

Plataforma de gestión de citas y servicios para profesionales autónomos y PYMES.

PROYECTO INTEGRADO — PI_E1: Especificación y análisis

Ana Vertedor

Centro: Cesur Málaga Este
Ciclo Formativo: Desarrollo de Aplicaciones Web (DAW)
Año Académico: 2025-2026

Índice

1. Introducción

- 1.1 Descripción del proyecto
- 1.2 Objetivo general
- 1.3 Objetivos específicos
- 1.4 Alcance del sistema

2. Actores y roles

- 2.1 Identificación de actores
- 2.2 Privilegios y responsabilidades

3. Casos de uso / funcionalidades principales

- 3.1 Diagrama de casos de uso
- 3.2 Descripción de casos de uso

4. Requisitos del sistema

- 4.1 Requisitos funcionales (RF)
- 4.2 Requisitos no funcionales (RNF)

5. Arquitectura del sistema

- 5.1 Arquitectura cliente–servidor
- 5.2 Módulos internos
- 5.3 Servicios externos
- 5.4 Diagrama preliminar de arquitectura

6. Tecnologías previstas

- 6.1 Tecnologías frontend
- 6.2 Tecnologías backend
- 6.3 Base de datos
- 6.4 Pasarelas de pago
- 6.5 Herramientas de apoyo

7. Esquema visual / Wireframes

- 7.1 Mapa de navegación
- 7.2 Bocetos de interfaz (páginas principales)

8. Conclusión

1. Introducción

El presente documento desarrolla la especificación y análisis del proyecto seleccionado en la entrega inicial. La aplicación propuesta consiste en una plataforma web para la gestión de citas y servicios dirigida a profesionales autónomos y pequeñas empresas que trabajan mediante sesiones reservadas. Tras definir en la entrega anterior la motivación, el problema detectado y el enfoque general, en esta fase se profundiza en los requisitos del sistema, los actores implicados, los casos de uso principales, la arquitectura prevista y los primeros bocetos visuales necesarios para orientar el desarrollo técnico.

1.1 Descripción del proyecto

La solución propuesta consiste en una aplicación web para la gestión de citas y servicios profesionales, orientada a autónomos que trabajan con clientes mediante sesiones reservadas, como psicoterapeutas, coaches, consultores, diseñadores o desarrolladores freelance.

La plataforma permite que cada profesional configure sus servicios, defina su disponibilidad semanal y reciba reservas confirmadas por parte de sus clientes. El sistema integra pago online, gestión automática de citas y recordatorios por correo electrónico, facilitando así la organización del profesional y mejorando la experiencia del cliente.

1.2 Objetivo general

Crear una aplicación web orientada a profesionales que trabajan de forma individual con sus clientes, como psicoterapeutas, coaches o consultores. El sistema permitirá gestionar servicios, horarios, reservas, cobros y la facturación en un único entorno privado y configurable. La plataforma centraliza todo el proceso y automatiza tareas como los recordatorios, los pagos y las reglas de cancelación o cambio de cita, con el fin de reducir la carga administrativa del profesional y ofrecer un flujo de trabajo simple, claro y adaptable a sus necesidades reales.

1.3 Objetivos específicos

Los objetivos específicos del proyecto se detallan a continuación y definen las funciones principales que deberá cumplir la aplicación.

- Permitir que el profesional configure sus servicios y defina su disponibilidad semanal.
- Generar un sistema de reservas claro y fácil de utilizar para el cliente.
- Integrar un proceso de pago seguro que confirme la cita de forma automática.
- Automatizar el envío de recordatorios y notificaciones relacionadas con la sesión.
- Gestionar cobros, facturación básica y el historial de citas del profesional.
- Aplicar reglas de cancelación y reprogramación que eviten cambios de última hora.
- Ofrecer un entorno privado y adaptable a las necesidades de distintos perfiles profesionales.

1.4 Alcance del sistema

El proyecto se plantea como una primera versión básica que cubre el proceso principal que necesita un profesional autónomo para gestionar sus citas. El sistema abarcará la configuración de servicios, la definición de horarios, la generación de huecos disponibles y el proceso completo de reserva con pago integrado. También se incluirán la automatización de recordatorios, la facturación básica y la aplicación de reglas sencillas de cancelación o cambio de cita.

Las funciones avanzadas que puedan ampliarse en el futuro, como estadísticas detalladas, integración con varias pasarelas de pago, sistemas de mensajería interna o herramientas de marketing, no forman parte del alcance inicial y se considerarán como posibles mejoras posteriores.

El objetivo es entregar una plataforma sólida en sus funciones esenciales, que permita demostrar la lógica del sistema y su utilidad real sin añadir elementos que dificulten la implementación durante las primeras fases del desarrollo.

2. Actores y roles

Esta sección identifica a los participantes que intervienen en el sistema y describe sus funciones dentro de la plataforma.

2.1 Identificación de actores

Esta sección presenta a los participantes que intervienen en el sistema y su papel dentro del funcionamiento de la plataforma.

- **Profesional:**

Persona autónoma que ofrece servicios por cita. Puede configurar su perfil, definir su disponibilidad, gestionar sus servicios, consultar reservas y acceder al historial de cobros y facturación. También puede crear o modificar citas de forma manual cuando sea necesario.

- **Cliente:**

Usuario que contrata un servicio puntual del profesional. Puede consultar la disponibilidad, seleccionar una fecha y realizar el pago correspondiente para confirmar la cita. Recibe recordatorios y avisos relacionados con la sesión reservada.

- **Pasarela de pago:**

Servicio externo que se encarga de procesar el pago del cliente. Devuelve al sistema el resultado de la operación para confirmar o rechazar la reserva.

- **Servicio de correo:**

Componente externo responsable del envío de recordatorios, confirmaciones y avisos de cancelación o cambio de cita. Permite automatizar la comunicación entre el sistema y los usuarios.

2.2 Privilegios y responsabilidades

Cada actor del sistema tiene un papel concreto y unas acciones determinadas dentro de la plataforma. A continuación se describen de forma sencilla sus funciones y el tipo de tareas que puede realizar.

- **Profesional**

El profesional utiliza un panel privado para gestionar su actividad diaria y mantener organizada su agenda.

- Configura sus servicios y establece su disponibilidad
- Confirma, modifica o cancela citas desde su panel
- Revisa las reservas y el historial de sesiones
- Añade o edita citas manualmente cuando lo necesita
- Consulta los cobros realizados y la facturación básica

- **Cliente**

El cliente accede a la parte pública del sistema para reservar una cita y recibir la información necesaria.

- Consulta la disponibilidad del profesional
- Selecciona un servicio, reserva una hora y realiza el pago
- Recibe confirmaciones y recordatorios
- Solicita cambios o cancelaciones según las normas del profesional

- **Pasarela de pago**

Este servicio externo se encarga exclusivamente de procesar el pago y devolver el resultado al sistema.

- Valida el pago del cliente
- Envía al sistema la confirmación o el error correspondiente

- **Servicio de correo**

El módulo de correo se utiliza para automatizar las comunicaciones importantes.

- Envía confirmaciones de reserva
- Genera recordatorios previos a la cita

- Notifica cambios o cancelaciones

3. Casos de uso / funcionalidades principales

El sistema se organiza en torno a una serie de acciones que pueden realizar tanto el profesional como el cliente. Cada caso de uso describe una interacción concreta con la plataforma y ayuda a entender cómo funcionará el sistema en la práctica. A continuación se presenta el diagrama general y, después, la descripción detallada de cada caso.

3.1 Diagrama de casos de uso

El siguiente esquema resume de forma visual las acciones principales que realizan los actores dentro de la plataforma. Sirve como vista rápida antes de entrar en la descripción individual de cada caso de uso.

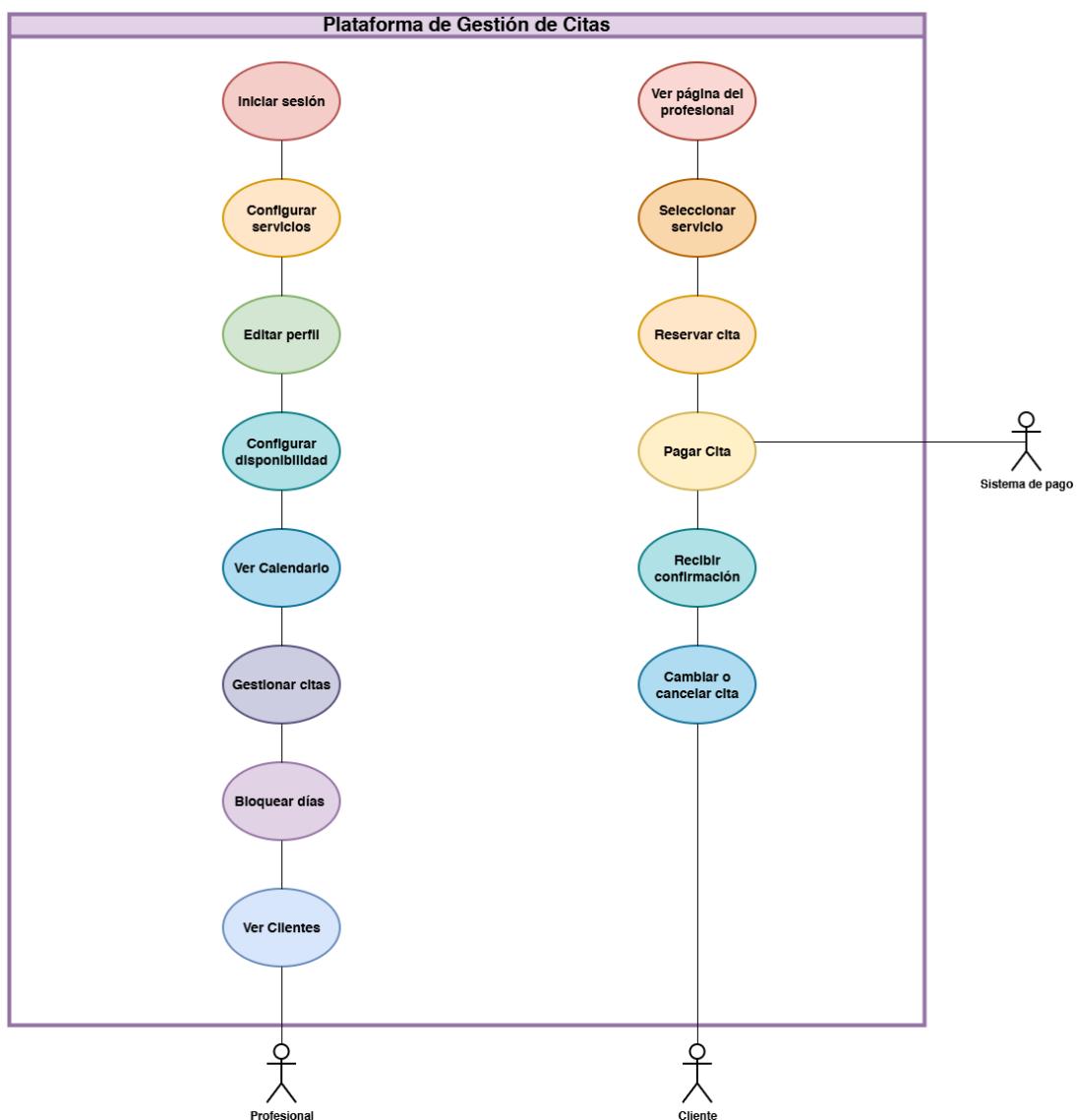


Fig. 1. Diagrama general de los casos de uso principales del sistema.

3.2 Descripción de casos de uso

A continuación se describen los principales casos de uso del sistema de gestión de citas para profesionales autónomos y pequeñas empresas.

- **CU1 · Registrarse como profesional**
 - *Actor:* Profesional
 - El profesional crea una cuenta proporcionando sus datos básicos.
 - *Resultado:* El sistema registra al nuevo usuario y habilita el acceso al panel.
- **CU2 · Iniciar sesión**
 - *Actor:* Profesional
 - El profesional introduce sus credenciales para acceder a su panel.
 - *Resultado:* Acceso correcto si los datos coinciden con los registrados.
- **CU3 · Editar perfil profesional**
 - *Actor:* Profesional
 - Permite modificar la información pública visible para los clientes.
 - *Resultado:* Perfil actualizado en la plataforma.
- **CU4 · Gestionar servicios**
 - *Actor:* Profesional
 - Permite crear, actualizar o desactivar servicios ofrecidos.
 - *Resultado:* Servicios visibles y disponibles para reserva.
- **CU5 · Configurar disponibilidad**
 - *Actor:* Profesional
 - Define horarios, días disponibles, descansos y bloqueos específicos.
 - *Resultado:* El sistema genera los huecos reservables en función de esta configuración.
- **CU6 · Ver calendario de citas**
 - *Actor:* Profesional
 - Consulta de próximas citas y su detalle, con filtros básicos.
 - *Resultado:* Calendario actualizado y accesible.
- **CU7 · Cancelar o modificar una cita**
 - *Actor:* Profesional
 - Permite anular o modificar citas ya registradas.
 - *Resultado:* La cita cambia de estado y el cliente recibe un aviso.

- **CU8 · Consultar perfil del profesional**
 - *Actor:* Cliente
 - Acceso a la página pública del profesional sin registro.
 - *Resultado:* Visualización de servicios y disponibilidad general.
- **CU9 · Reservar una cita**
 - *Actor:* Cliente
 - El cliente selecciona servicio, fecha y horario, e introduce sus datos.
 - *Resultado:* La cita queda registrada y se envía la confirmación por correo.
- **CU10 · Realizar un pago (modo test)**
 - *Actor:* Cliente
 - Tras reservar, el cliente completa el pago mediante la pasarela configurada.
 - *Resultado:* El sistema actualiza la cita según el estado enviado por la pasarela
- **CU11 · Cancelar una cita como cliente**
 - *Actor:* Cliente
 - Permite cancelar la cita dentro de las normas marcadas por el profesional.
 - *Resultado:* La cita se anula y el profesional recibe un aviso.
- **CU12 · Envío de notificaciones**
 - *Actor:* Sistema
 - Envía automáticamente correos de confirmación, recordatorios y avisos de cambios.
 - *Resultado:* Cliente y profesional reciben los mensajes correspondientes.

4. Requisitos del sistema

Antes de comenzar el desarrollo, es necesario definir qué debe hacer el sistema y qué condiciones debe cumplir. Los siguientes requisitos recogen el comportamiento esperado de la plataforma y los criterios mínimos de calidad.

4.1 Requisitos funcionales (RF)

En esta sección se detallan las funciones que debe ofrecer la plataforma para cumplir su propósito.

- **RF1:** El sistema permite que un profesional se registre y acceda a su panel privado.
- **RF2:** El profesional puede crear, modificar y desactivar los servicios que ofrece.
- **RF3:** El profesional puede definir su disponibilidad semanal y los bloqueos puntuales.

- **RF4:** El sistema genera los huecos de agenda en función de la disponibilidad configurada.
- **RF5:** El profesional puede consultar, modificar o cancelar citas.
- **RF6:** El cliente puede ver el perfil del profesional y consultar sus servicios.
- **RF7:** El cliente puede reservar una cita seleccionando servicio, fecha y hora.
- **RF8:** El cliente puede realizar el pago a través de la pasarela configurada.
- **RF9:** El sistema envía correos automáticos de confirmación, recordatorios y avisos de cambios.
- **RF10:** El sistema registra el historial de reservas y cobros.

4.2 Requisitos no funcionales (RNF)

Los requisitos no funcionales describen las condiciones de calidad que debe cumplir la plataforma durante su uso.

- **RNF1:** La plataforma debe ser accesible desde cualquier navegador moderno.
- **RNF2:** El sistema debe mantener tiempos de carga razonables incluso con varias reservas simultáneas.
- **RNF3:** Los datos personales y de pago deben tratarse de forma segura.
- **RNF4:** La interfaz debe ser clara y fácil de usar para profesionales y clientes.
- **RNF5:** La aplicación debe poder mantenerse y ampliarse sin reescribir la base completa del sistema.

5. Arquitectura del sistema

En este apartado se describe la estructura general del sistema y la forma en que los distintos componentes interactúan entre sí.

5.1 Arquitectura cliente-servidor

La aplicación utiliza una arquitectura cliente-servidor sencilla, pensada para separar la interfaz del usuario de la lógica interna y del acceso a los datos. Este enfoque facilita el mantenimiento y permite ampliar partes del sistema sin afectar a las demás.

El flujo básico funciona así:

- El profesional o el cliente acceden a la plataforma desde un navegador web.
- El navegador envía peticiones al servidor a través de la API.
- El servidor procesa la información, aplica reglas de negocio y consulta la base de datos.

- El servidor devuelve al cliente la información necesaria para mostrar cada pantalla.

5.2 Módulos internos

El backend se organiza en módulos que separan la lógica del calendario, los pagos y las notificaciones, lo que facilita el mantenimiento y la ampliación del sistema.

Cada módulo cumple una función concreta dentro del servidor:

- **Gestión de agenda:** controla la disponibilidad del profesional, genera los huecos reservables y actualiza el estado de cada cita.
- **Gestión de servicios:** permite crear, modificar o desactivar los servicios que ofrece el profesional.
- **Gestión de pagos:** recoge la información del pago, la envía a la pasarela y registra el resultado en la plataforma.
- Módulo de notificaciones: envía correos automáticos de confirmación, recordatorio y cambios de cita.

5.3 Servicios externos

Para completar algunas funciones de la plataforma, el sistema se apoya en servicios externos que realizan tareas especializadas que no se gestionan directamente desde el backend.

Estos servicios intervienen de la siguiente manera:

- Pasarela de pago: procesa el pago realizado por el cliente y devuelve al sistema el estado de la operación.
- Servicio de correo: envía los mensajes automáticos relacionados con la reserva, el recordatorio y cualquier cambio o cancelación.

5.4 Diagrama preliminar de arquitectura

Para mostrar de forma visual cómo se relacionan los distintos componentes del sistema, se incluye un esquema sencillo que resume el papel del cliente, el servidor, la base de datos y los servicios externos.

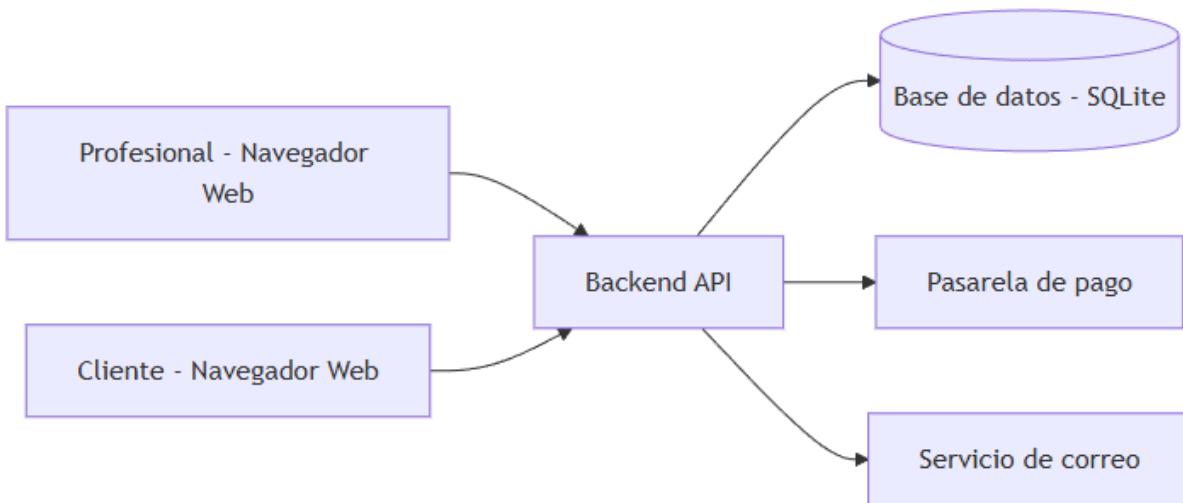


Fig. 2. Esquema general de la arquitectura básica del sistema.

6. Tecnologías previstas

La selección tecnológica se ha definido a partir del análisis inicial realizado en la fase E0. En aquella etapa se consideraron varias alternativas tanto en frontend como en backend. En esta fase se detallan las tecnologías que se emplearán en la primera versión del proyecto, dejando abierta la posibilidad de ampliaciones si fueran necesarias.

6.1 Tecnologías frontend

La interfaz comenzará con una implementación ligera en HTML, CSS y JavaScript para obtener una versión inicial estable y funcional. Este enfoque permite avanzar con rapidez en las primeras iteraciones y mantener el desarrollo centrado en la lógica y el flujo de la aplicación.

Desde el inicio se plantea una arquitectura preparada para evolucionar hacia un frontend más avanzado. En fases posteriores, y especialmente cuando la plataforma alcance una versión beta con más interacción y dinamismo, podrá integrarse un framework como React sin necesidad de rehacer la base del sistema.

Tecnologías previstas para la fase inicial:

- HTML para la estructura de las páginas
- CSS para el estilo y la maquetación
- JavaScript para la interacción básica con la interfaz

Evolución prevista:

- Integración de un frontend en React cuando la plataforma alcance una versión beta que requiera mayor dinamismo, componentes reutilizables y una interacción más fluida.

6.2 Tecnologías backend

En E0 se valoraron dos posibilidades principales: Node.js con Express o Python con FastAPI. Finalmente he optado por Python debido a la claridad de su ecosistema para construir APIs de forma rápida, mantenible y bien documentada.

Tecnologías previstas:

- Python como lenguaje principal
- Flask o FastAPI para la creación de la API
- Control de rutas, validación y reglas de negocio desde el servidor

El backend será también el encargado de comunicarse con las pasarelas de pago y de gestionar el proceso de confirmación de citas.

6.3 Base de datos

La base de datos comenzará utilizando SQLite, que permite avanzar con rapidez en las primeras fases del desarrollo y realizar iteraciones sin necesidad de configurar un servidor dedicado. Esta elección encaja con la arquitectura inicial del proyecto y facilita las pruebas y el despliegue temprano.

Desde el diseño de los modelos de datos se tendrá en cuenta la futura migración a PostgreSQL. Cuando la plataforma alcance fases más avanzadas (por ejemplo, una versión beta con mayor volumen de citas, usuarios y reglas de negocio), la base podrá escalarse a PostgreSQL sin modificar la lógica del sistema.

Tecnología prevista:

- SQLite como motor de base de datos integrado

Evolución prevista:

- Migración a PostgreSQL en fases posteriores para mejorar concurrencia, rendimiento y capacidad de crecimiento

6.4 Pasarelas de pago

El sistema incluirá un proceso de pago integrado, en línea con la planificación descrita en E0. Durante esta fase del proyecto, se trabajará en modo test, sin transacciones reales, pero empleando las mismas herramientas que se utilizarían en un entorno productivo.

Pasarelas previstas:

- PayPal Checkout (sandbox)
- Stripe (modo test)

Ambas opciones permiten simular pagos reales de forma segura y estable, y cuentan con una API compatible con el backend en Python. Estas integraciones pueden activarse en modo real si el proyecto evoluciona más allá del prototipo académico.

6.5 Herramientas de apoyo

Además de las tecnologías principales, se utilizarán herramientas complementarias para el desarrollo y la gestión del proyecto.

Herramientas previstas:

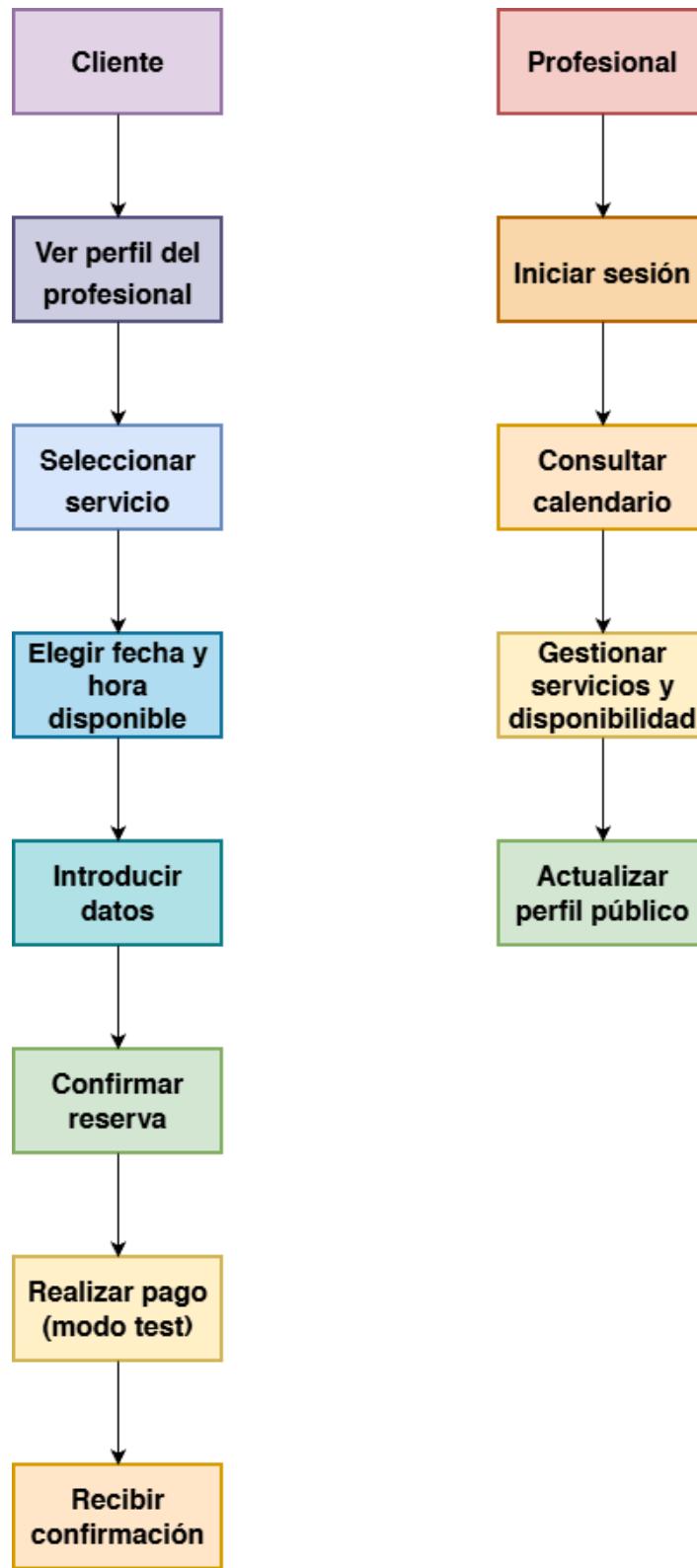
- GitHub como repositorio de documentación
- GitHub Desktop para el control de versiones y la organización del repositorio
- Draw.io y mermaidchart.com para diagramas
- Google Docs para la redacción de entregas
- Entorno virtual de Python (venv) para gestionar dependencias
- Visual Studio Code como editor principal.

7. Esquema visual / Wireframes

Esta sección recoge una primera representación visual del funcionamiento de la plataforma y de sus principales pantallas. Su objetivo es ofrecer una visión clara del flujo de navegación y de la estructura inicial de la interfaz antes de avanzar al diseño definitivo.

7.1 Mapa de navegación

El mapa de navegación muestra cómo se relacionan las distintas páginas de la plataforma y qué caminos puede seguir un usuario desde el inicio hasta completar una acción importante.

*Fig. 3. Resumen del flujo de navegación del sistema.*

Este esquema permite identificar las rutas principales del cliente y del profesional, así como los puntos clave del recorrido del usuario dentro del sistema.

7.2 Bocetos de interfaz

Los siguientes bocetos muestran la estructura básica de las pantallas principales de la plataforma. Son representaciones esquemáticas que ayudan a visualizar la organización de la información antes de trabajar en el diseño final.

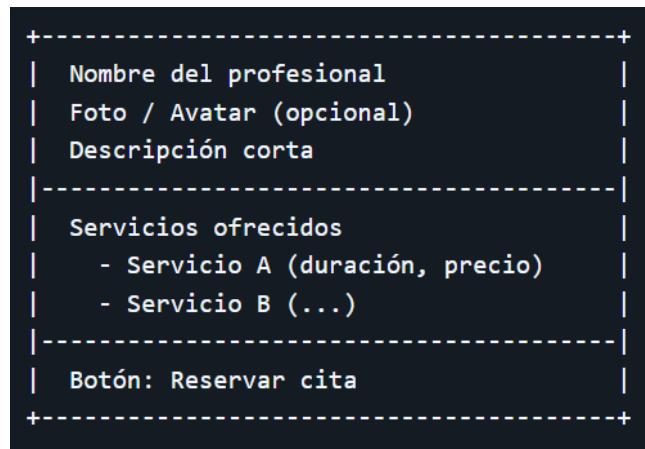


Fig. 4. Página pública del profesional (boceto)

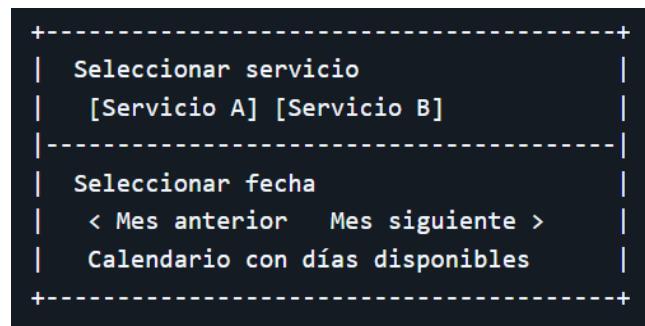


Fig. 5. Selección de servicio y fecha (boceto)

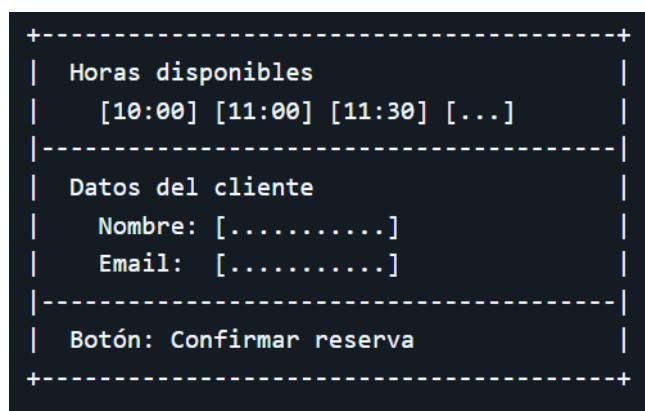
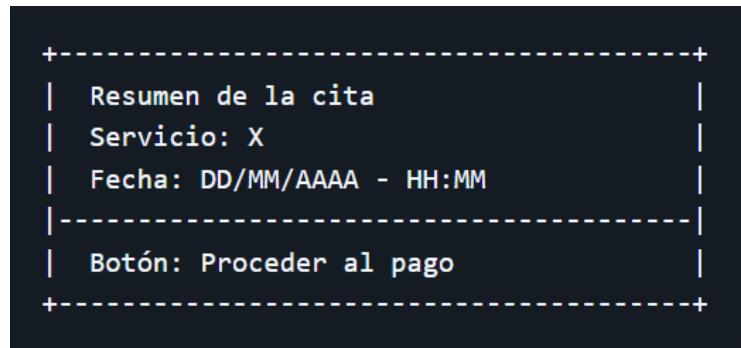
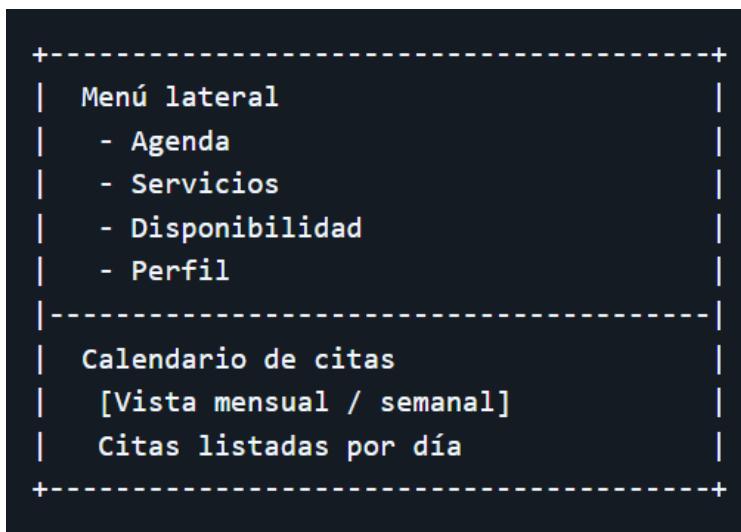


Fig. 6. Selección de hora y datos del cliente (boceto)

*Fig. 7. Pantalla de pago (boceto)**Fig. 8. Panel del profesional (boceto)***Nota:**

Los bocetos mostrados arriba, se han extraído directamente del repositorio de mi proyecto en GitHub, donde se mantienen las versiones actualizadas de la documentación y los wireframes en formato Markdown.

- <https://github.com/nerineaoi/tfg-plataforma-citas-docs>

8. Conclusión

Esta primera fase ha permitido definir con claridad el alcance del sistema, los actores implicados y las funcionalidades clave que deberá cubrir la plataforma. También se han establecido los requisitos que guiarán su desarrollo, junto con una propuesta inicial de arquitectura y tecnologías. Con esta base ya estructurada, el proyecto puede avanzar hacia la fase de implementación con una visión coherente y bien fundamentada.