ha sido entregado.

## Tipografía

Se presenta una tabla detallada con el tipo de letra y sus tamaños correspondientes para diferentes elementos textuales:

Elemento	Fuente	Tamaño (px)	Estilo
Encabezado Principal	Montserrat	32px	Negrita
Subtítulo	Montserrat	24px	Regular
Texto Principal	Montserrat	16px	Regular
Texto Secundario	Montserrat	14px	Regular

## Diseños de pantalla UI/UX



## Vista de proceso de compra y pantalla de usuario.

Proceso de Compra

9:30

< Historial de Compra

MasterCard

VISA

PayPal

**Nombre de la tarjeta**

**No. Tarjeta**

**Vencimiento**

Subtotal

\$43.00

IVA

\$4.30

Total

\$47.30

Pagar

Checklist

9:30

✓

Éxito !

Su pago se realizó correctamente.  
Se le ha enviado un recibo de esta compra a su correo electrónico.

Regresar

usuarios

9:30

< Usuario

**Nombre**

**Correo**

**Contraseña**

**Teléfono**

**Dirección**

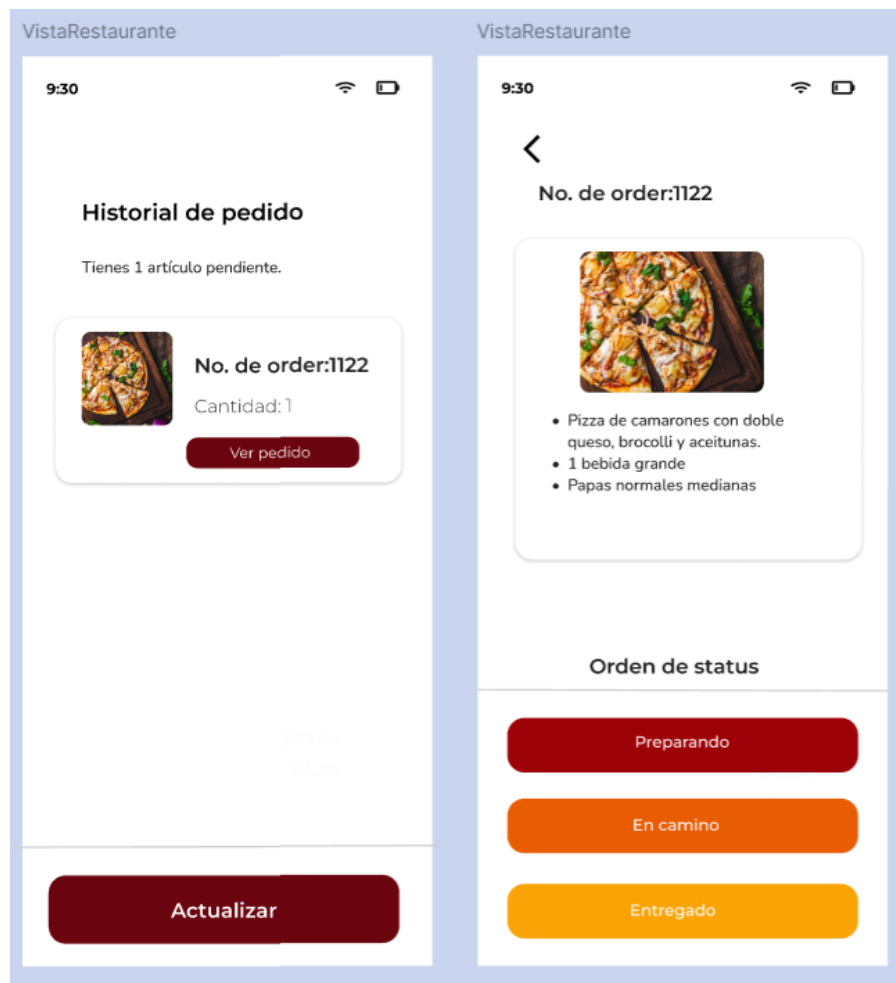
**Formas de pago**  

Tarjeta

Cerrar Sesión



Pantalla de recibo de pedidos hacia el restaurante.



## Lógica y diagramas UML

### - Definiciones

- **Usuario:** cualquier persona natural que interactúe con la APP happyPedidos, en calidad de cliente o de propietario/administrador de un restaurante registrado en la APP.
- **Cliente:** Usuario registrado dentro de la APP happyPedidos que interactúa con los módulos de la APP desde la vista de usuario.
- **Administrador de restaurante:** Usuario registrado dentro de happyPedidos, que interactúa con las vistas y módulos diseñados para usuarios con el privilegio de Administrador de restaurantes.

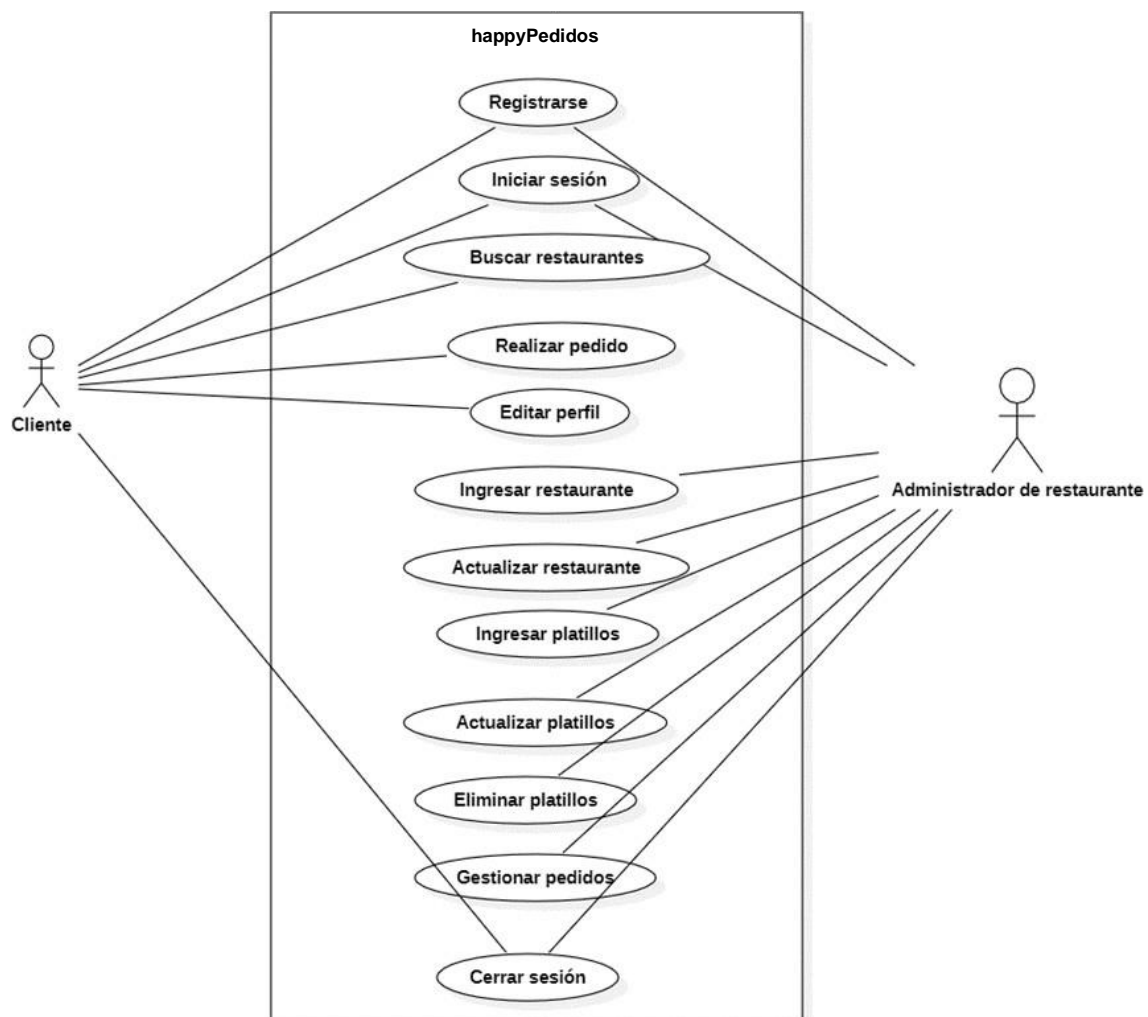
- Suposición de escenario:

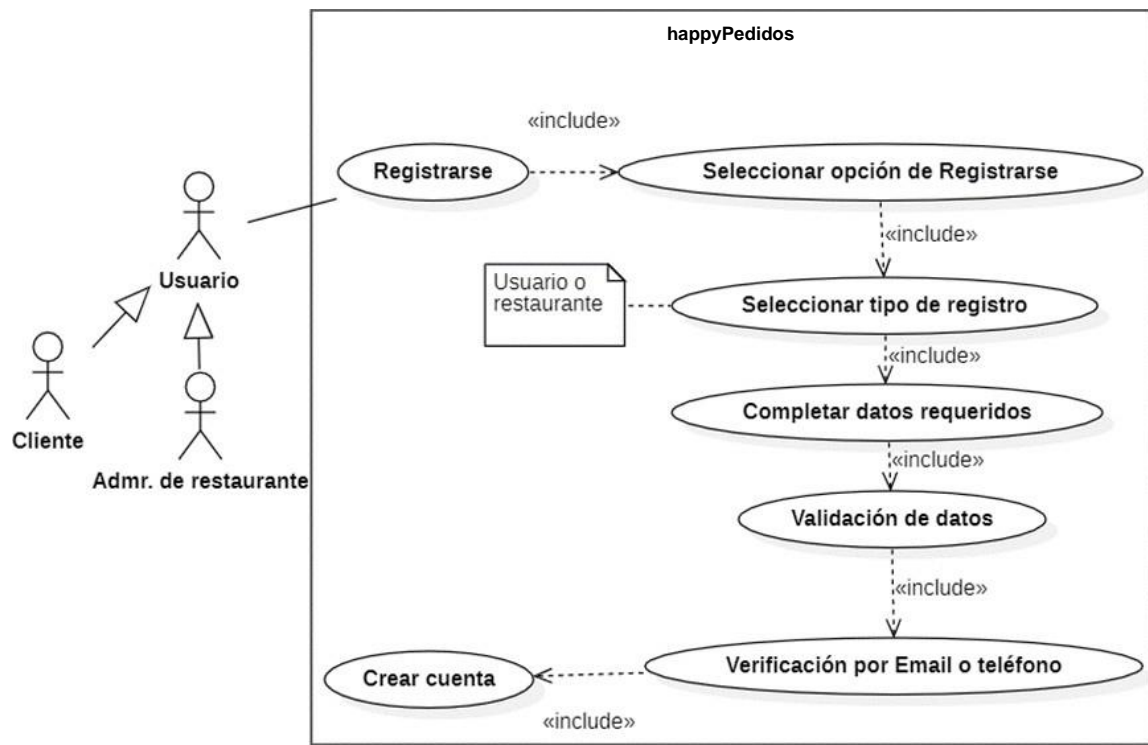
Los casos de uso donde intervengan los actores Cliente y Administrador de restaurante omiten el caso de uso “Iniciar Sesión” porque se asumen como usuarios ya registrados dentro de la APP.

En los casos de uso específicos se han omitido ciertos casos de uso presentes en el CU general, debido a que poseen un ciclo de vida muy corto.

### Caso de uso: General

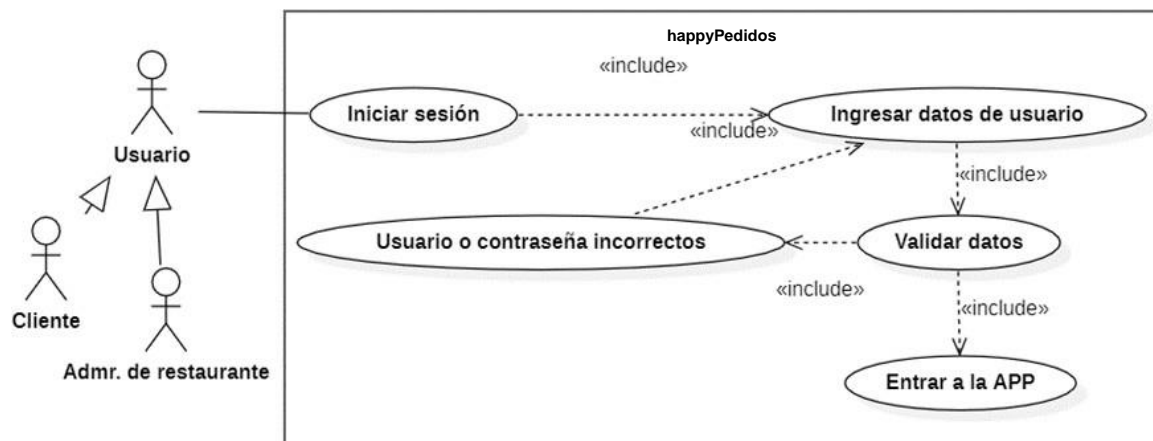
**Objetivo:** Ilustrar de manera general las funcionalidades de la APP HappyPedidos, definiendo los artefactos de acciones que son accesibles por los actores.





### Caso de uso: Registrarse

**Objetivo:** Permitir a los usuarios registrarse dentro de la APP según el perfil correspondiente.

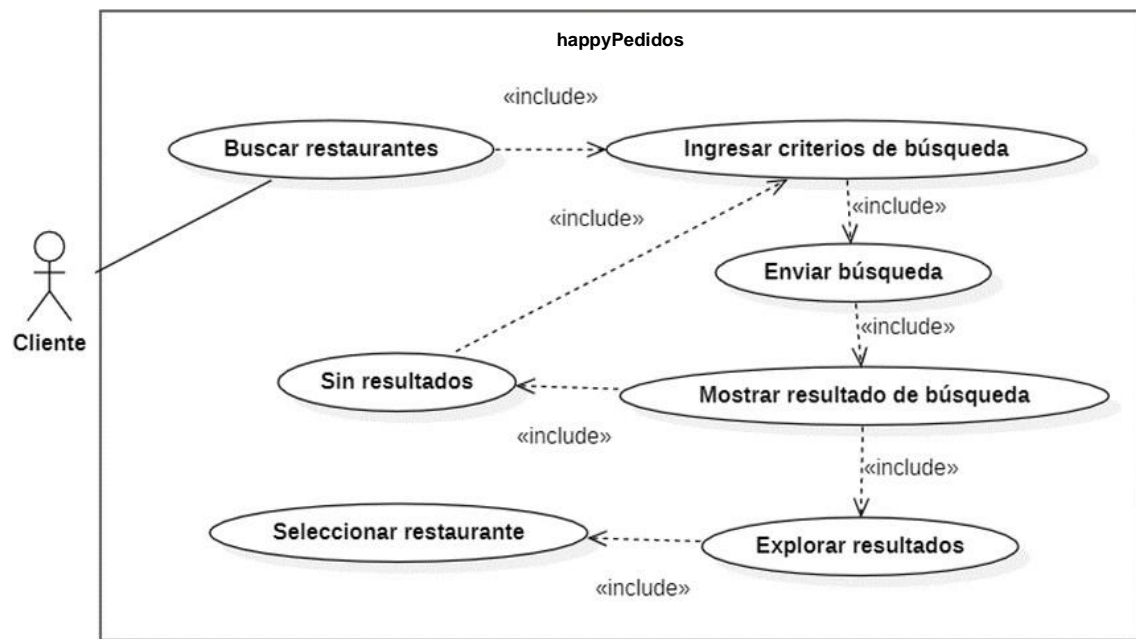


### Caso de uso: Inicio de sesión

**Objetivo:** Validar el acceso a los usuarios con sus credenciales de registro

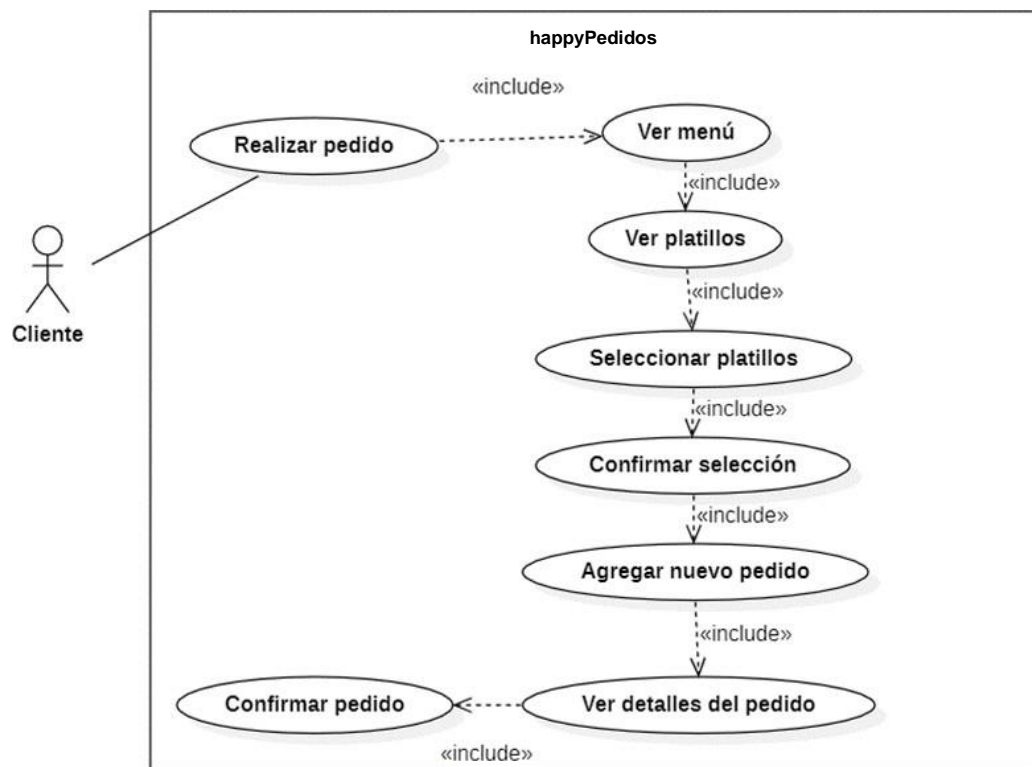
### Caso de uso: Restaurantes

**Objetivo:** Mostrar al cliente la lista de restaurantes disponibles para ofrecer una variedad de opciones para elegir.



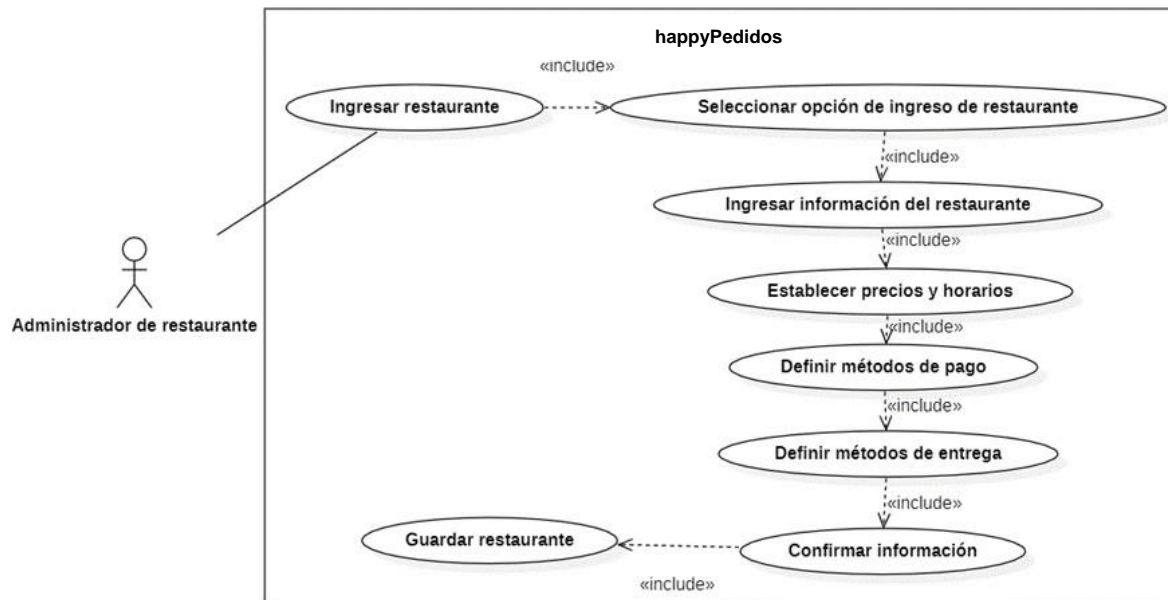
### Caso de Uso: Realizar pedido

**Objetivo:** Proporcionar al cliente un ruta para poder realizar una orden a un restaurante previamente seleccionado.



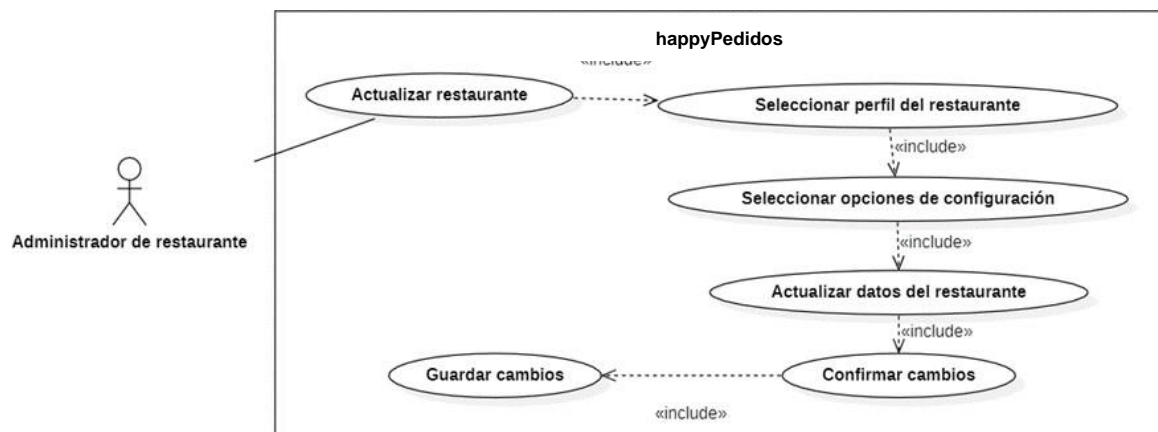
### Caso de uso: Ingresar restaurante

**Objetivo:** Permitir al Admr. de restaurante el ingreso de un restaurante para promocionarse dentro de la APP.



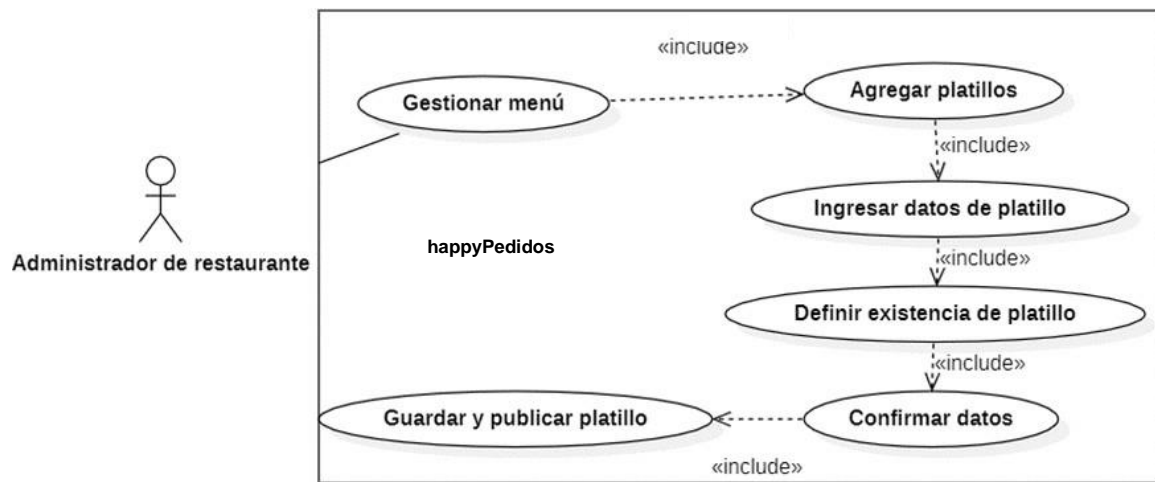
### Caso de uso: Actualizar restaurante

**Objetivo:** Permitir al Admr. de restaurante la actualización/modificación del restaurante registrado.



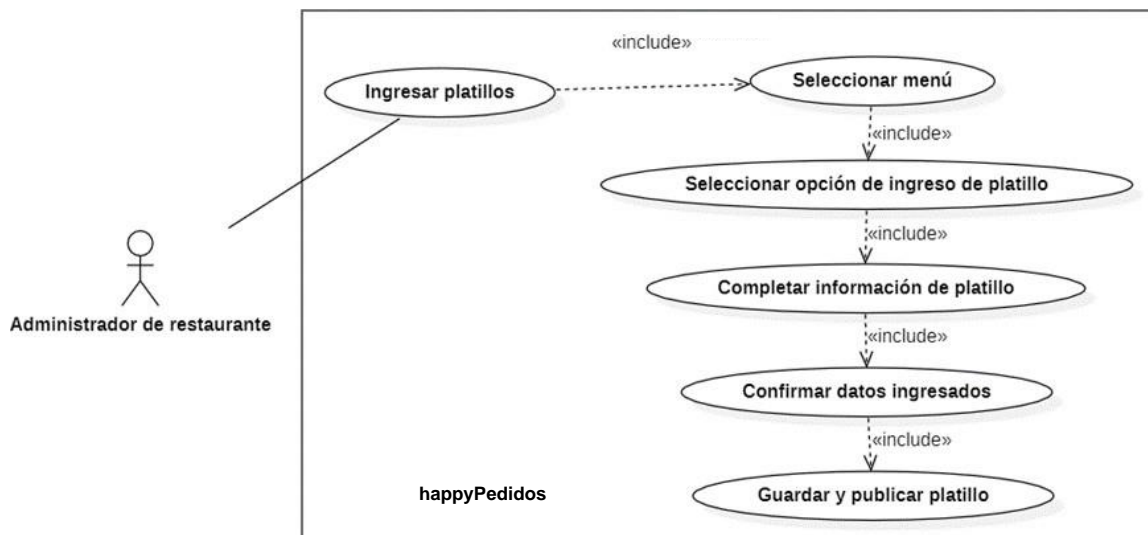
**Caso de uso:** Gestionar menú

**Objetivo:** Permitir al Admr. de restaurante la organización/gestión de menús, mediante el manejo de inventario de platillos.



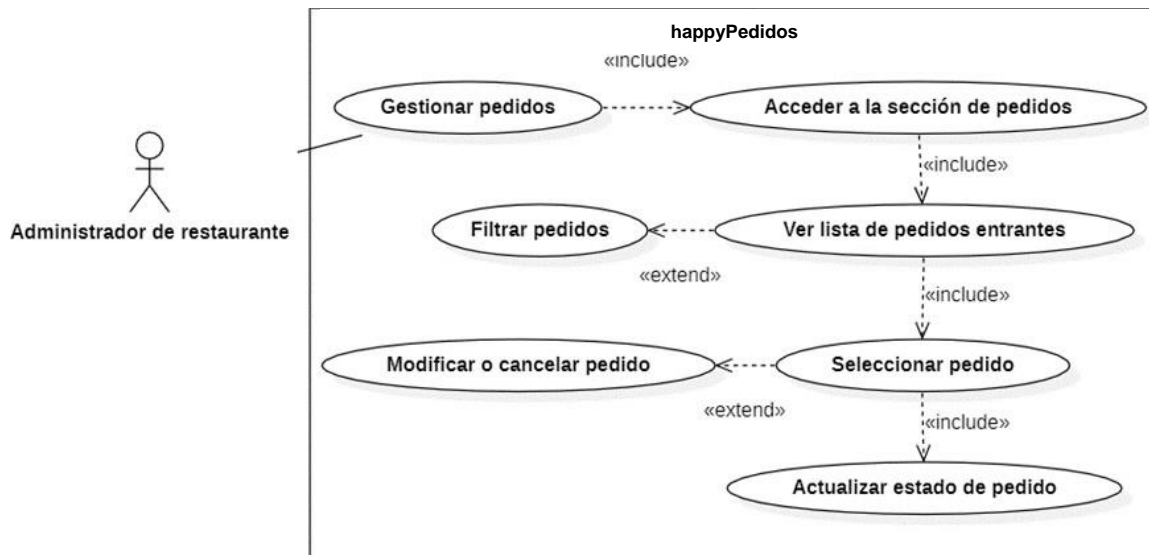
**Caso de uso:** Ingresar platillos

**Objetivo:** Permitir al Admr. de restaurante la gestión de menús del restaurante registrado, para actualizar la información relacionada de los platillos ofertados, asimismo, el control de existencias de los platillos.



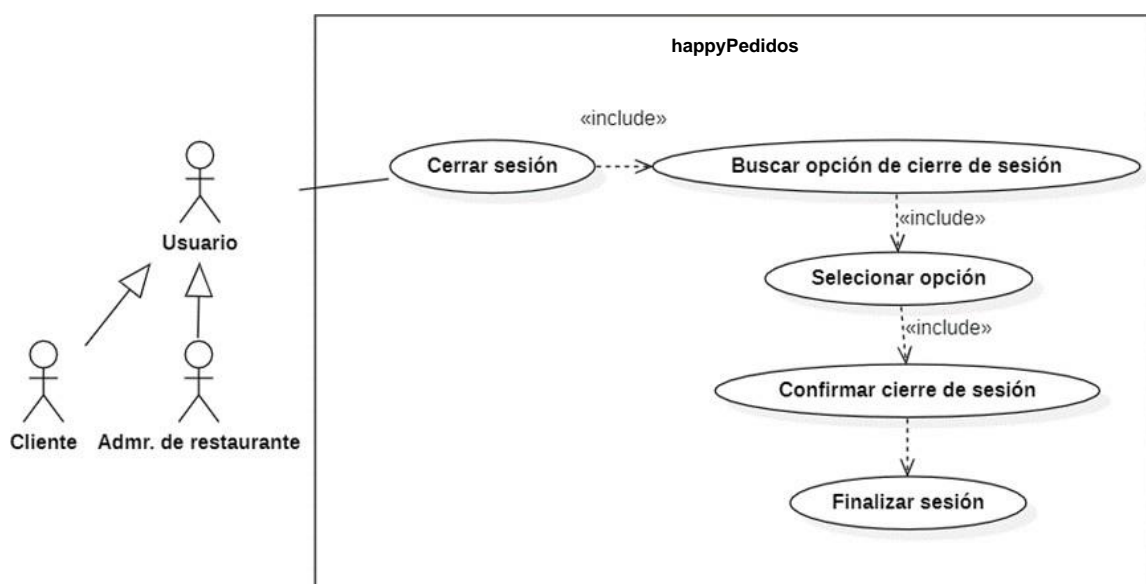
### Caso de uso: Gestionar pedidos

**Objetivo:** Permitir al Admr. de restaurante gestionar los pedidos entrantes, de tal manera que se pueda actualizar el estado de cada uno de los pedidos: Entregado o cancelado (sólo en excepciones).

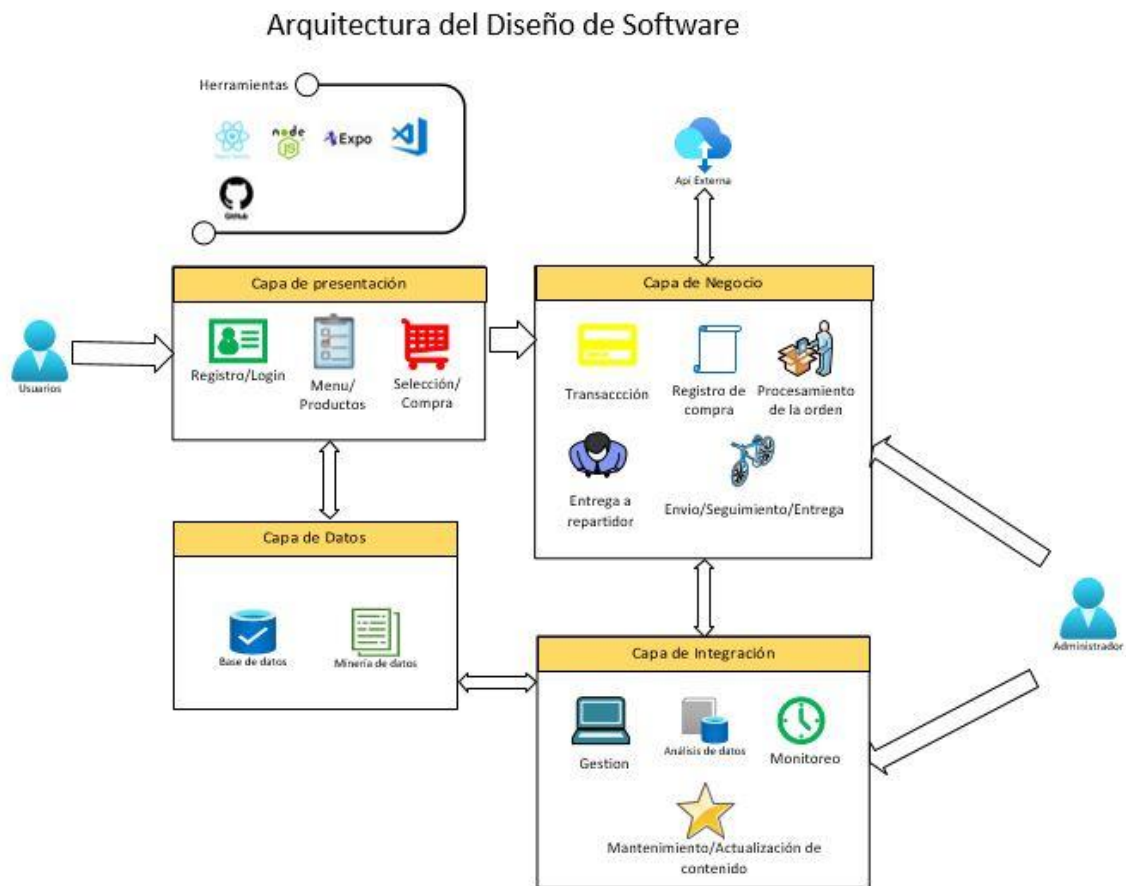


### Caso de uso: Cerrar sesión

**Objetivo:** Permitir a los usuarios el cierre de sesión dentro de la APP para garantizar una salida segura.



## Diagrama gráfico del diseño de la arquitectura del proyecto de software





## Herramientas

### React



React es una biblioteca de JavaScript desarrollada por Facebook para la construcción de interfaces de usuario. Su enfoque basado en componentes reutilizables permite a los desarrolladores crear interfaces de manera eficiente y modular. Cada componente puede manejar su propio estado y lógica, lo que hace que la actualización de la interfaz sea rápida y escalable.

React también cuenta con una curva de aprendizaje relativamente baja para desarrolladores con experiencia en JavaScript, lo que permite incorporar nuevos miembros al equipo sin una transición prolongada. Esta eficiencia, junto con la capacidad de React para integrarse con frameworks de backend como Node.js, lo convierte en una elección natural para proyectos web modernos que requieren escalabilidad y alto rendimiento.

En nuestro proyecto, optamos por usar React debido a su capacidad para crear aplicaciones web dinámicas, con una experiencia de usuario fluida y un rendimiento optimizado. Además, su gran comunidad y ecosistema de herramientas complementarias garantizan una integración sencilla con otras tecnologías.

### React Native



React Native permite la creación de aplicaciones móviles nativas utilizando JavaScript y React. En lugar de desarrollar aplicaciones por separado para iOS y Android, React Native permite construir una sola aplicación que se ejecuta en ambas plataformas, aprovechando los componentes nativos del sistema operativo.

Elegimos React Native para nuestro proyecto debido a la eficiencia que ofrece en el desarrollo multiplataforma, lo que nos permite reducir el tiempo de desarrollo y mantenimiento. Además, React Native facilita el acceso a las capacidades nativas del dispositivo, ofreciendo una experiencia de usuario casi indistinguible de las aplicaciones desarrolladas en código nativo.

React también cuenta con una curva de aprendizaje relativamente baja para desarrolladores con experiencia en JavaScript, lo que permite incorporar nuevos miembros al equipo sin una transición prolongada. Esta eficiencia, junto con la capacidad de React para integrarse con frameworks de backend como Node.js, lo convierte en una elección natural para proyectos web modernos que requieren escalabilidad y alto rendimiento.

## MySQL



MySQL es un sistema de gestión de bases de datos relacional de código abierto que ha sido ampliamente adoptado en la industria desde su lanzamiento en 1995. Conocido por su rendimiento, flexibilidad y confiabilidad, MySQL se utiliza para gestionar grandes volúmenes de datos y es compatible con una variedad de aplicaciones web y empresariales.

En nuestro proyecto, optamos por utilizar MySQL debido a su integración fluida con lenguajes de programación como PHP y su capacidad para manejar bases de datos robustas sin comprometer la velocidad de consulta. Además, MySQL cuenta con una comunidad activa y soporte comercial, lo que facilita su mantenimiento y la resolución de problemas. Su escalabilidad y eficiencia lo convierten en la opción ideal para aplicaciones que requieren gestión avanzada de datos, sin sacrificar la simplicidad en su uso y configuración.

## PhpMyAdmin



PhpMyAdmin es una herramienta basada en web para la administración de bases de datos MySQL y MariaDB. Desde su lanzamiento en 1998, PhpMyAdmin ha sido una de las herramientas más utilizadas por desarrolladores y administradores de bases de datos para gestionar bases de datos de manera visual, sin necesidad de interactuar directamente con comandos SQL.

La interfaz gráfica de PhpMyAdmin simplifica tareas comunes como la creación de tablas, la ejecución de consultas, la gestión de usuarios y la exportación/importación de datos. En nuestro proyecto, optamos por PhpMyAdmin debido a su facilidad de uso y su capacidad para simplificar la gestión de nuestras bases de datos. Esta herramienta es especialmente valiosa en entornos de desarrollo colaborativo, donde diferentes miembros del equipo pueden necesitar realizar cambios en la base de datos sin ser expertos en SQL.

## Node.js



Node.js es un entorno de ejecución de JavaScript del lado del servidor que ha transformado la forma en que se desarrollan aplicaciones web desde su lanzamiento en 2009. Basado en el motor V8 de Google, Node.js permite a los desarrolladores utilizar JavaScript para construir aplicaciones escalables y de alto rendimiento que pueden manejar múltiples solicitudes simultáneamente sin bloquear el servidor.

Una de las razones clave por las que elegimos Node.js para nuestro proyecto es su capacidad para unificar el desarrollo del frontend y el backend en un solo lenguaje: JavaScript. Esto no solo simplifica el desarrollo, sino que también permite una mayor coherencia y colaboración entre los diferentes equipos de desarrollo.

El ecosistema de Node.js, impulsado por su gestor de paquetes Npm, es otro factor determinante en nuestra elección. Con más de un millón de paquetes disponibles, Node.js ofrece una solución rápida y eficiente para casi cualquier necesidad de desarrollo, lo que acelera significativamente el tiempo de entrega del proyecto.

## Npm



Npm, que significa Node Package Manager, es el gestor de paquetes predeterminado para Node.js, y se ha convertido en una herramienta indispensable para los desarrolladores JavaScript desde su lanzamiento en 2010. Npm permite instalar, compartir y administrar librerías y dependencias de manera sencilla, lo que facilita la reutilización de código y acelera el desarrollo de aplicaciones.

En nuestro proyecto, optamos por utilizar Npm debido a su capacidad para gestionar las dependencias de nuestros módulos y mantener un entorno de desarrollo consistente. Npm no solo facilita la instalación de paquetes de terceros, sino que también permite crear scripts de automatización para tareas comunes de desarrollo, como la ejecución de pruebas o la compilación de código. Esto contribuye a un flujo de trabajo más eficiente y reduce la posibilidad de errores humanos.

Además, el ecosistema de Npm es extremadamente vasto, con una amplia gama de paquetes disponibles para prácticamente cualquier necesidad de desarrollo. Esta versatilidad nos permite integrar rápidamente nuevas funcionalidades en nuestro proyecto sin tener que desarrollar todo desde cero, ahorrando tiempo y recursos.

## GitHub



GitHub es un sistema de control de versiones distribuido que ha revolucionado la gestión de código en proyectos de desarrollo desde su creación en 2005 por Linus Torvalds. Git permite a los desarrolladores rastrear cambios en el código de manera eficiente, colaborar con otros desarrolladores en tiempo real y revertir errores de manera segura.

En nuestro proyecto, Git se convierte en una herramienta esencial para gestionar el código fuente, facilitando el trabajo en equipo y asegurando que todos los miembros del equipo puedan colaborar sin temor a perder trabajo o introducir errores accidentales. Además, Git permite trabajar en diferentes ramas de desarrollo simultáneamente, lo que nos brinda la flexibilidad de desarrollar nuevas funciones o corregir errores sin interrumpir la versión principal del proyecto.

Otro beneficio de Git es su amplia adopción en la industria y su integración con plataformas como GitHub y GitLab, que ofrecen herramientas adicionales de colaboración y revisión de código. Esta flexibilidad y control detallado sobre el código hacen de Git una opción imprescindible en cualquier proyecto de desarrollo serio.

## Visual Studio Code



Visual Studio Code, comúnmente conocido como VS Code, es un editor de código desarrollado por Microsoft que ha ganado una inmensa popularidad entre los desarrolladores desde su lanzamiento en 2015. VS Code combina la ligereza y rapidez de un editor de texto con las poderosas características de un entorno de desarrollo integrado (IDE), ofreciendo lo mejor de ambos mundos.

En nuestro proyecto, optamos por utilizar Visual Studio Code debido a su flexibilidad y capacidad de personalización. VS Code cuenta con una amplia variedad de extensiones que permiten a los desarrolladores adaptar el editor a sus necesidades específicas, ya sea que trabajen con JavaScript, Python, PHP u otros lenguajes. Además, su integración nativa con Git facilita el control de versiones directamente desde el editor, lo que mejora la productividad al eliminar la necesidad de cambiar constantemente de herramientas.

Otra ventaja clave de VS Code es su soporte para depuración integrada, autocompletado inteligente, y su facilidad para trabajar en colaboración a través de extensiones como Live Share. Estas características convierten a VS Code en una opción completa y eficiente para desarrollar aplicaciones modernas.

## Bootstrap



Bootstrap, anteriormente conocido como Twitter Blueprint, fue lanzado como código abierto en 2011. Desde entonces, ha experimentado varias actualizaciones. Este framework combina CSS y JavaScript para mejorar la apariencia y la funcionalidad de los elementos en una página web. Más allá de simplemente cambiar los colores de botones y enlaces, Bootstrap añade interactividad a las páginas mediante una variedad de componentes, como menús de navegación, barras de progreso, y controles de paginación.

Su popularidad radica en su facilidad de uso, permitiendo crear sitios web responsivos sin necesidad de escribir mucho código CSS. Sus componentes prediseñados ofrecen un estilo moderno y limpio, ideal para diseñar interfaces atractivas y funcionales.

La elección de esta herramienta se dio gracias a su fácil uso y aplicación, ya que nos permite diseñar aplicaciones responsivas que funcionen en una gran variedad de dispositivos sin escribir demasiado código CSS y sus componentes prediseñados, lucen minimalistas, atractivos e intuitivos.

## Laravel



Laravel es un framework PHP moderno y flexible que ha ganado popularidad por su capacidad para simplificar el desarrollo de aplicaciones web robustas. Desde su lanzamiento en 2011, Laravel se ha destacado por su enfoque en una sintaxis clara y elegante, así como por su conjunto de herramientas que aceleran el desarrollo sin sacrificar la calidad.

En nuestro proyecto, elegimos Laravel por su capacidad para gestionar la lógica de la aplicación de manera eficiente. Laravel también se destaca por su enfoque en la seguridad, ofreciendo protección integrada contra amenazas comunes como ataques de inyección de SQL y CSRF. Otro punto a favor de Laravel es su ecosistema, que incluye características avanzadas como colas de trabajo, servicios en tiempo real y la capacidad de manejar aplicaciones a gran escala. Todo esto, combinado con su comunidad activa y documentación extensa, hace de Laravel una elección ideal para proyectos que requieren estabilidad, seguridad y escalabilidad.

## Cronograma de trabajo

### Cronograma del proyecto

TAREAS	JULIO			AGOSTO			SEPTIEMBRE			OCTUBRE		
Selección de tema de proyecto, aprobación de tema de proyecto.												
Recopilación de información.												
Creación de documentación.												
Entrega de primer avance.												
Recopilación de requerimientos para el desarrollo correcto del proyecto.												
Creación y diseño de proyecto web.												
Desarrollo de proyecto web.												
Finalización del proyecto.												
Defensa del proyecto.												

## Presupuesto

Productos/Servicios	Julio	Agosto	Septiembre	Octubre	Total
Análisis para desarrollo del sistema	\$300	\$500	\$500	\$500	\$1,800
Desarrollo del sistema	\$1,200	\$1,200	\$1,000	\$1,000	\$4,800
Control de calidad	\$200	\$200	\$200	\$1,200	\$1,800
Implementación del proyecto	\$500	\$500	\$500	\$500	\$2,000
Total					\$10,400