

Aliro

Sistema de gestión de citas para locales

Trabajo de Fin de Máster

Manuel García de la Torre

CONTENIDO

01 Monivación

Contexto, problema

02 Desarrollo del proyecto

Tecnologías, metodología,
planificación.

03 Desarrollo de la aplicación

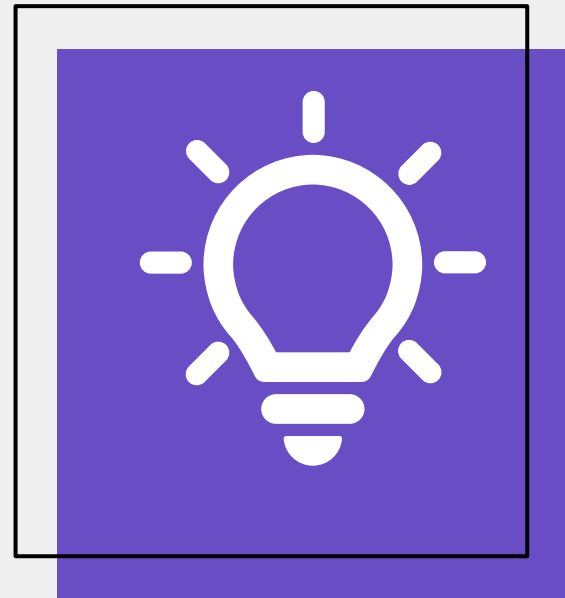
Desarrollo y diseño de las
capas. Trabajos futuros.

04 Conclusiones

Solución de los problemas
y conclusiones

¿Por qué Aliro?

- ❑ Buscando un ámbito poco digitalizado.
- ❑ Se detectó que la mayoría de locales que visitaba utilizan un sistema de citas tradicional.
- ❑ Muchos de ellos tenían problemas derivados de la gestión de citas:
 - ❑ Sobre aforo en algunas horas.
 - ❑ Pérdida de clientes por no poder coger el teléfono.
 - ❑ Mala previsión.



Existen soluciones pero...

- ❑ Grandes costes de desarrollo.
- ❑ Necesidad de contratar licencias con permanencia.
- ❑ Necesidad de contratar otros servicios adicionales.
- ❑ Tener que disponer de hardware específico.



Objetivos y requisitos



Objetivo principal

Desarrollo de una plataforma web para la gestión de citas.



Objetivo principal

Aplicar conocimiento adquirido en el máster.



Objetivos secundarios



Requisito R-001

Adaptable a diferentes tipos de resolución.



Requisito R-002

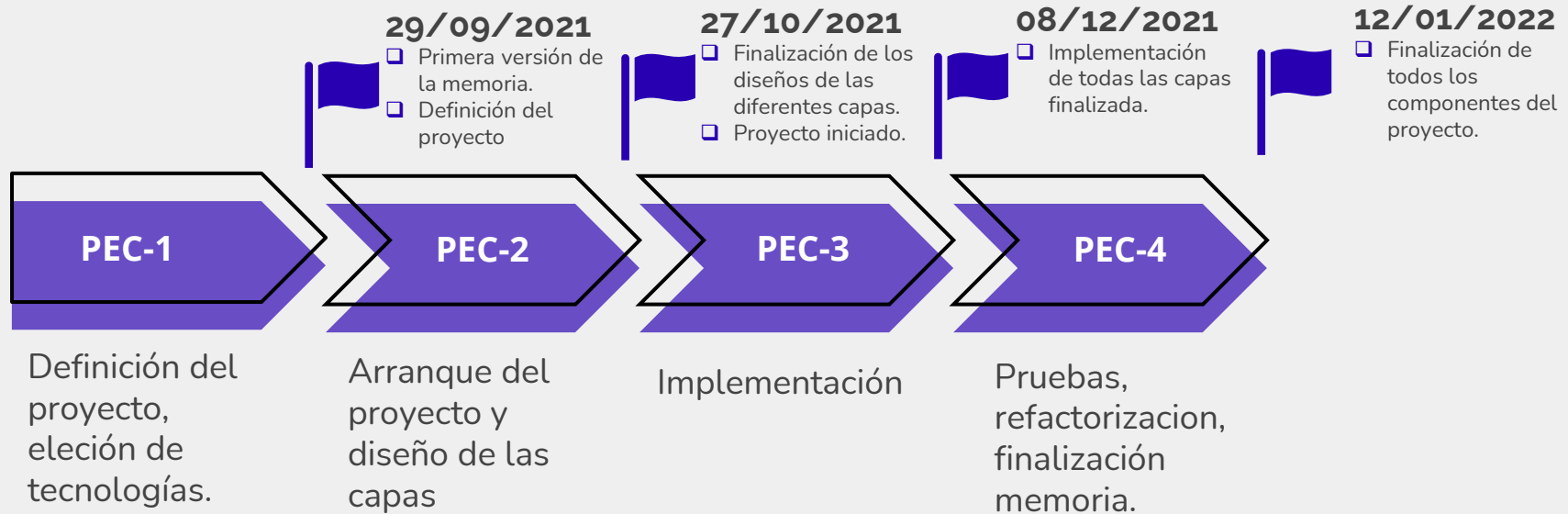
Acceso a las utilidades de la aplicación en base a suscripciones prepagos.



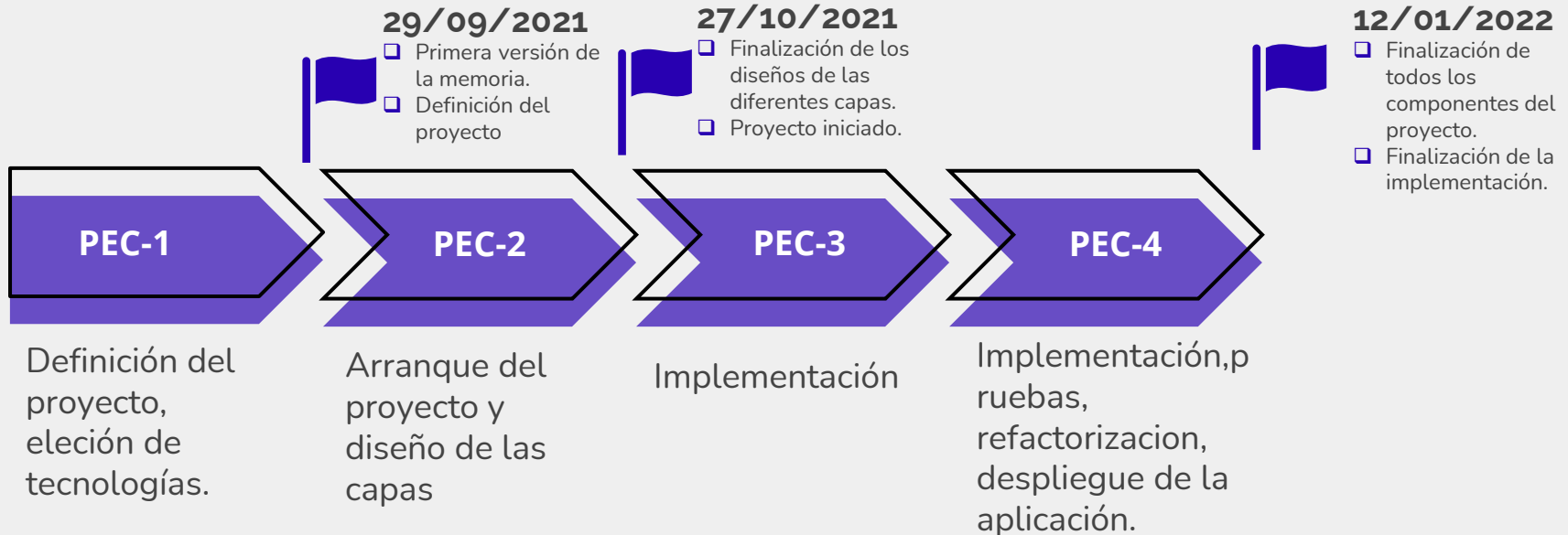
Requisito R-003

Diseño moderno, simple y funcional.

Planificación Inicial

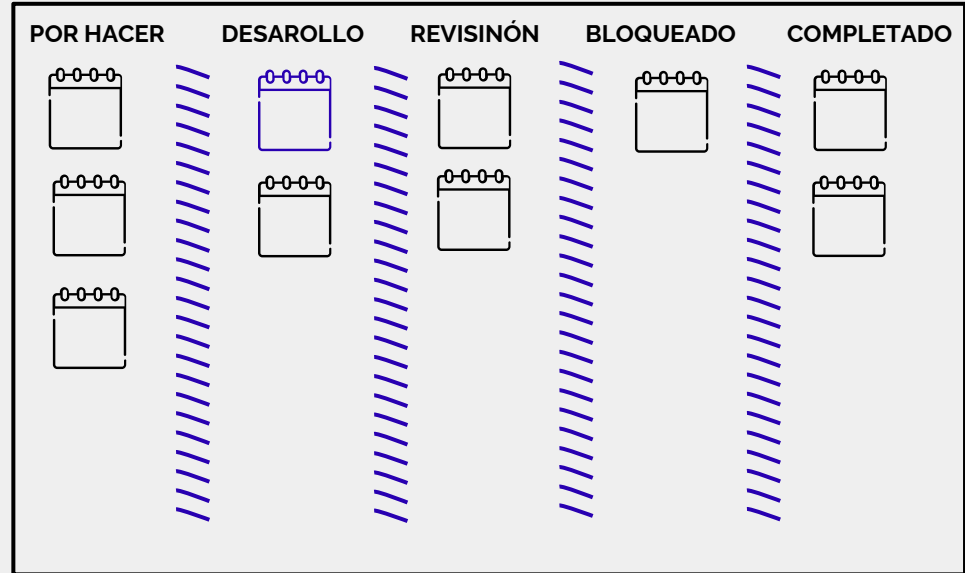


Planificación Real



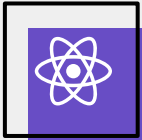
Metodología

- Metodologías ágiles.
- No se puede aplicar SCRUM.
- Kanban.
- Control de versiones en GIT.
- Backlog por cada PEC.
- Tareas especiales de investigación.



Tecnologías

- ❑ Tecnologías, actuales que ayudan al cumplimiento de los requisitos.



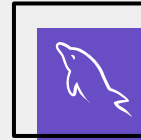
ReactJS

Librería de JavaScript, de código abierto desarrollada por Facebook.



Git

Gestor de versiones distribuido.



MySQL

Gestor de base de datos OpenSource.



Spring

Framework de uso libre para el desarrollo de aplicaciones empresariales en Java



Stripe

Plataforma de gestión de pagos online. En este caso gestor de las suscripciones.



ExpressJS

Framework de uso libre de JS para el desarrollo e API REST.

¿Quién va interactuar con la Web?



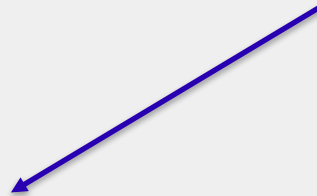
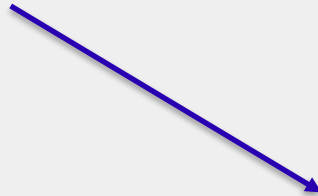
Dueño/gerente del local

Actor que se encarga de gestionar las citas por parte del local



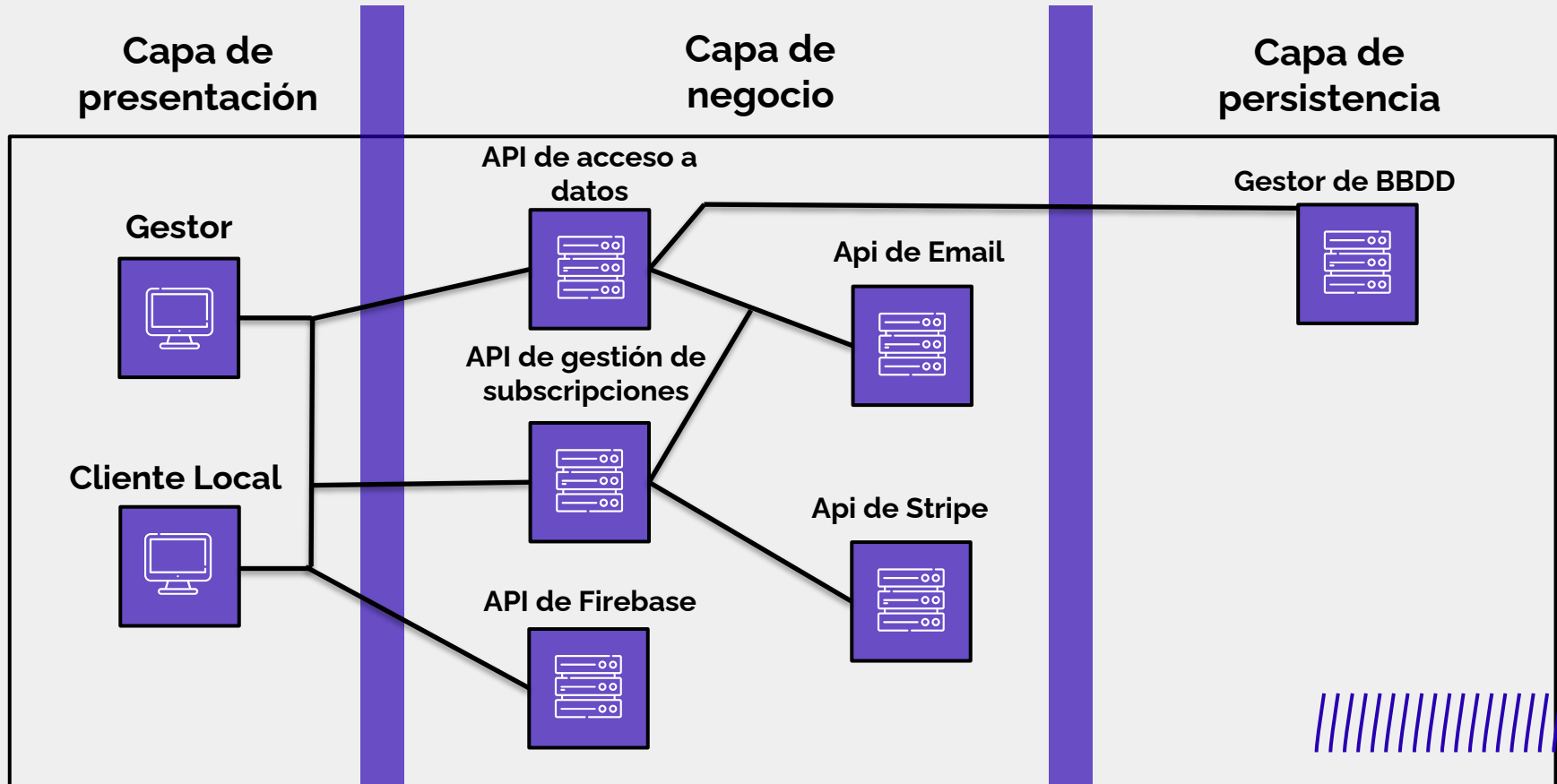
Cliente del local

Actor que desea coger citas y gestionarlas.



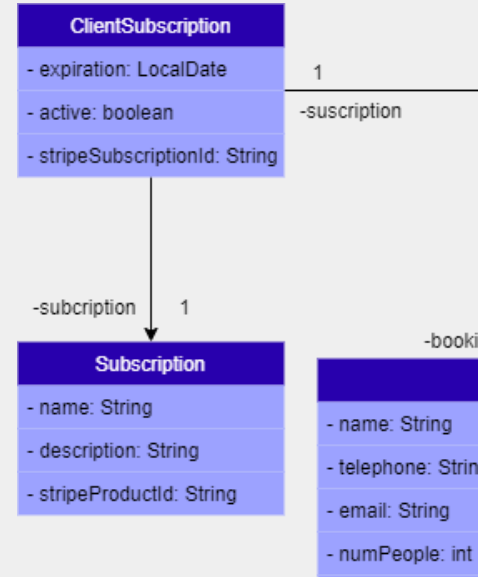
Casos de uso

Diseño arquitectónico en tres capas



Capa de persistencia

- ❑ Diseño del dominio con un diagrama de clases en UML.
- ❑ Implementación en Java con Spring y JPA.
- ❑ Anotaciones de JPA que describen las relaciones.
- ❑ MySQL
- ❑ Necesidad de crear un disparador para evitar sobre aforos.





Capa de negocio

APIs Propias

- ❑ Hay tres APIs propias desarrolladas para Aliro y conectadas entre sí.
- ❑ Una API realizada en Java con Spring, que es la de acceso a datos y la principal de la lógica de negocios. Ej: registrar una cita.
- ❑ Dos APIs desarrolladas en NodeJS con ExpressJS:
 - ❑ Una para gestión de las suscripciones y intermediaria con Stripe. Ej: crear un checkout.
 - ❑ La otra, un gestor de email que envía mensajes a los clientes. Ej: Se ha confirmado una cita en un local.

Capa de negocio

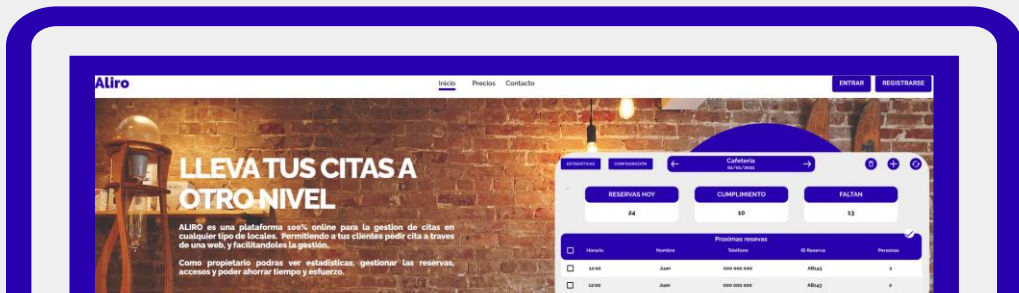
APIs de terceros

- ❑ Dos APIs de terceros son utilizadas para el funcionamiento de Aliro:
 - ❑ **Firebase:** Servicio de Google para el desarrollo de webs y aplicaciones. Proporciona a Aliro, el sistema de autenticación de usuarios y de almacenamiento de imágenes.
 - ❑ **Stripe:** Plataforma de gestión de pagos en línea, dispone de varios servicios. En Aliro se utiliza el servicio de suscripciones, el cual que gestiona los pagos recursivos, la facturación y la gestión de la suscripción.



Capa de presentación

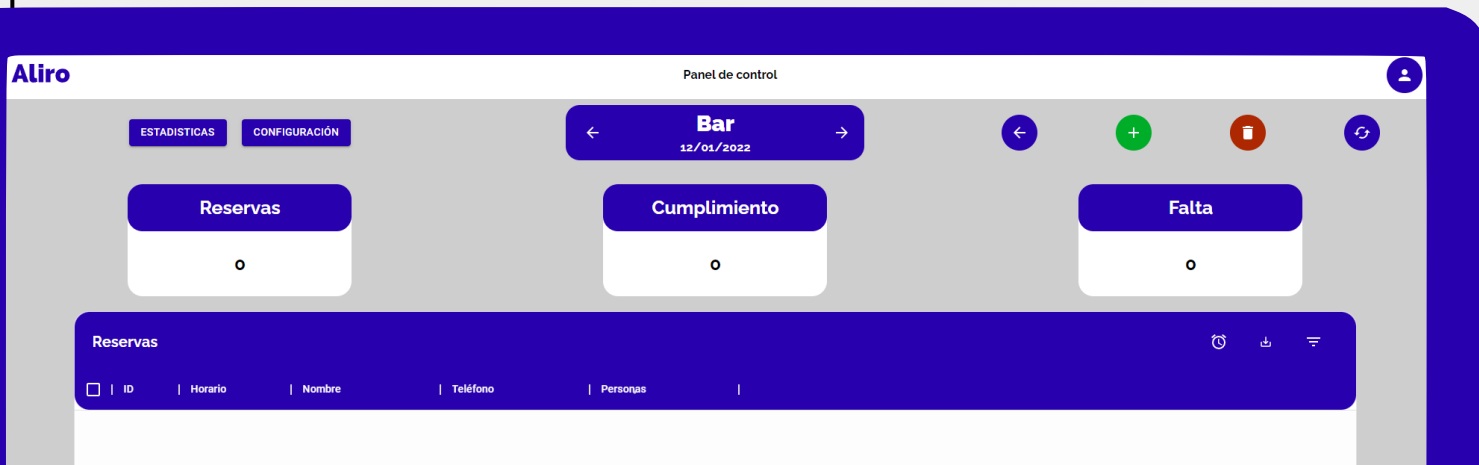
- ❑ Prototipo y bocetos realizados con Figma.
- ❑ Se ha utilizado la herramienta Anima para agilizar el desarrollo, ayudando a pasar los estilos del boceto a la realidad.
- ❑ Desarrollado con React y CSS.
- ❑ Web adaptable a diferentes resoluciones.
- ❑ Utilización de componentes de terceros como Bootstrap o Material UI.



Aliro

Capa de presentación

- ❑ Desarrollo de componentes propios, para conseguir el resultado requerido. Ej: selector de hora y día de la cita.
- ❑ Adaptación de componentes de terceros tanto en funcionalidad como apariencia.
- ❑ Interacción con la capa de negocio a través de llamadas AJAX.



Aliro

Despliegue

- ❑ Se ha desplegado la aplicación a modo demostración en un servidor VPS con sistema operativo Ubuntu.
- ❑ Se ha configurado el servidor e instalado los programas necesarios.
- ❑ Se generan los diferentes archivos de despliegue con la configuración del ambiente de producción. Ej: ".env.production"
- ❑ Se utilizan herramientas propias para el despliegue de cada tecnología, Tomcat, PM2 y Nginx.



Aliro

Trabajos futuros

- ❑ La aplicación es funcional, pero sería recomendable realizar algunas acciones:
 - ❑ Agregar seguridad a las APIs.
 - ❑ Añadir certificados SSL para encriptar las conexiones.
- ❑ Otros aspectos que se tienen pensados realizar en un futuro:
 - ❑ Entrada y verificación de la entrada a través de un código QR, para maximizar los tiempos.
 - ❑ Añadir un sistema que envíe un día antes un recordatorio de las citas a los clientes
 - ❑ Sistema de configuración de horarios mas completo para suscripciones superiores.

¿Cómo se ha solucionado el problema?

- ❑ Los problemas detectados en los locales:
 - ❑ No poder coger el teléfono: **Pidiendo la cita a través de la web, sin necesidad de contactar directamente con el local.**
 - ❑ Mala organización: **Disponer de estadísticas, como el número de reservas para hoy o días sucesivos.**
 - ❑ Sobre aforo: **El sistema sólo permite el aforo permitido por cada rango de tiempo.**

¿Cómo se ha solucionado el problema?

- ❑ Los problemas detectados de otras aplicaciones existentes:
 - ❑ Grandes costes de desarrollo: **Es un servicio online, no específico para un comercio.**
 - ❑ Permanencia: **El servicio se da en base a suscripciones prepago que se pueden cancelar cuando se desee.**
 - ❑ Servicios combinadas: **No es necesario contratar otro servicio para disponer de Aliro.**
 - ❑ Tener Hardware específico: **Puede ser utilizado desde cualquier dispositivo con conexión a internet y un navegador.**

Aliro

Conclusión

Aliro

Conclusiones

- ❑ Se han cumplido los objetivos del Trabajo de Fin de Máster.
 - ❑ Se ha conseguido implementar la aplicación.
 - ❑ Se han aplicado conocimientos del máster.
- ❑ Utilidad de la metodología de la UOC.
- ❑ Adquirir nuevo conocimientos y experiencia.
- ❑ Satisfacción personal.

Fin

Aliro

