



## APLICACIÓN ANDROID DE PEDIDOS - ALLAN PIZZA

### Integrantes UNITECH:

Josué Eduardo García Estrada	<b>GE240098</b>	Desarrollador Backend
Jeremy Edenilson Flores Portillo	<b>FP240479</b>	Desarrollador Frontend
Oscar Daniel Soto Jovel	<b>SJ241841</b>	Desarrollador Frontend
Edgar Josué Gómez Meléndez	<b>GM240279</b>	Desarrollador Backend

### Materia:

Desarrollo de Aplicaciones Móviles | **DSM441 - 01T**

### Docente:

Ing. Juan Carlos Menjívar

### Fecha de entrega:

Martes 26 de Agosto del 2025

# Índice

<b>Introducción</b>	<b>2</b>
<b>Perfil del proyecto</b>	<b>3</b>
Problema a resolver:	3
Objetivo General:	4
Metas:	4
Resultados esperados:	5
Metodología:	6
Área geográfica y beneficiarios del proyecto:	7
Factores críticos de éxito y posibles soluciones:	7
Carta de presentación del proyecto	9
<b>Presentación del diseño en Figma</b>	<b>10</b>
<b>Logica a Utilizar</b>	<b>12</b>
1) Contexto y alcance del módulo lógico	12
2) Reglas de negocio clave	12
3) Flujos funcionales	14
3.1 Crear Pedido (Cliente)	14
3.2 Avanzar estado (Empleado)	14
3.3 Cancelar/ Rechazar	15
4) Validaciones y manejo de errores	15
5) Seguridad y autorización	15
6) Notificaciones y “tiempo real”	15
7) Diagrama de Clases (Dominio)	16
8) Diagrama de Crear Pedido	17
9) Diagrama de Actualizar Pedido	18
10) Estado de Pedido	18
<b>Diseño de la Arquitectura de Software</b>	<b>19</b>
<b>Detalles de todas las herramientas a utilizar</b>	<b>20</b>
<b>Presupuesto de la aplicación</b>	<b>21</b>
<b>Cronograma de Trabajo</b>	<b>22</b>
<b>Anexos y Documentación de Apoyo</b>	<b>23</b>
<b>Referencias Bibliográficas</b>	<b>24</b>

## **Introducción**

La pizzería Allan Pizza busca dar un paso hacia la modernización de sus procesos a través del desarrollo de una aplicación web que facilite la gestión de pedidos y una aplicación móvil para hacer pedidos. La idea nace de la necesidad de agilizar el trabajo del personal y, al mismo tiempo, brindar a los clientes una forma más sencilla y confiable de realizar sus órdenes.

Con esta herramienta, los clientes podrán hacer pedidos desde cualquier dispositivo android y dar seguimiento en tiempo real al estado de su orden, lo que les dará mayor comodidad y confianza. Para los empleados, la aplicación se convertirá en un apoyo clave, ya que permitirá llevar un mejor control de la información, reducir errores en la toma manual de pedidos y organizar de manera más eficiente la preparación y entrega de los productos.

Además, al contar con un sistema automatizado, Allan Pizza podrá aprovechar mejor sus recursos y ofrecer un servicio más ágil y ordenado. A largo plazo, este proyecto no solo busca resolver los problemas actuales en la gestión de pedidos, sino también abrir la puerta a futuras mejoras, como promociones personalizadas, integración de pagos en línea o programas de fidelización.

De esta manera, este sistema representa una oportunidad para fortalecer la relación con los clientes, mejorar la productividad del equipo de trabajo y posicionar a Allan Pizza como un negocio innovador, adaptado a las nuevas formas de consumo.

## Perfil del proyecto

**Nombre del proyecto:** Aplicación Android de Pedidos | Allan Pizza

**Nombre de la aplicación:** Allan Pizza

### Problema a resolver:

Actualmente, la pizzería Allan Pizza gestiona los pedidos de sus clientes a través de WhatsApp Business, lo que genera diversas dificultades operativas y afecta la experiencia tanto de los clientes como de los empleados. Entre los principales inconvenientes se encuentran:

- **Desorganización en la gestión de pedidos:** Los empleados deben registrar manualmente cada pedido, lo que puede generar confusión, retrasos y errores en la preparación.
- **Falta de seguimiento en tiempo real:** Los clientes no tienen una forma clara de saber si su pedido ha sido recibido, está en preparación o listo para entrega. Esto genera incertidumbre y posibles reclamos.
- **Carga operativa para los empleados:** La actualización del estado de los pedidos se hace de forma manual, lo que implica una mayor carga de trabajo, demoras en cuanto al tiempo y reduce la eficiencia en la atención al cliente.

Para solucionar estos problemas, se desarrollará una aplicación web (administrador) y una aplicación móvil (clientes) que permitirá a los clientes realizar pedidos de manera más rápida y sencilla, consultar su estado en tiempo real y gestionar su historial de compras. Por otro lado, los empleados podrán visualizar los pedidos en orden de llegada y actualizar su estado de forma automatizada, optimizando así el flujo de trabajo dentro del negocio.

A futuro, la plataforma podría incluir funciones adicionales como informes de ventas para el administrador, mejorando aún más la eficiencia y el control del negocio.

## **Objetivo General:**

Desarrollar un ecosistema compuesto por una **aplicación web para los administradores y una aplicación móvil** que permita a los clientes de Allan Pizza realizar pedidos de manera más rápida y sencilla, ofreciendo una mejor experiencia al usuario y facilitando el seguimiento de pedidos en tiempo real, ayudando al crecimiento del negocio, además de optimizar la gestión interna y logística de la pizzería, permitiendo a los empleados visualizar los pedidos en orden de llegada y actualizar su estado fácilmente desde la plataforma web, mejorando la eficiencia operativa y el control del flujo de trabajo.

## **Metas:**

- Diseñar e implementar un sistema de pedidos en línea capaz de optimizar la experiencia de los usuarios, permitiéndoles seleccionar su orden y personalizar su pedido, eliminando así la necesidad de usar WhatsApp como tecnología para realizar la orden, lo que garantiza la seguridad y acceso de la información de los usuarios.
- Optimizar la carga de trabajo del personal al reducir la necesidad de ingresar pedidos manualmente, automatizando el registro y actualización de cada orden dentro del sistema, permitiendo a los empleados enfocarse en la preparación y entrega de los pedidos y a los usuarios saber cuándo llegarán.
- Implementar un sistema de autenticación de usuarios para clientes y empleados, permitiendo un acceso seguro a la plataforma con diferentes niveles de permisos según el rol asignado en la aplicación, ofreciendo a su vez una interfaz responsive y accesible para adaptarse a la necesidad de los usuarios.

## Resultados esperados:

Al momento de gestionar la idea y realizar un sistema para Allan Pizza, logramos identificar algunos aspectos a resolver. Nuestro objetivo no es solo crear un sistema, sino, transformar realmente la forma en que se gestionan sus pedidos.

- 1. Experiencia del cliente mejorada:** Con el sistema queremos lograr que los clientes de esta pizzeria al momento de comprar se sientan más cómodos al realizar pedidos y así mismo mejorar la facilidad en que lo hacen. Por ejemplo, las personas no tendrán que mandar más mensajes por WhatsApp o alguna otra red social o app de terceros para ver su estado de la pizza o cuanto tarda en llegar.
- 2. Procesos internos más fluidos:** Tomando en cuenta que es un restaurante de comida rápida, tienden a manejar los pedidos a mano. Nuestra solución les dará herramientas para ser más eficientes y reducir errores.
- 3. Información centralizada:** Tener notas en libretas, chats perdidos y una mala organización queda atrás. Todo quedará registrado de manera clara y accesible a los administradores, ayudándoles a estudiar su mercado.
- 4. Listos para crecer:** Diseñaremos el sistema pensando no solo en lo que Allan Pizza necesita hoy, sino en lo que podría necesitar mañana. Queremos dejar espacio para futuras mejoras e integraciones.

## **Metodología:**

Sabemos que cada proyecto es único, por eso no vamos a seguir un método rígido, sino que nos adaptamos constantemente a las necesidades reales de Allan Pizza. Hemos decidido dividir nuestro desarrollo en etapas más pequeñas y manejables. La idea es poder ver resultados rápido y poder ajustar sobre la marcha.

- **Planificación colaborativa**

Tomaremos como referencia el modelo de Allan Pizza para comprender a fondo cada detalle del sistema, incluyendo aspectos como la paleta de colores, la información a incluir, la estructura de la base de datos, entre otros. Además, desarrollaremos prototipos en Figma que permitirán visualizar de manera clara y eficiente las funcionalidades del sistema. Este enfoque nos ayudará a lograr la mejor versión posible del producto y evitar inconvenientes durante la entrega.

- **Desarrollo por etapas**

Para garantizar el mejor funcionamiento del sistema, nuestro equipo llevará a cabo reuniones periódicas para revisar y presentar los avances del desarrollo, tanto en su aspecto funcional como digital. Este seguimiento interno nos permitirá mantener una retroalimentación constante y asegurar que el proyecto progrese de manera óptima.

- **Pruebas con usuarios reales**

No nos quedaremos solo con pruebas técnicas. Queremos que personas externas puedan hacer pruebas de nuestra aplicación y ver que tan amigable y fácil de usar es nuestro proyecto.

## Área geográfica y beneficiarios del proyecto:

La empresa Allan Pizza tiene ubicación **Bolv. Las Pavas, Ilopango, El Salvador** contando con un local que permite ser encontrado fácilmente, llegando así a una cantidad de clientes.

Los clientes principales son los habitantes de la colonia Altavista, debido a la accesibilidad del establecimiento, por supuesto el negocio cuenta con otra cartelera de clientes con otras ubicaciones no tan cercanas, por lo que todos serán los principales **beneficiarios** del sistema que se realizará.

## Factores críticos de éxito y posibles soluciones:

- **Adopción por parte de clientes y empleados**
  - **Posible solución:** Diseñar una interfaz intuitiva y realizar capacitaciones breves para empleados. Además, ofrecer tutoriales o asistencia a clientes para facilitar su uso.
- **Reducción de errores en pedidos**
  - **Posible solución:** Implementar un sistema automatizado que registre los pedidos de forma precisa y envíe confirmaciones automáticas a los clientes y empleados.
- **Mejora en tiempos de entrega**
  - **Posible solución:** Integrar una función que organice los pedidos en orden de llegada y notifique a los empleados sobre nuevos pedidos en tiempo real.



- **Escalabilidad y mantenimiento**
  - **Posible solución:** Diseñar la plataforma con tecnologías que permitan futuras expansiones sin afectar el rendimiento, además de planificar mantenimientos periódicos.
- **Impacto positivo en las ventas**
  - **Posible solución:** Mejorar la experiencia del usuario con un proceso de pedido rápido y promociones exclusivas dentro de la plataforma para incentivar más compras.
- **Soporte y capacitación efectiva**
  - **Posible solución:** Crear manuales, tutoriales interactivos , así mismo brindar soporte técnico para resolver dudas de empleados y clientes rápidamente.

## **Carta de presentación del proyecto**

Aplicación de Pedidos para Allan Pizza

Ilopango, El Salvador, 25 de Agosto de 2025

Señores

Allan Pizza

Presente

Estimados/as:

Nos complace presentar el proyecto “Aplicación Android de Pedidos | Allan Pizza”, una solución tecnológica que busca modernizar la gestión de pedidos de su negocio.

El sistema incluye una app móvil para clientes, que permitirá realizar pedidos y darles seguimiento en tiempo real, y una plataforma web para empleados y administradores, que facilitará el control y actualización de las órdenes, reduciendo errores y mejorando la eficiencia.

Nuestro objetivo es optimizar los procesos internos, brindar mayor comodidad a los clientes y apoyar el crecimiento de Allan Pizza mediante el uso de tecnologías modernas y escalables.

Quedamos a su disposición para cualquier aclaración y agradecemos la oportunidad de colaborar en este proyecto.

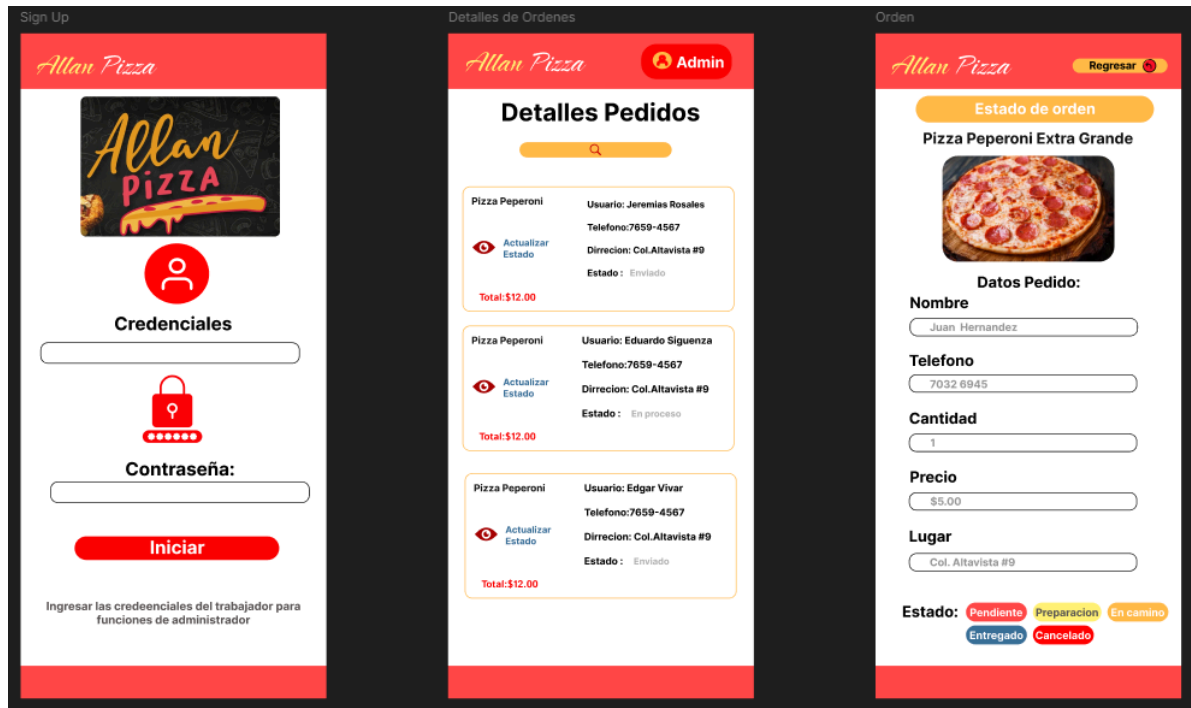
Atentamente,

Equipo de Desarrollo

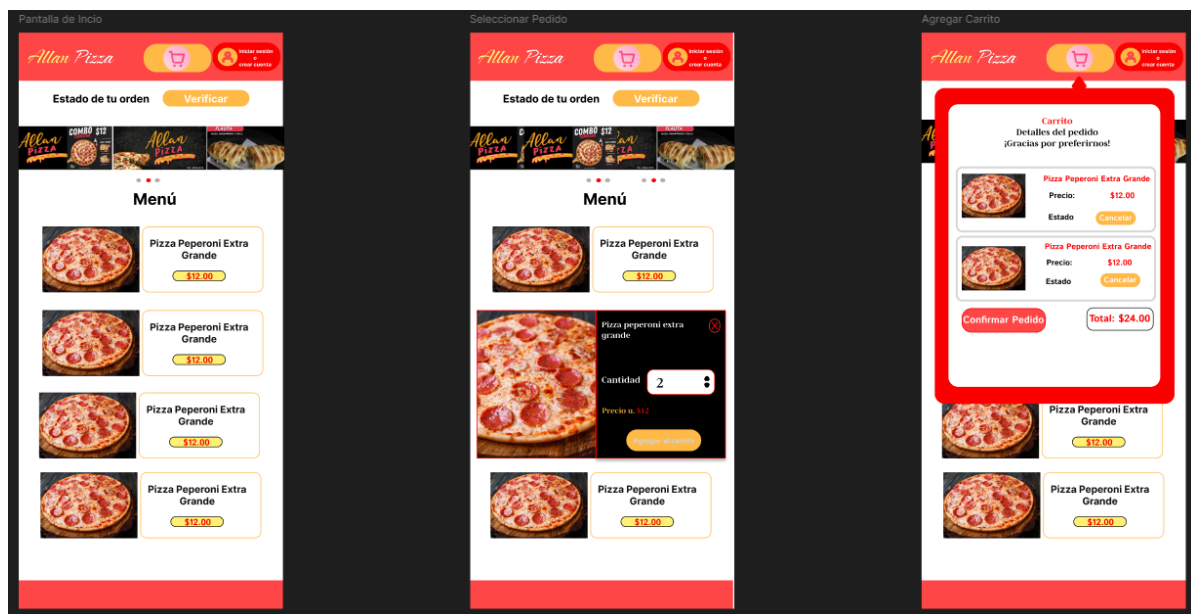
# Presentación del diseño en Figma

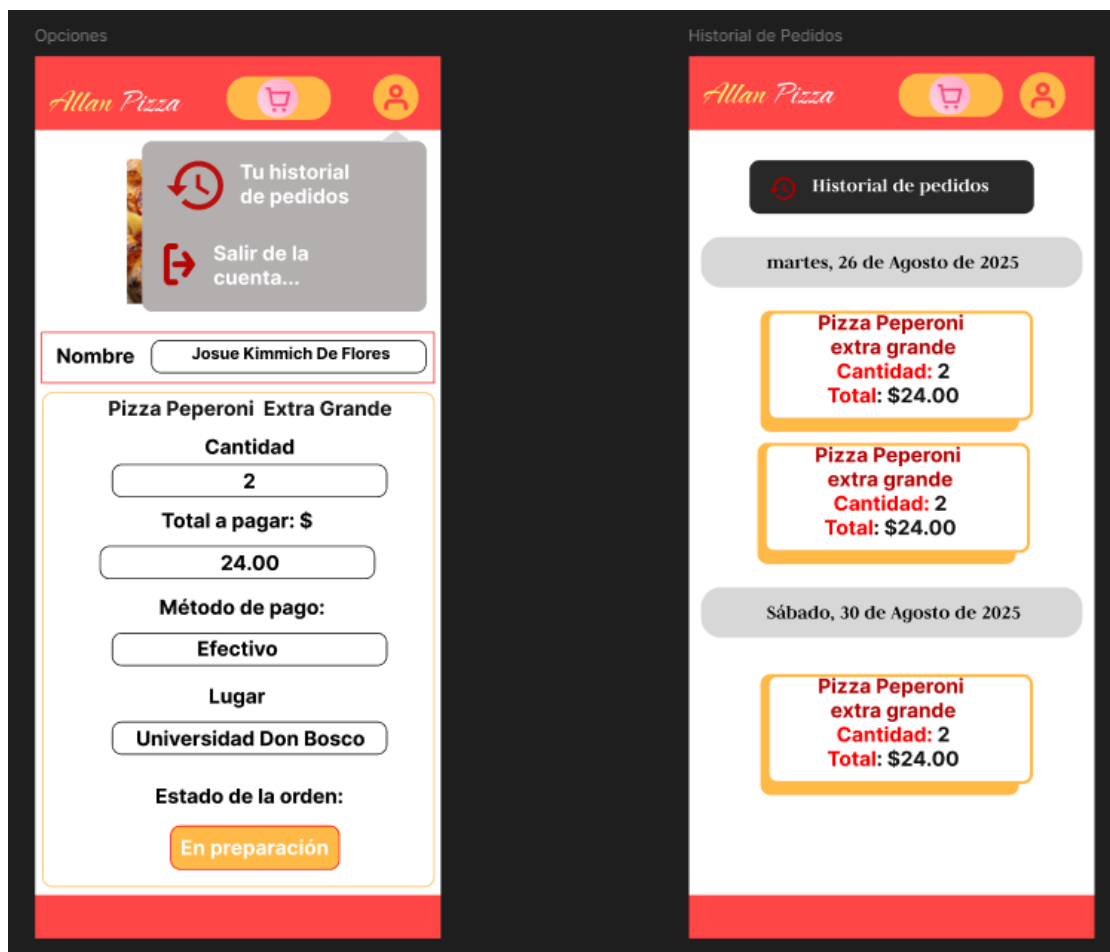
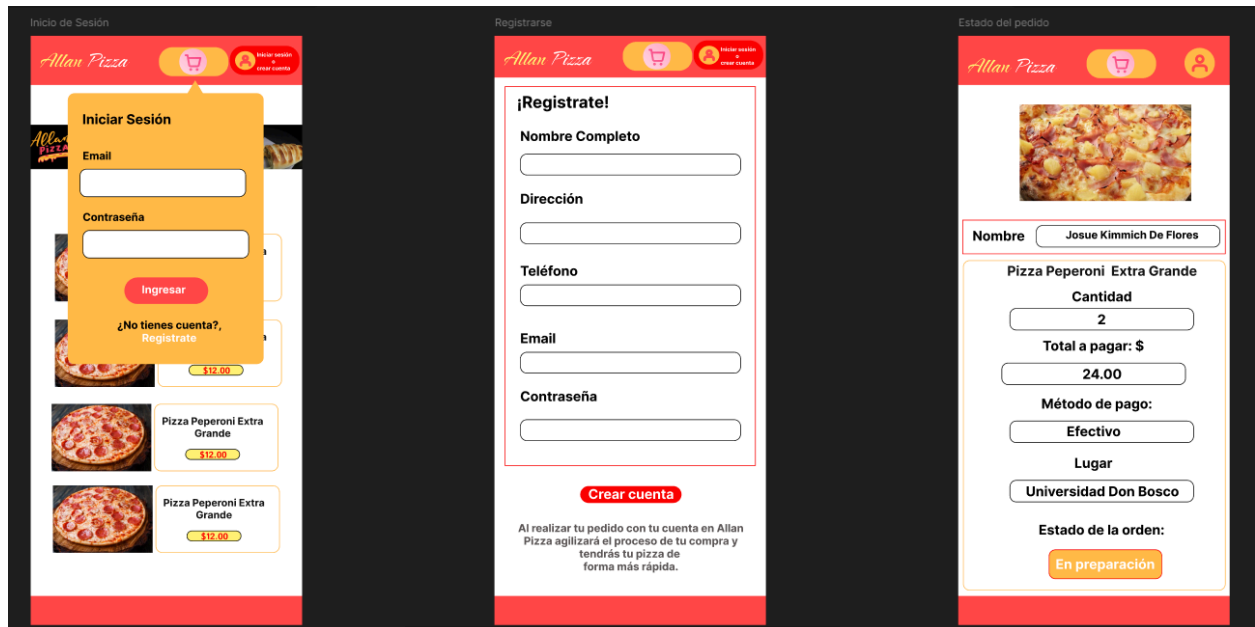
Mock Up de Figma:

## Sección Empleado

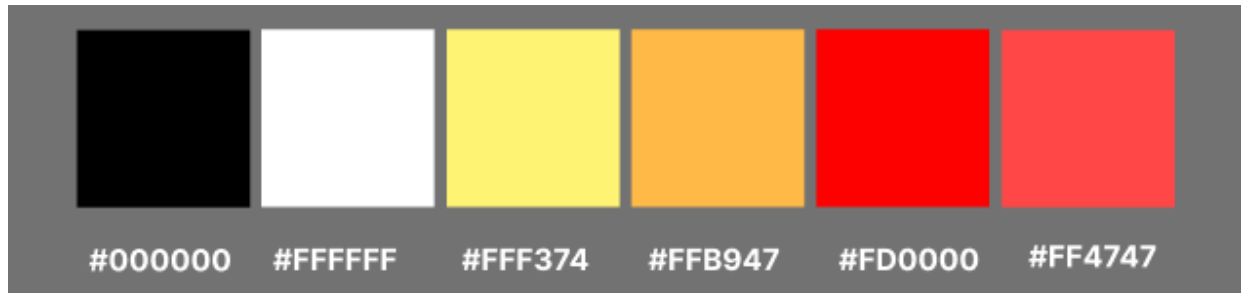


## Sección Cliente





Paleta de colores:



Iconos a utilizar: <https://tabler.io/icons>

Tipografía: Inter y Judson

# Logica a Utilizar

## 1) Contexto y alcance del módulo lógico

El sistema de Allan Pizza se compone de:

- **App móvil (clientes)** para hacer pedidos y ver su estado.
- **App web (administración/empleados)** para gestionar menú y actualizar estados.
- **API REST (Spring)** que centraliza reglas de negocio y acceso a la base de datos en **MySQL**.
- **Autenticación con Firebase** para clientes y **notificaciones push con FCM** para cambios de estado.

La lógica describe **cómo se transforma la intención del usuario en cambios coherentes de estado** (por ejemplo, de “Creado” → “En preparación” → “Listo” → “Entregado/Cancelado”), garantizando **validaciones, seguridad, consistencia y feedback en tiempo real** (push).

## 2) Reglas de negocio clave

### 1. Catálogo

- Un **Producto** tiene **precio, disponibilidad, categoría** y puede estar **no disponible** temporalmente.
- Los **precios** los define el administrador y se usan para calcular el total.

### 2. Pedido

- Estados permitidos: **CREADO** → **EN\_PREPARACION** → **LISTO** → **ENTREGADO**.

- Transiciones válidas:
  - **CREADO** → **EN\_PREPARACION** (cuando un empleado acepta/agarra el pedido)
  - **EN\_PREPARACION** → **LISTO**
  - **LISTO** → **ENTREGADO**
  - **Cancelaciones:**
    - El cliente puede **cancelar** mientras el pedido siga en **CREADO**.
    - Empleado/Admin pueden **rechazar** (por falta de stock) sólo en **CREADO**.
- Todo cambio de estado **genera notificación** (FCM) al cliente.

### 3. Cálculo de totales

- **Total** =  $\Sigma$  (**precioUnitario** \* **cantidad**) de cada ítem en el momento de crear el pedido.

### 4. Ubicación/entrega

- El cliente **registra dirección** (texto + lat/Ing). Se valida alcance de entrega si aplica.

### 5. Cola de trabajo

- En la web de empleados los pedidos se **ordenan por hora de creación** y se filtran por estado.

Un pedido **no puede** pasar a **EN\_PREPARACION** si ya está en un estado posterior.

## 6. Auditoría

- Cada transición registra: **pedidoId**, **estadoAnterior**, **estadoNuevo**, **actor** (empleado/cliente), **timestamp** y **motivo** opcional.

## 7. Seguridad

- **Clientes**: se autentican con **Firebase Auth** en la app, envían el **ID Token** al backend; el backend lo **verifica** y enlaza con su **Cliente.id**.
- **Empleados/Admin**: se autentican vía app web, con roles/claims para autorizar endpoints (Spring Security).

# 3) Flujos funcionales

## 3.1 Crear Pedido (Cliente)

1. Cliente **se autentica** (Firebase).
2. Selecciona **productos** y **cantidades**.
3. Define **dirección** y **notas** (p. ej. "sin cebolla").
4. App calcula **subtotal/total** y muestra **resumen**.
5. App envía **POST /pedidos** a la API con los ítems + dirección.
6. API válida stock y reglas, **persiste en MySQL** y retorna el **id del pedido** en **CREADO**.
7. API dispara **notificación (FCM)** a canal empleado o en cola visualización en la web admin.

## 3.2 Avanzar estado (Empleado)

1. El Empleado ve la **lista** de pedidos en la web.
2. Presiona "**Tomar/Aceptar**" → API cambia **CREADO** → **EN\_PREPARACION**.
3. Al terminar, pasa a **LISTO**.
4. Al entregar, marca **ENTREGADO**.
5. Cada cambio se notifica **por FCM** al cliente.



### 3.3 Cancelar/ Rechazar

- El **Cliente** puede cancelar si el estado es **CREADO**.
- **Empleado/Admin** pueden **rechazar** en **CREADO** con un motivo (falta de stock, dirección fuera de zona, etc.).
- Se notifica al cliente.

### 4) Validaciones y manejo de errores

- **Producto** debe existir y estar **disponible**.
- **Cantidad** > 0 y **límite** razonable.
- **Precio** se revalida en backend para evitar manipulación.
- **Dirección** obligatoria si el pedido es a domicilio.
- Errores devuelven JSON con **code**, **message**, **details**, **pagination**.

### 5) Seguridad y autorización

- **Clientes (Firebase Auth)**: La app envía **ID Token**; el backend lo **verifica** (Firebase Admin SDK) y mapea al **Cliente**.
- **Empleados/Admin (Web)**: **Spring Security** + roles (**ROLE\_EMPLOYEE**, **ROLE\_ADMIN**).

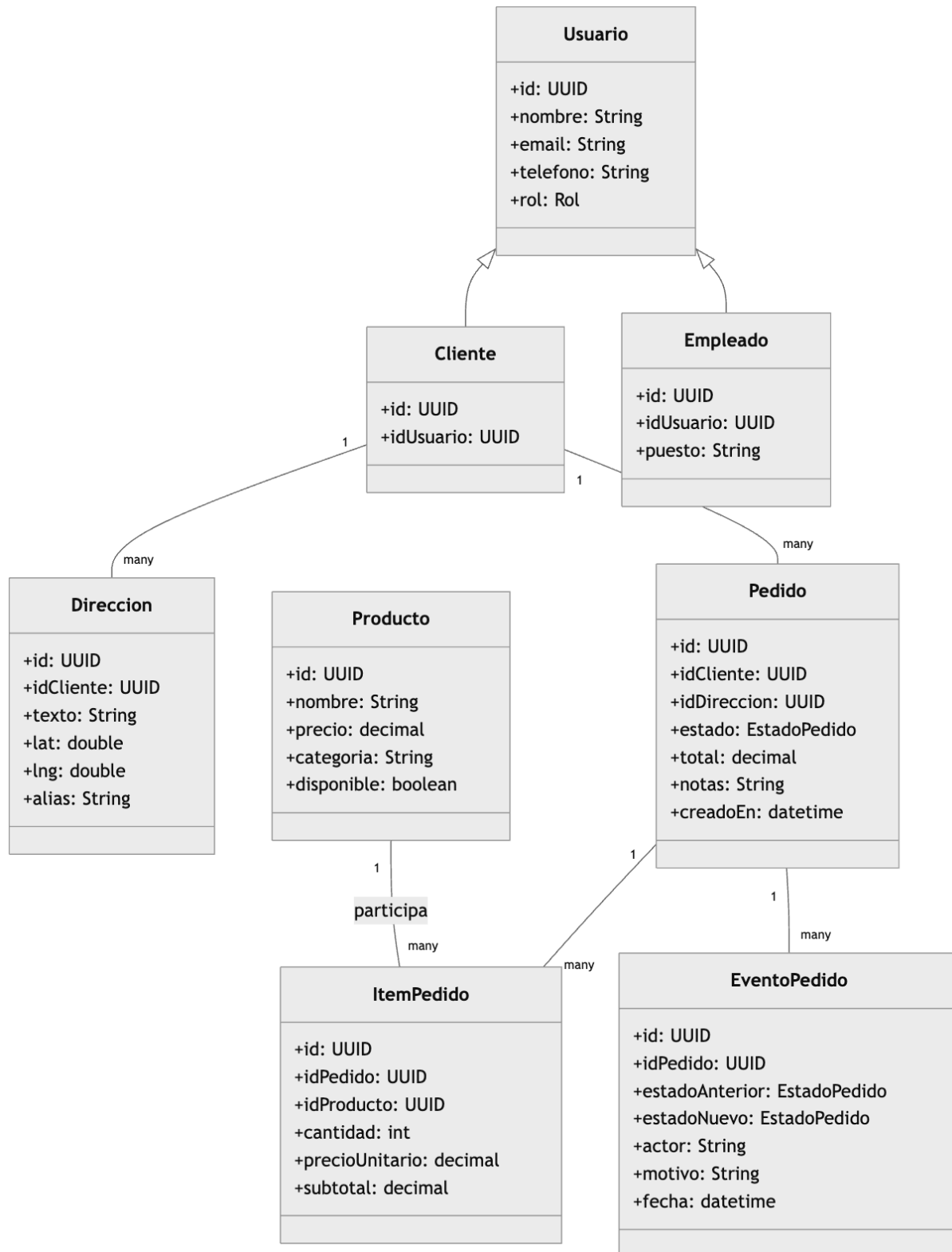
**Autorización por endpoint** (ejemplos):

- **POST /pedidos** → **ROLE\_CLIENT**
- **PATCH /pedidos/{id}/estado** → **ROLE\_EMPLOYEE | ROLE\_ADMIN**
- **GET /admin/products** (CRUD) → **ROLE\_ADMIN**
- **Regla**: Un usuario sólo puede ver **sus propios pedidos** (cliente) o **todos** (empleado/admin).

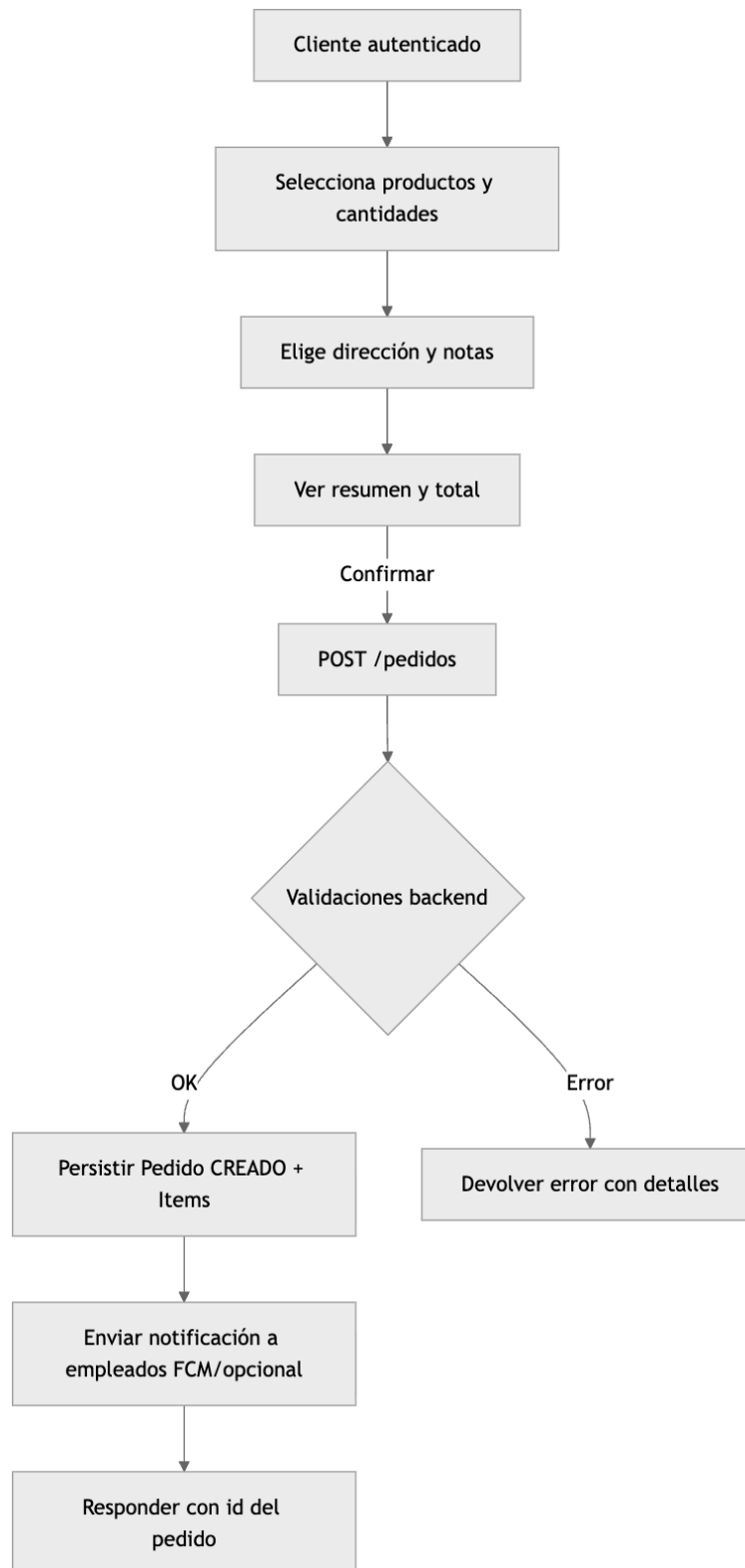
### 6) Notificaciones y “tiempo real”

- La API envía **FCM** al cliente cuando su pedido cambia de estado.
- En la app móvil, se **actualiza la UI** al recibir el push.

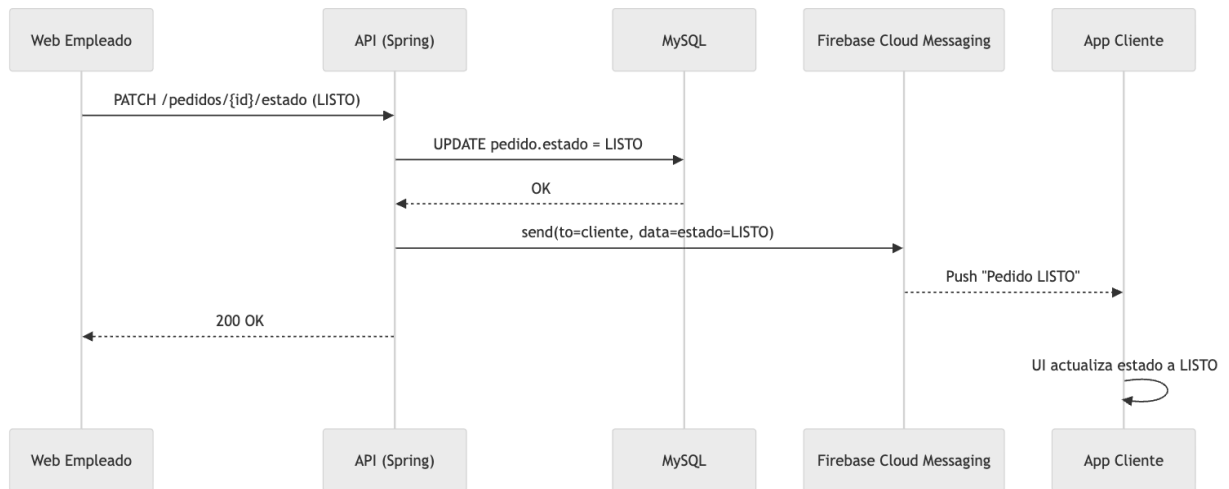
## 7) Diagrama de Clases (Dominio)



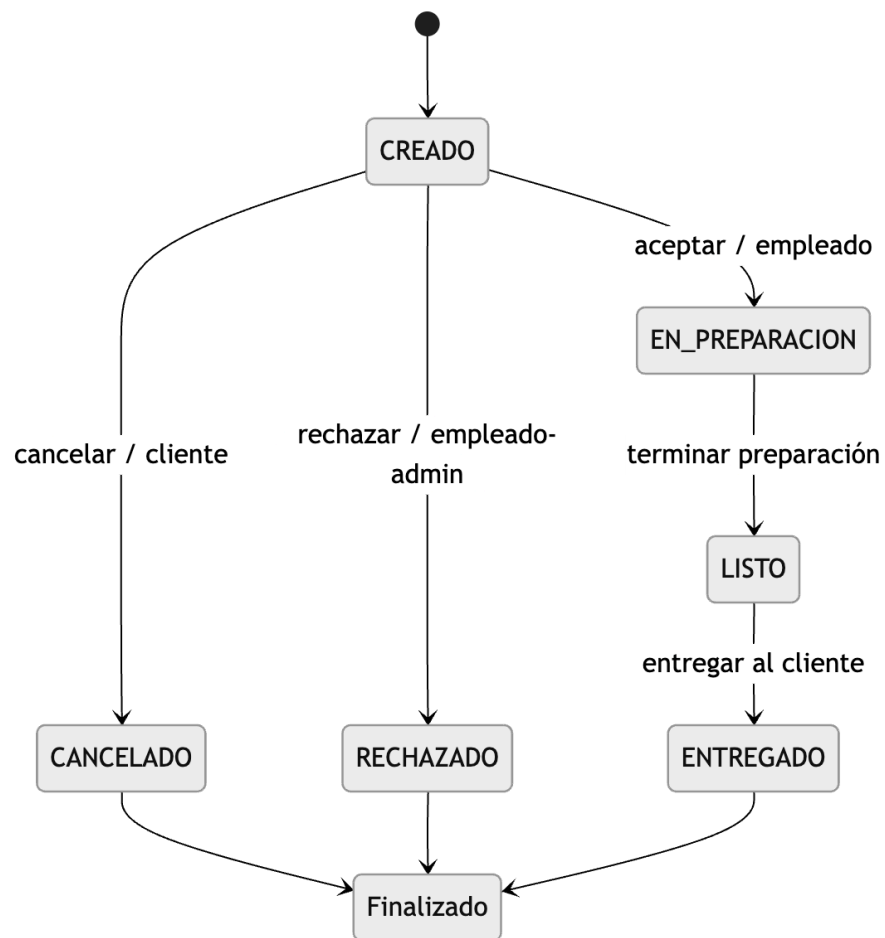
## 8) Diagrama de Crear Pedido



## 9) Diagrama de Actualizar Pedido



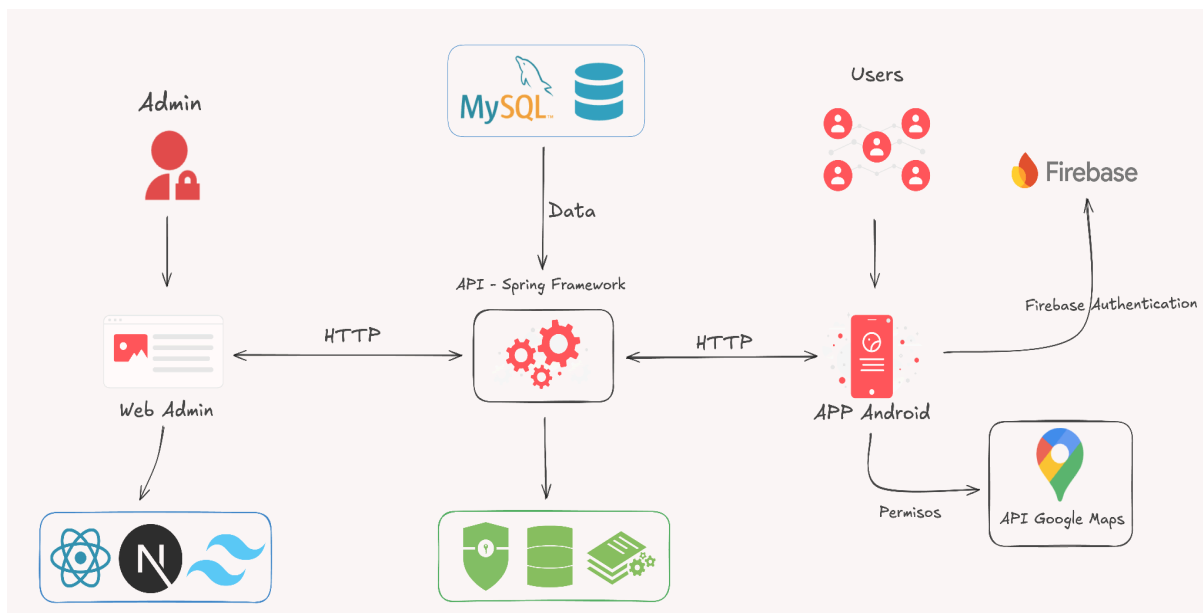
## 10) Estado de Pedido



## Diseño de la Arquitectura de Software

El sistema está compuesto por dos aplicaciones: una **aplicación móvil (Android)** y una **aplicación web para administradores**, ambas conectadas a una **API RESTful desarrollada con Spring Framework**.

- **Base de Datos:** Toda la información se almacena en una base de datos relacional **MySQL**, consumida exclusivamente a través de la API.
- **Aplicación Web (Admin):** Construida con **React, Next.js y TailwindCSS**, permite a los administradores gestionar la información del sistema mediante solicitudes HTTP a la API.
- **Aplicación Móvil (Usuarios):** Se conecta a la API vía HTTP y utiliza **Firebase Authentication** para la gestión de usuarios y autenticación segura. Además, hace uso de la **API de Google Maps** para funcionalidades basadas en ubicación y permisos relacionados.



## **Detalles de todas las herramientas a utilizar**

- **Nuestro Stack Tecnológico**

Elegimos tecnologías y herramientas que nos permitan ser ágiles y flexibles:

### **Gestión de proyecto:**

- Trello
- GitHub
- Google Meet

### **Herramientas:**

- Firebase
- Firebase Cloud Messaging
- Android Studio
- Figma
- Excalidraw
- IntelliJ IDEA

### **Frontend:**

- Kotlin
- Next.js
- Tailwind CSS
- JavaScript

### **Backend**

- Kotlin
- MySQL
- Java + Spring Framework
- Docker
- Warp - Terminal

## Presupuesto de la aplicación

El siguiente presupuesto detalla los costos estimados para el desarrollo del sistema de pedidos de **"Allan Pizza"**. Se han considerado aspectos clave como el tiempo de desarrollo, el equipo de trabajo, los recursos tecnológicos necesarios y posibles costos adicionales en caso de futuras mejoras. Este presupuesto busca ser accesible y realista, alineándose con las necesidades del negocio y asegurando una solución eficiente tanto para clientes como empleados.

Categoría	Detalle	Costo Estimado (USD)
Desarrollo de Software	Diseño y programación de la app web, app móvil en Kotlin y API	\$2,000
Firebase Cloud Messaging (Free)	Envío de notificaciones push	\$0
Infraestructura y Hosting	Hosting, dominio, base de datos (Vercel, Firebase).	\$300
Publicar en la Play Store	Cuenta de desarrollador (pago único)	\$25
Diseño UI/UX	Interfaces gráficas y prototipado en Figma	\$300
Pruebas y Depuración	Pruebas funcionales, de usabilidad y en dispositivos reales.	\$200
Soporte y Mantenimiento (opcional)	Posibles mejoras o nuevas funcionalidades después de la entrega.	\$200+
Total Estimado		\$3,025

## Cronograma de Trabajo

Para asegurar que el proyecto de Allan Pizza avance de manera organizada y se cumplan los plazos, hemos creado un cronograma detallado. Este divide el trabajo en etapas claras, como las interfaces de Figma, el desarrollo del backend y frontend y las pruebas finales, para poder entregar nuestro proyecto final. Cada fase tiene fechas específicas, lo que nos permitirá mantener el control del progreso y entregar la aplicación a tiempo.

A continuación, se muestra el calendario con las tareas y plazos definidos.

## CRONOGRAMA DE PROYECTO

[illegible]

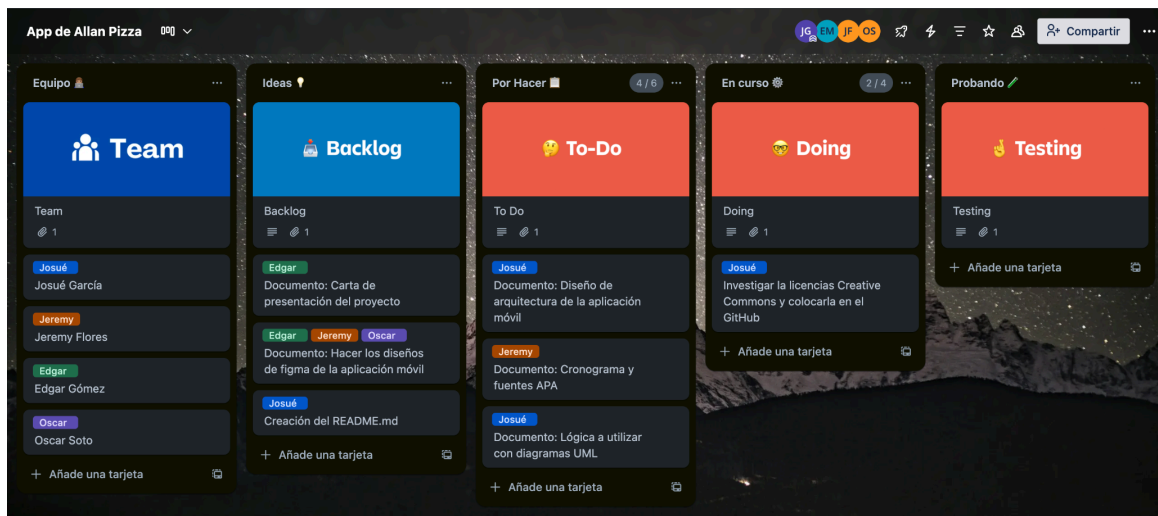


## Anexos y Documentación de Apoyo

Para complementar el desarrollo del proyecto, se han recopilado documentos y recursos clave que respaldan la planificación, implementación y funcionamiento del sistema.

1. Enlace de tablero de trello donde utilizamos KanBan como metodología ágil:

<https://trello.com/b/5muz30RT/app-de-allan-pizza>



2. Enlace al repositorio de GitHub donde está toda la estructura del proyecto y donde estará subido el código fuente del la aplicación móvil:

<https://github.com/Josueduardev/app-allan-pizza>

3. Enlace al repositorio de GitHub donde está toda la lógica y el desarrollo de la API (backend) para consumirla después en la aplicación móvil y web:

<https://github.com/Josueduardev/backend-allan-pizza>

4. Enlace del prototipo de figma de todas las pantallas y la interacción entre ellas:

<https://www.figma.com/design/tQVRV9Z5Ho1sGg9oUooo3M/DSS-MOCKUP-ALLAN'S-PIZZA>

## Referencias Bibliográficas

*Deed - Atribución/Reconocimiento-NoComercial 4.0 Internacional - Creative Commons.*

(s. f.). <https://creativecommons.org/licenses/by-nc/4.0/deed.es>

*Firebase Authentication.* (s. f.). Firebase.

<https://firebase.google.com/docs/auth?hl=es-419>

*Firebase Cloud Messaging.* (s. f.). Firebase.

<https://firebase.google.com/docs/cloud-messaging?hl=es-419>

Infobae. (2023, 3 enero). Cuánto cuesta publicar una aplicación en Google Play Store y App Store. *Infobae*.

<https://www.infobae.com/america/tecno/2023/01/03/cuanto-cuesta-publicar-una-aplicacion-en-google-play-store-y-app-store/>

*Mermaid.* (s. f.). Diagramming And Charting Tool. <https://mermaid.js.org/>

*Papa John's El Salvador - Ordenar pizza online.* (s. f.). Papa Johns el Salvador.

<https://www.papajohns.com.sv/>