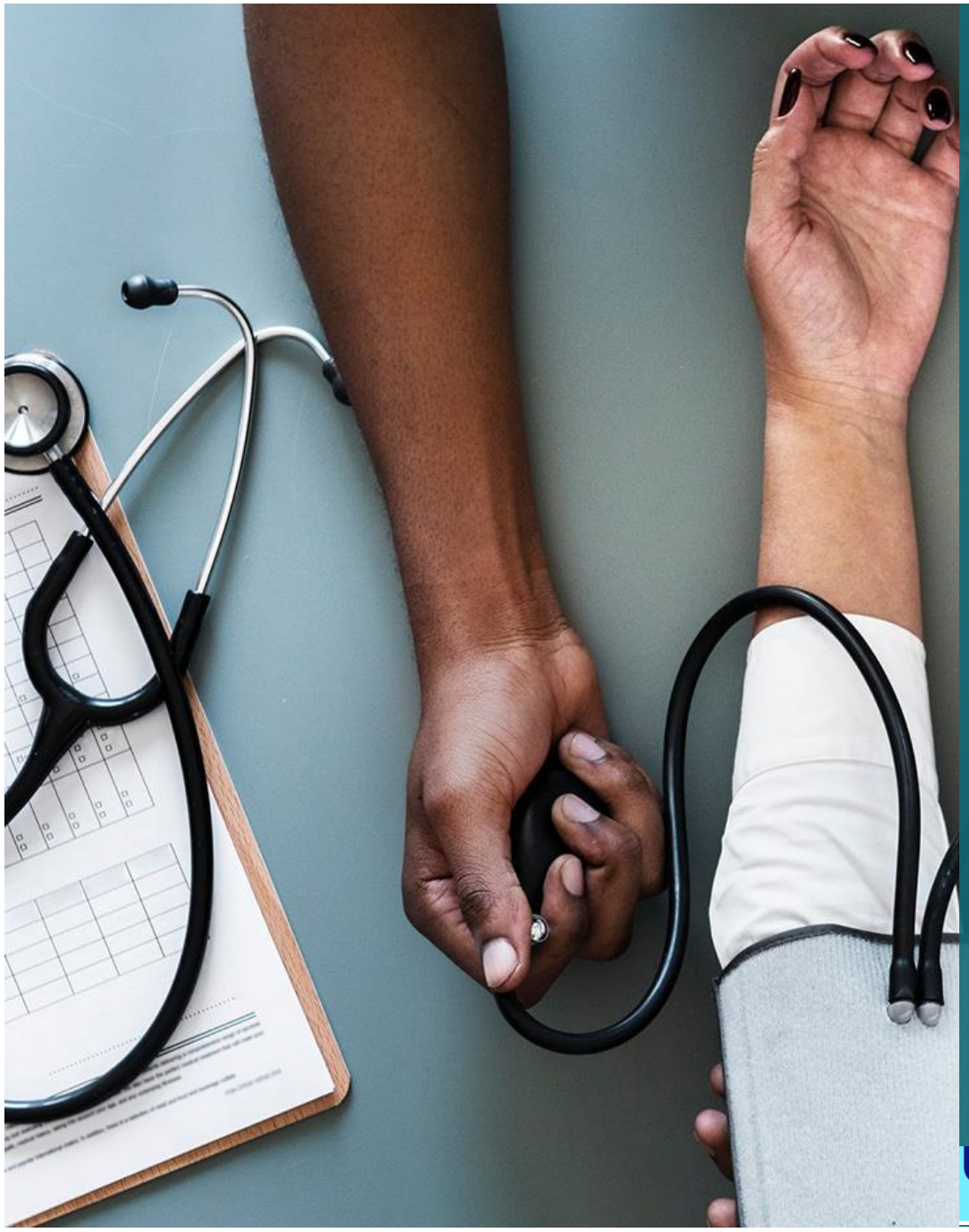


Bless You

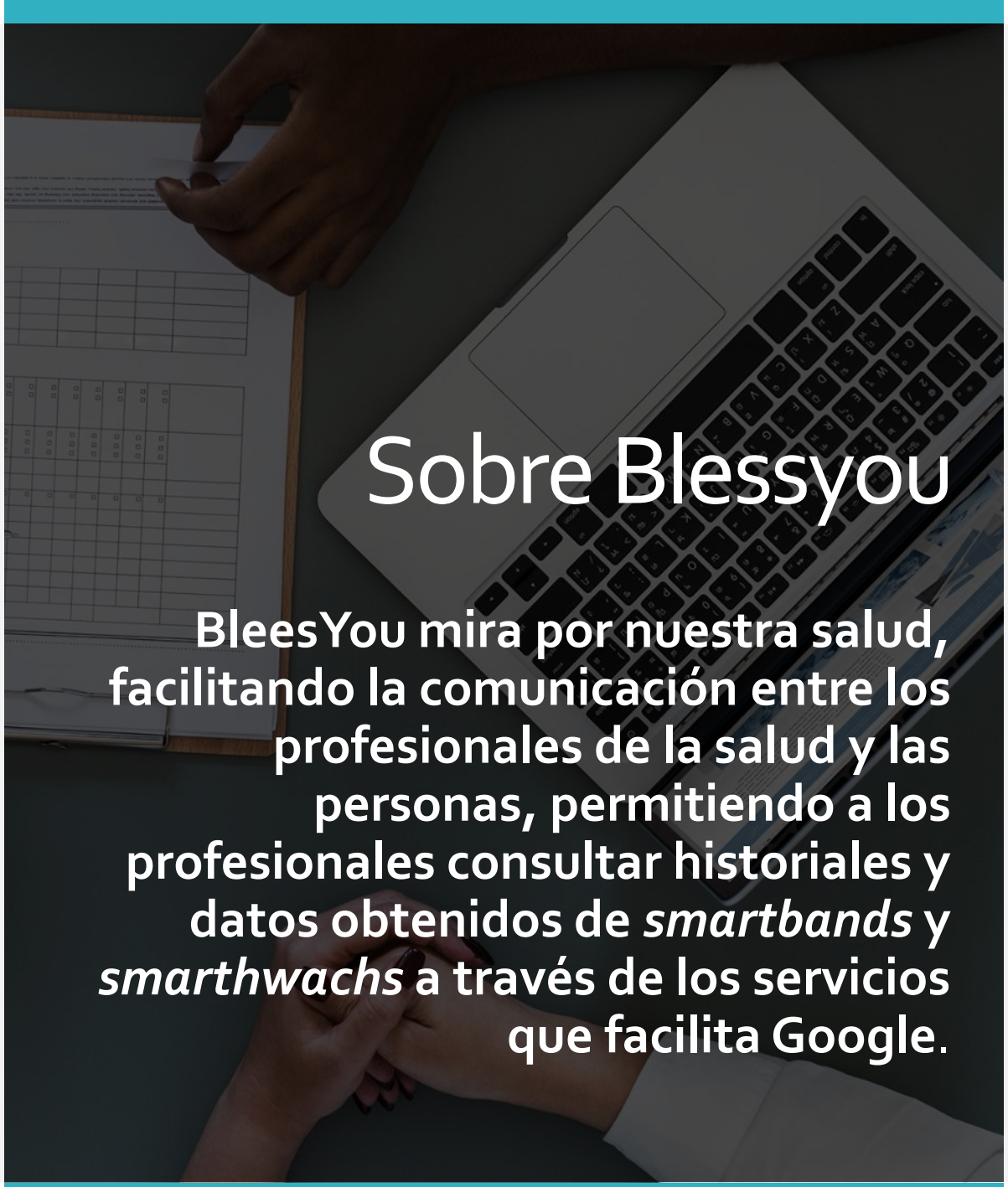
TFG Desarrollo web

Paloma Terán Pérez

Universitat Oberta de Catalunya

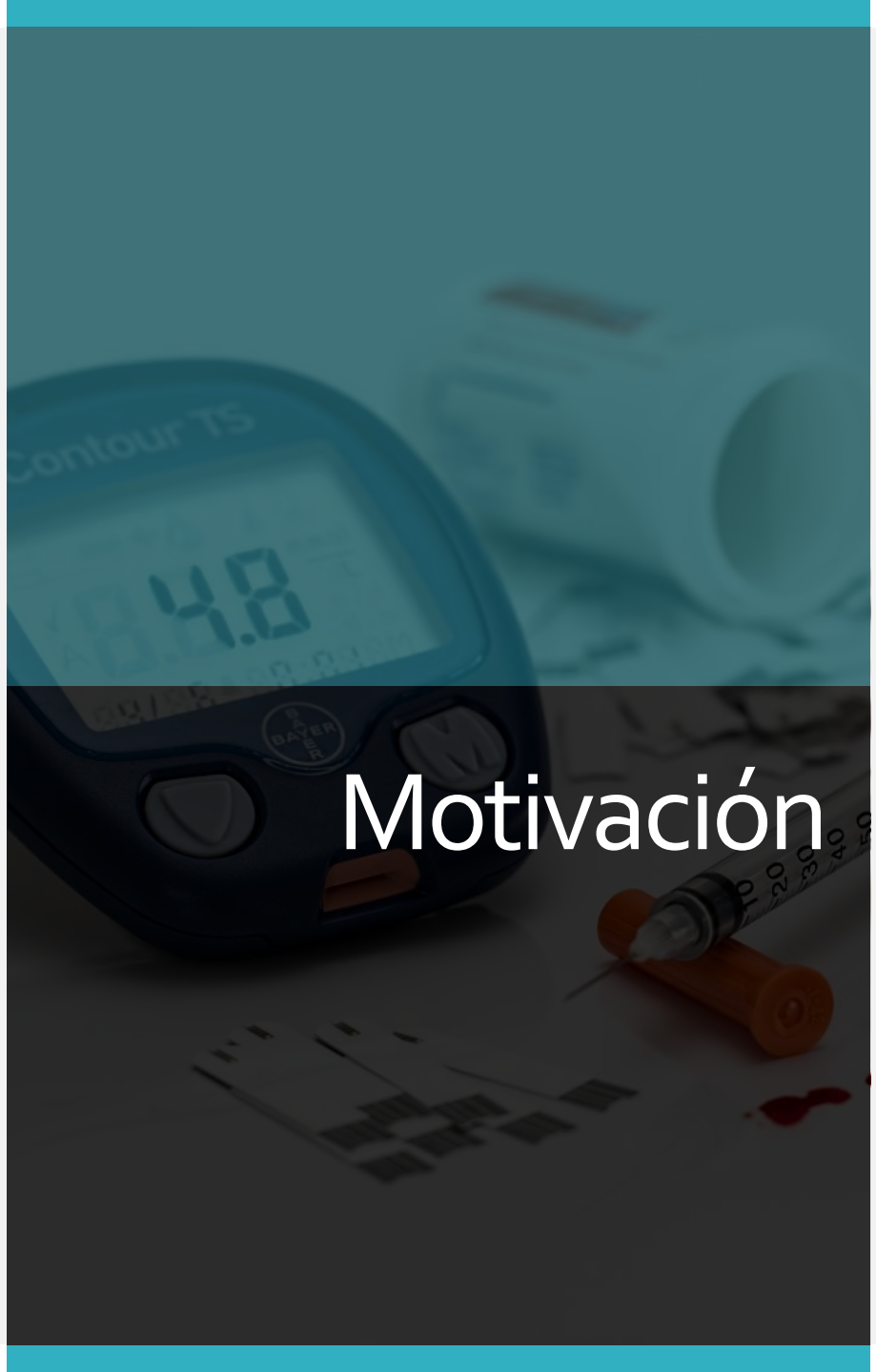
uoc.edu

Índice



Sobre Blessyou

BleesYou mira por nuestra salud, facilitando la comunicación entre los profesionales de la salud y las personas, permitiendo a los profesionales consultar historiales y datos obtenidos de *smartbands* y *smarthwachs* a través de los servicios que facilita Google.



Motivación

Digitalización

Transformación digital en auge, lenta en algunos sectores.

Covid

La pandemia del Covid obliga a buscar nueva formas de trabajo.

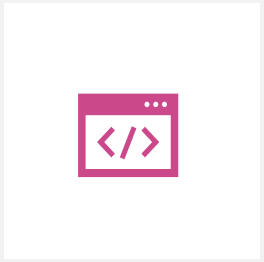
Nube

Información de *smartwachs* y *smartbands* en Google

Objetivos principales

Análisis

Analizar el desarrollo una de aplicación web.



Desarrollo

Desarrollo de una aplicación web.

Conexión

Conectar personas con profesionales de la salud de forma online.

Salud

Integrar los datos de Google Fit en la aplicación.

Documentar

Documentar el desarrollo de la aplicación.

Enfoque y método

Java

Lenguaje orientado a objetos semicompilado para backend.

Angularjs

Framework MVC de JavaScript para el frontend.

MySQL

Sistema de gestión de bases de datos relacionales.

Trello

Metodología ágil Kanban.

Documentación del análisis

Documentación de los objetivos funcionales que se quieren alcanzar en el desarrollo.

Requisitos

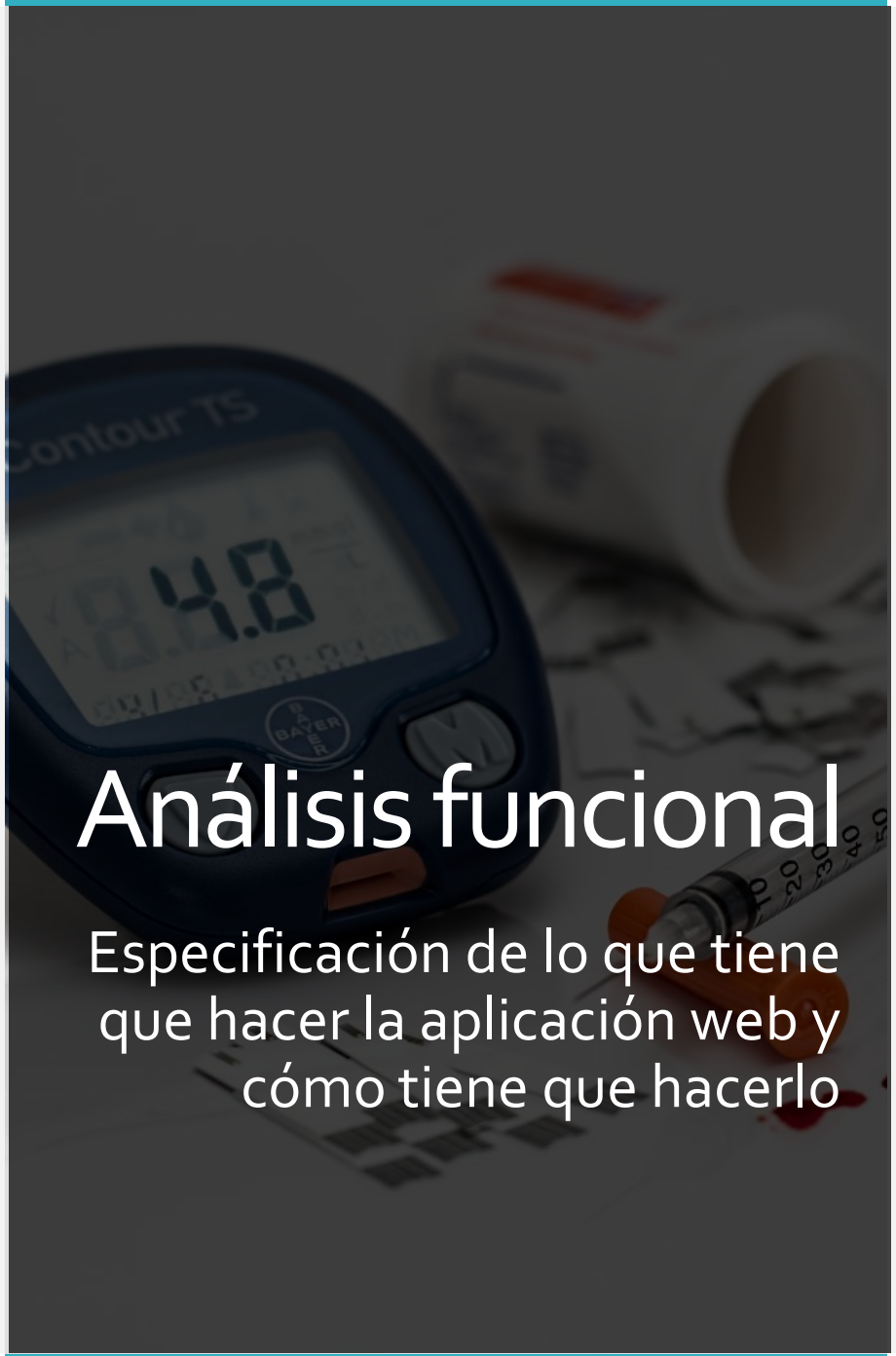
Tabla de requisitos funcionales y no funcionales

Casos de uso

Diagrama de las actividades que deberá realizar alguien o algo para llevar a cabo un proceso

Mocks

Prototipado de las pantallas que se van a desarrollar en la aplicación.



Análisis funcional

Especificación de lo que tiene que hacer la aplicación web y cómo tiene que hacerlo

Bless You

Paloma Terán

Proyectos

Blessyou-web

Contiene los controladores para los servicios necesarios del front y el propio front de la aplicación.

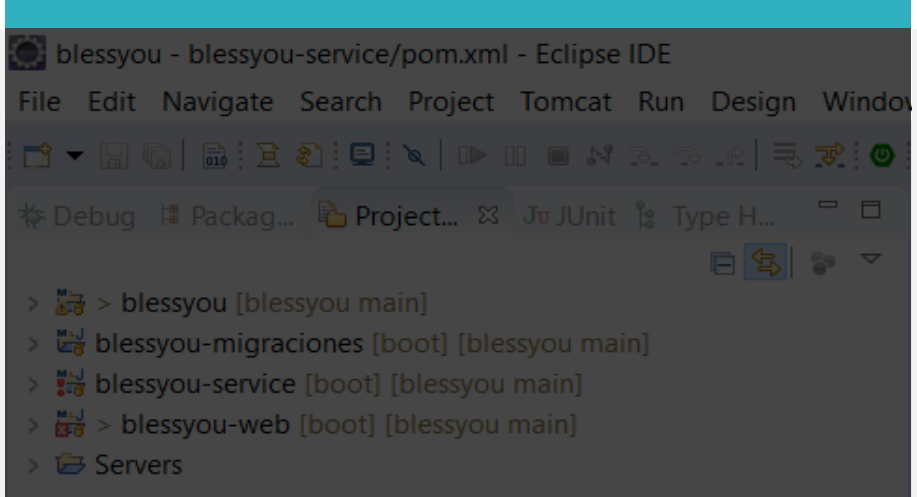
Blessyou-services

Contiene toda la lógica de negocio de la aplicación:

- Entidades y DTOs
- Proxies y servicios
- Mappers
- Repositorios

Blessyou-migraciones

Contiene los scripts SQL necesarios para la actualización de la aplicación.



Diseño

Arquitectura de la aplicación

Librerías utilizadas

Contenido en blessyou-web

Bootstrap

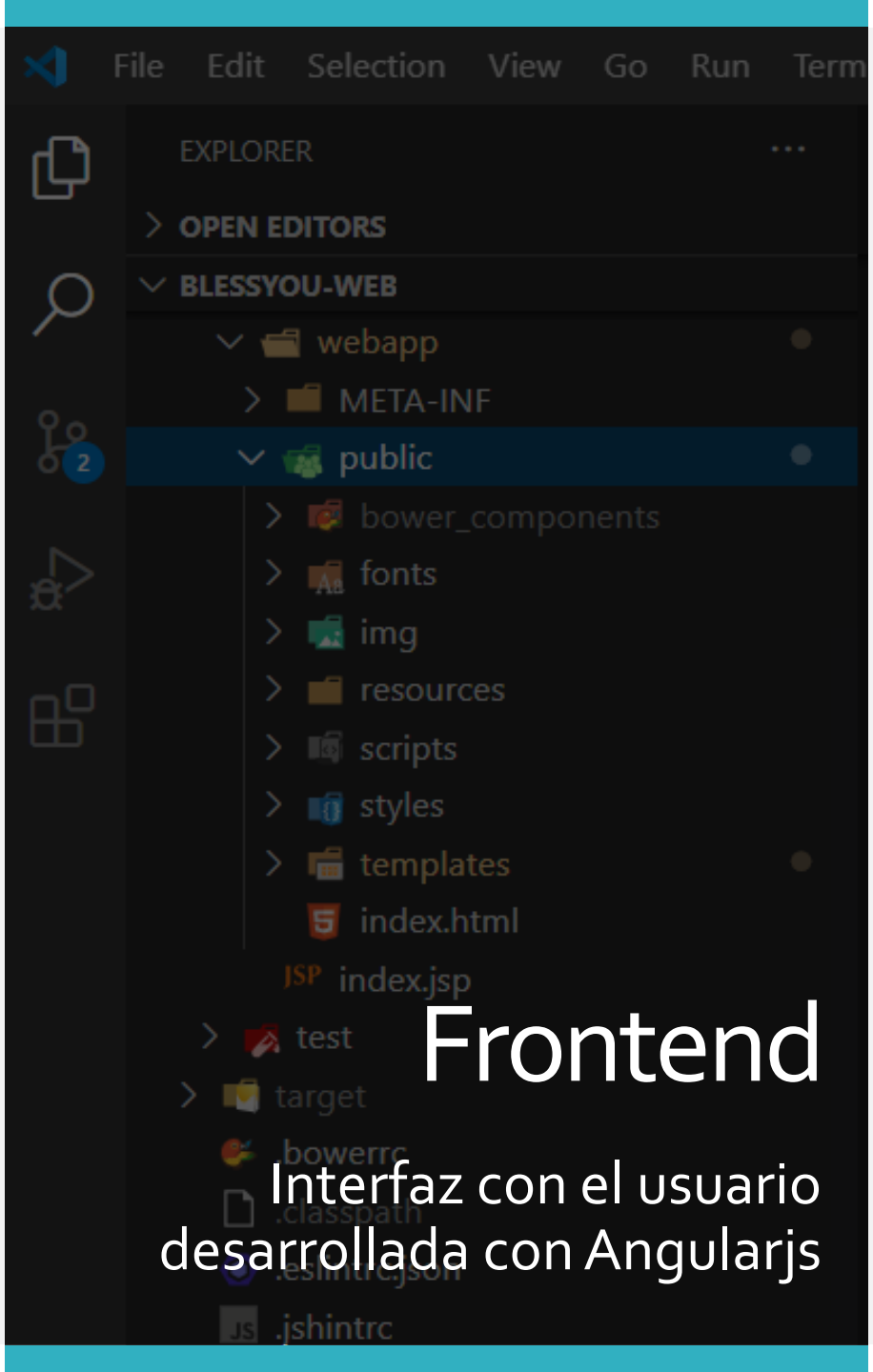
Kit de herramientas de código abierto para desarrollos web responsive con HTML, CSS y JavaScript

Font awesome

Framework utilizado para sustituir imágenes de iconos por gráficos vectoriales convertidos en fuentes

Chartjs

Biblioteca JavaScript gratuita de código abierto para la visualización de datos



Frontend

Interfaz con el usuario desarrollada con Angularjs

Frameworks y herramientas utilizadas

Maven

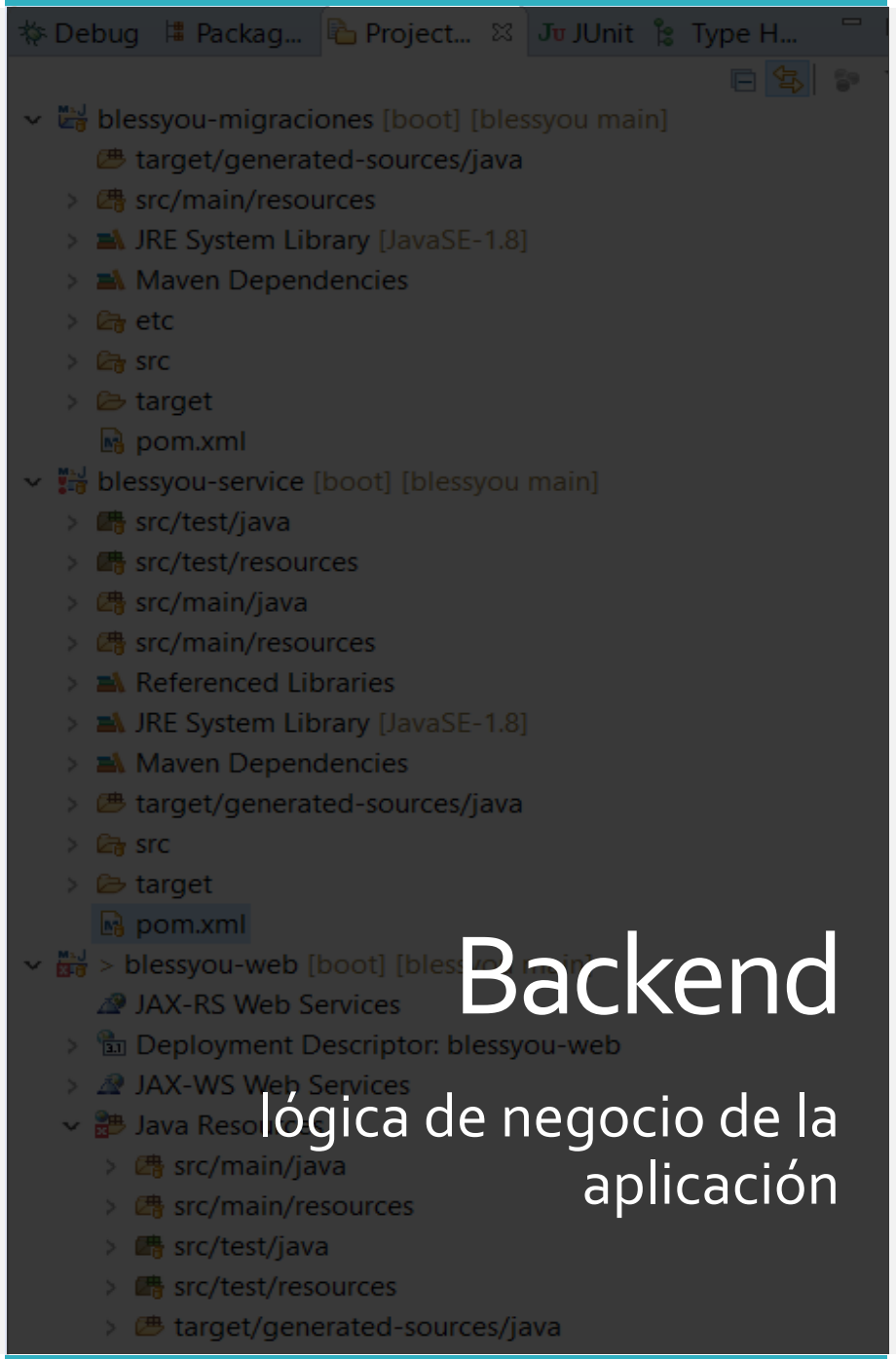
Herramienta de software para la gestión y construcción de proyectos Java.

Spring

Spring es un framework para el desarrollo de aplicaciones.

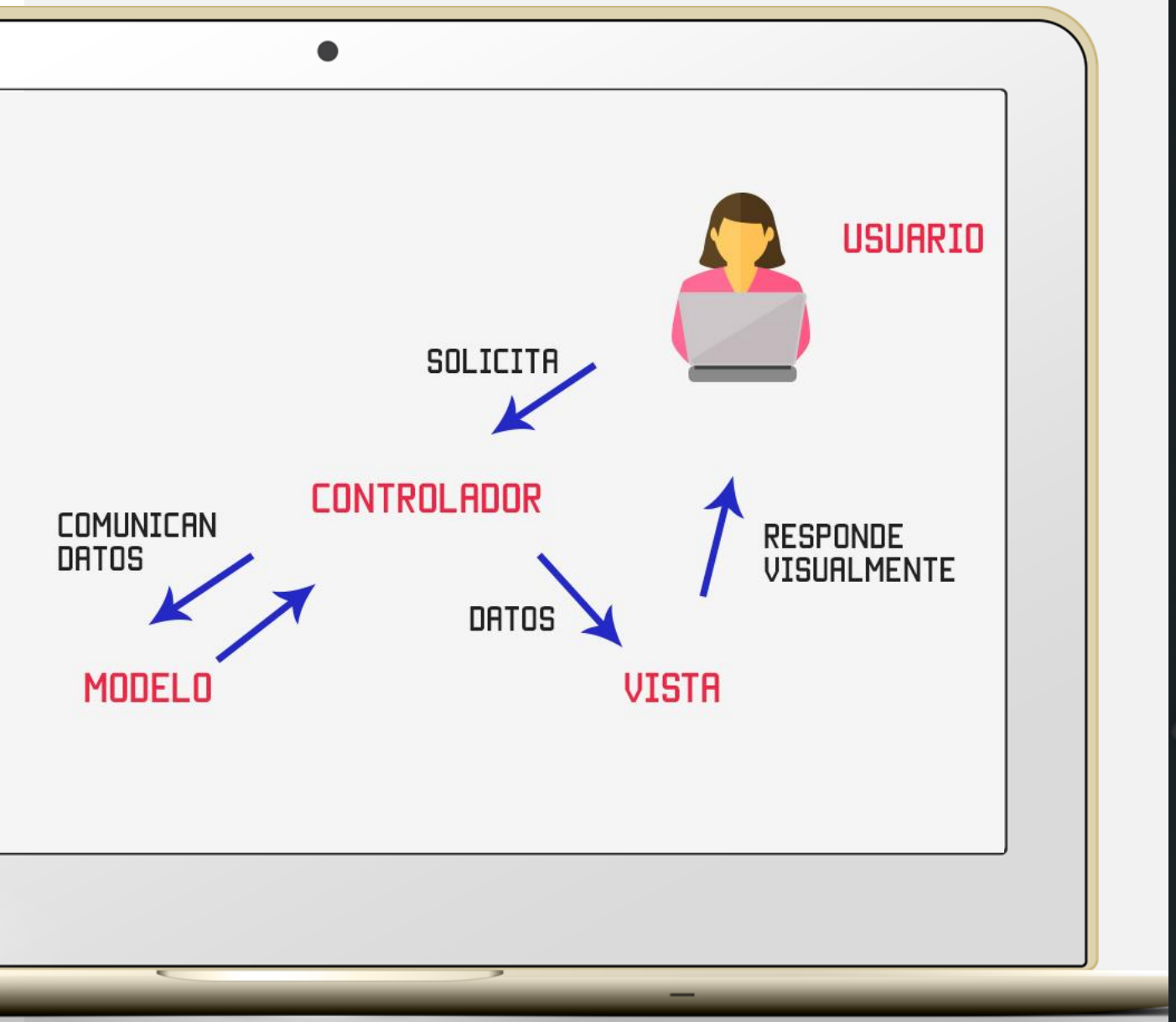
Hibernate

Herramienta de mapeo objeto-relacional para la plataforma Java.



Backend

lógica de negocio de la aplicación



Modelo Vista Controlador (**MVC**)

Estilo de arquitectura de software (patrón) que separa en tres componentes distintos:

- los datos de una aplicación
- la interfaz de usuario
- y la lógica de control

Otros patrones

Patrón Data Transfer Object (DTO)

- Crea de objetos planos (POJO o DTO) con una serie de atributos.
- Estos POJOS pueden ser enviados o recuperados del servidor en una sola invocación.
- Un DTO puede contener información de múltiples fuentes o tablas y concentrarlas en una única clase simple.

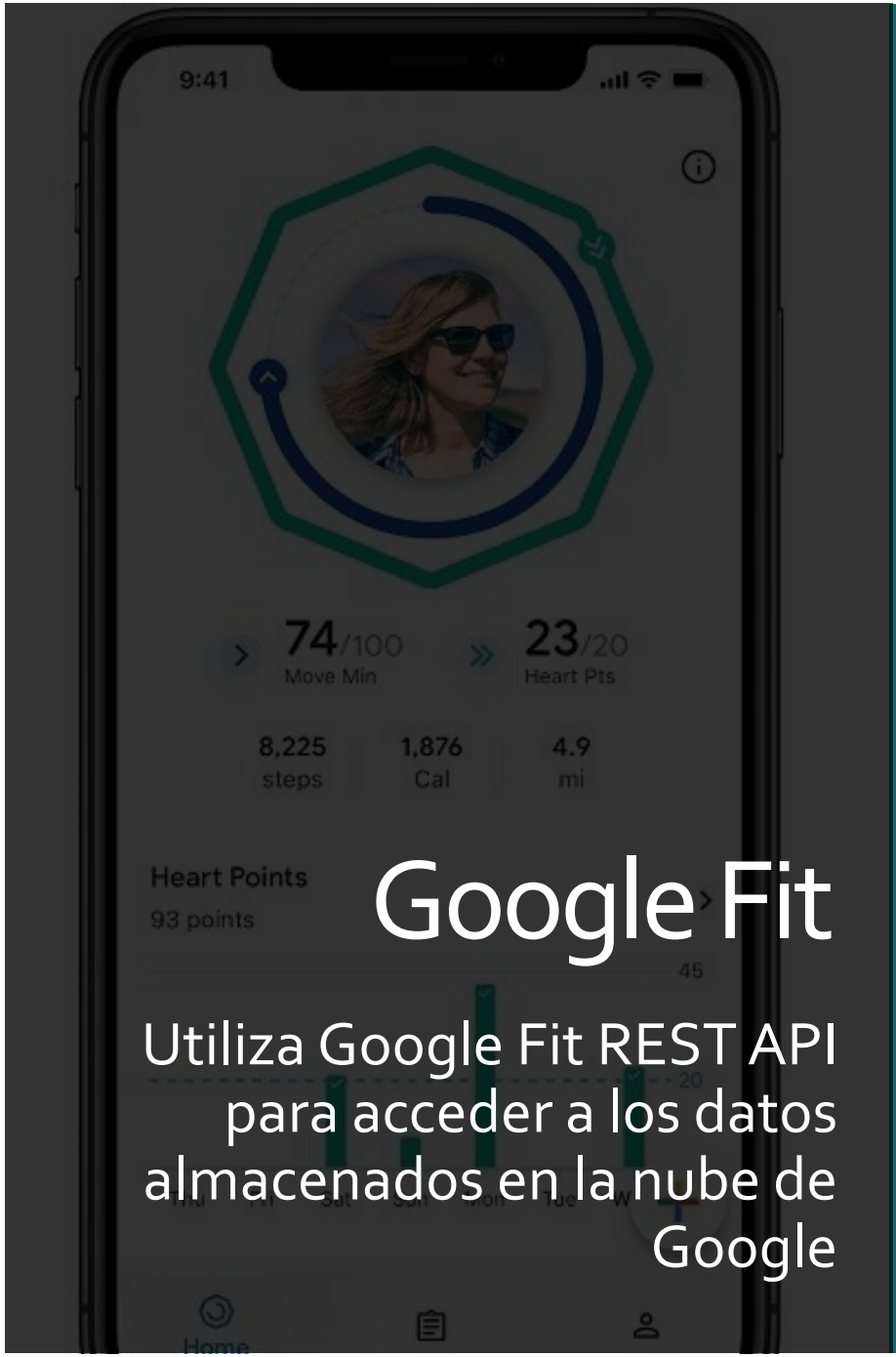
Patrón Proxy

- Patrón estructural.
- Proporciona un subrogado o intermediario de un objeto para controlar su acceso.

Patrón interface

- Define un comportamiento independiente de donde vaya a ser utilizado .
- Desacople entre comportamiento y clase.

Google Fit

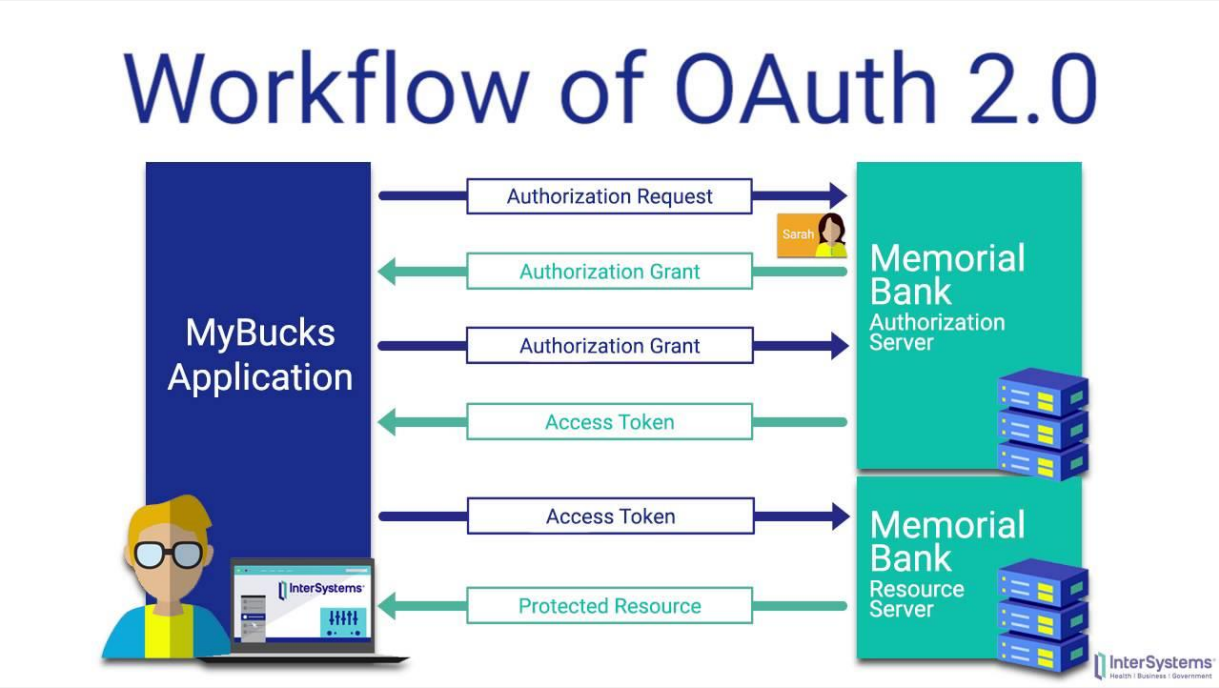


Google Fit

Utiliza Google Fit REST API para acceder a los datos almacenados en la nube de Google

Funcionamiento de Google Fit

1. Oauth2



2. REST API

- Lee o guarda información almacenada en la nube de Google.
- Necesita Access Token.

https://developers.google.com/fit/rest

Bless You

Frameworks y herramientas utilizadas

Diagrama de bases de datos

Diagrama entidad-relación de la base de datos.

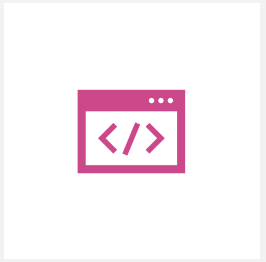
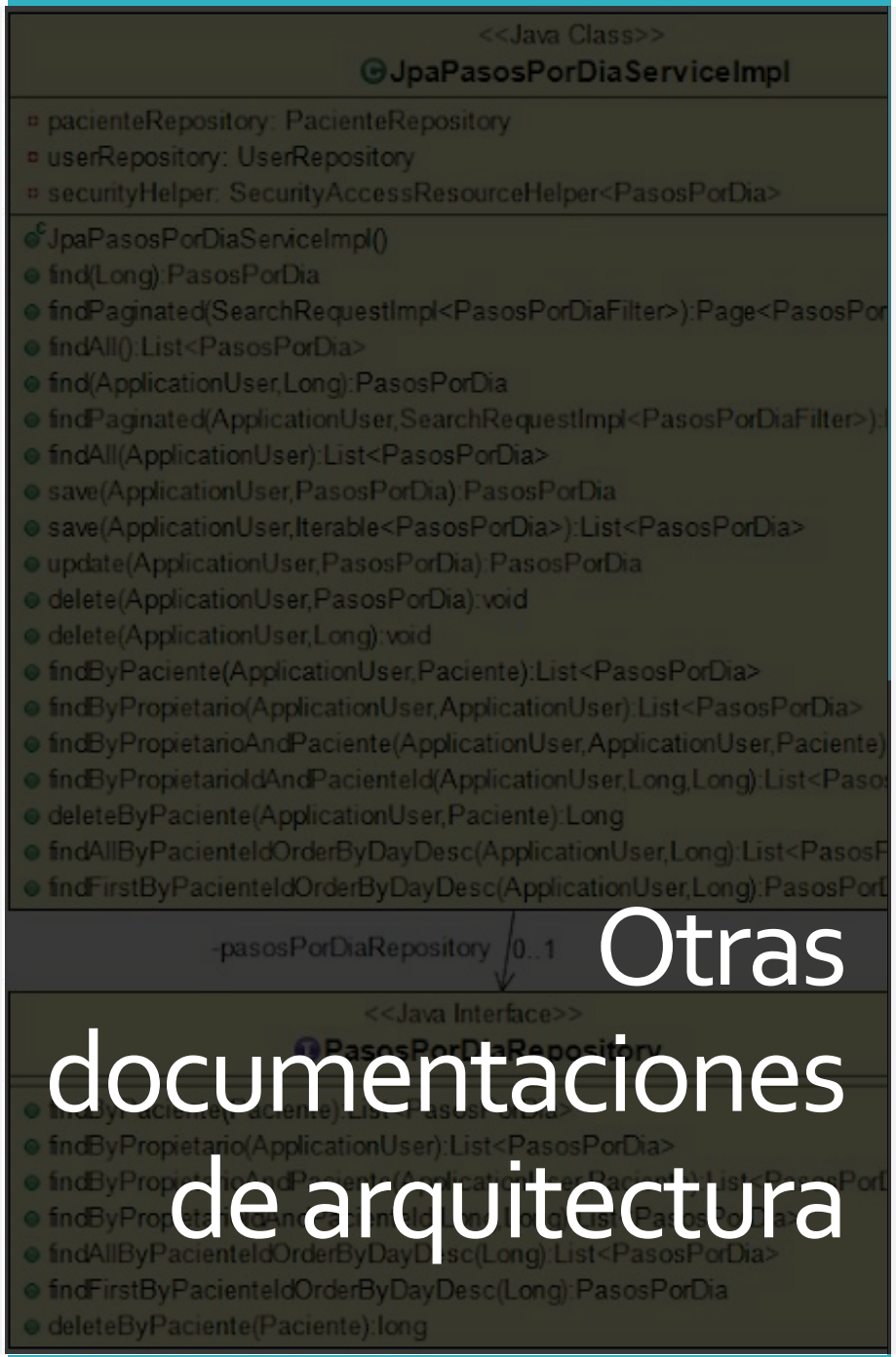


Diagrama de clases

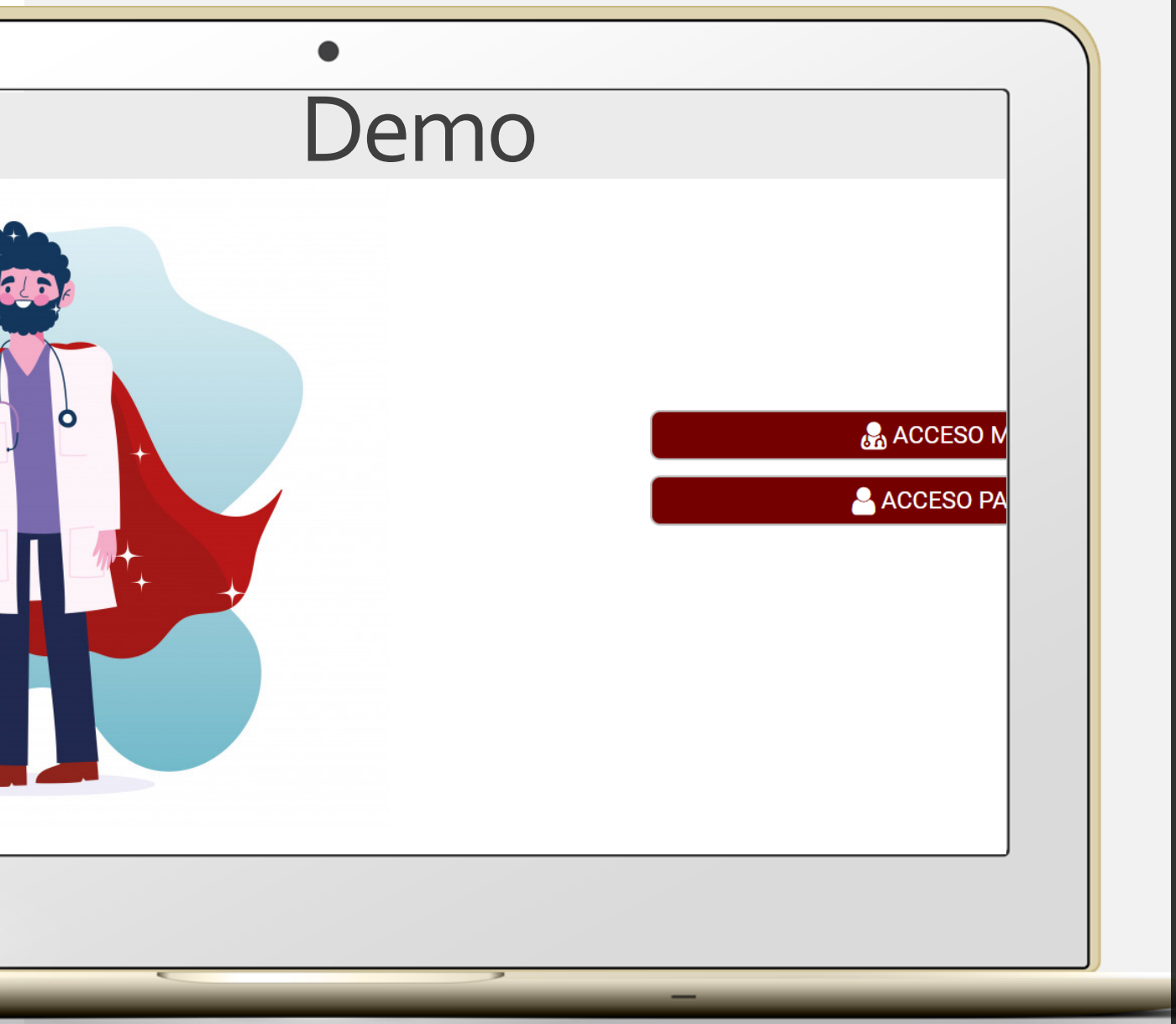
Diagramas de clases del código Java más importante.

Ejemplo de flujo de código

Ejemplo de flujo de código de la sincronización de pasos de Google Fit.



Otras documentaciones de arquitectura



Pruebas

- Pruebas manuales
  - Se realizan pruebas por cada requisito.
  - Además se comprueban distintos escenarios de los requisitos.
- Junit
  - Se preparan los proyectos con las librerías necesarias.

Conclusiones

Personales

- Covid ha influenciado en el desarrollo del proyecto.
- Mejora de la seguridad profesional.
- Motivación por finalizar.

Proyecto

- Objetivos generales alcanzados.
- A nivel más detallado, se podrían haber alcanzado mejores objetivos:
  - Diseño responsive.
  - Test unitarios

Metodología

- Adecuada.
- Visualización de trabajo realizado/pendiente.
- Alguna pequeña adaptación.

Mejoras

- Diseño responsive.
- Mejora accesibilidad.
- Test unitarios.
- Test integración.
- Mejora de la conectividad médico/paciente.
- Calendario citas médicas.
- Pago online.

Bless You

Gracias

Paloma Terán Pérez

palomateran@uoc.edu