HEALTHCARE SYSTEM

Lorenzo Rocchi – 294418
Beatrice Tomassi – 295351

TABLE OF CONTENTS

01.
INTRODUCTION

DOMAIN

The healthcare system operates within the medical services domain, focusing on **optimizing administrative processes** such as managing patient information, scheduling appointments, and maintaining medical histories.

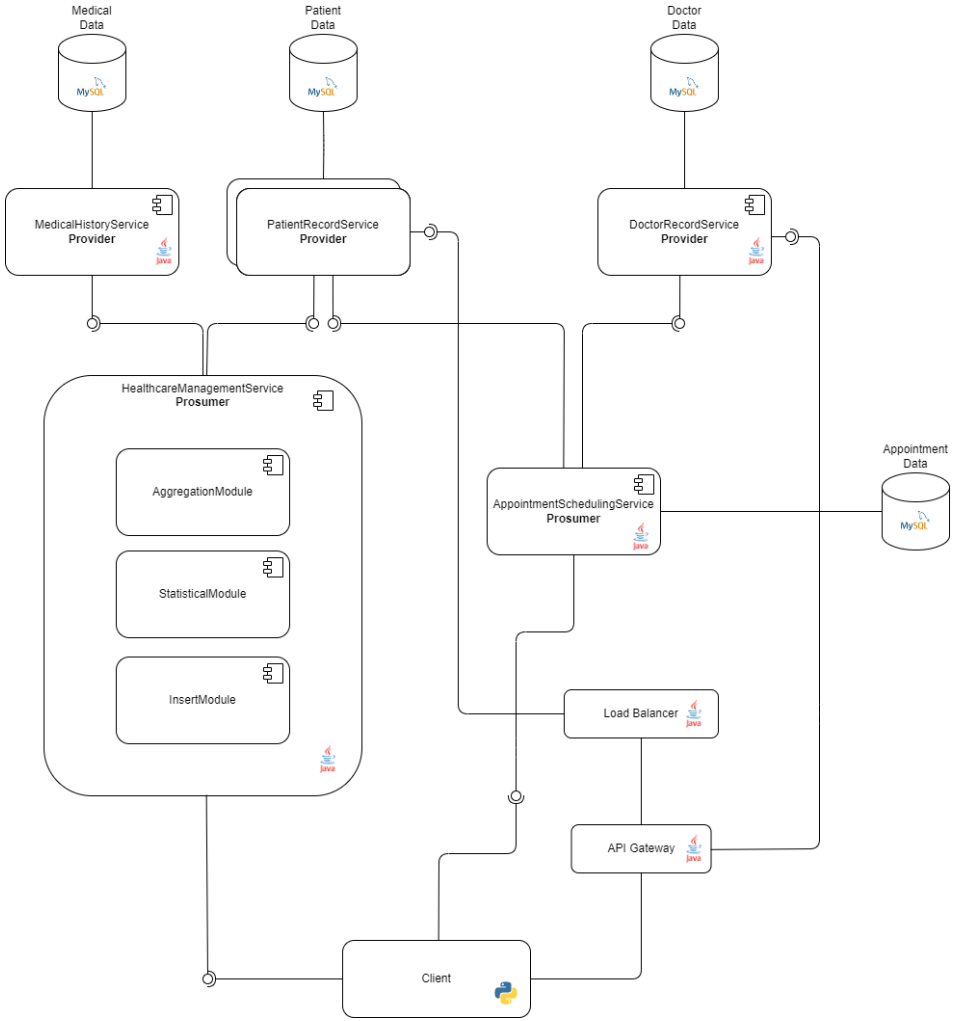
INTENDED USERS

Administrative Staff of a hospital or a clinic.

In particular, the system is focused on the needs of receptionists and healthcare providers who are “behind the desk”.

02.
ARCHITECTURE

COMPONENT DIAGRAM



OUR SERVICES

Service Name	Type	Arch. Style	Implementation
Patient Record Service (PRS)	Provider	REST microservice	Spring Boot
Doctor Record Service (DRS)	Provider	REST microservice	Apache CXF
Medical History Service (MHS)	Provider	REST	Spring Boot
Appointment Scheduling Service (ASS)	Prosumer	SOAP	Apache CXF
Healthcare Management Service (MHS)	Prosumer	REST	Spring Boot

PATIENT RECORD SERVICE

The PRS provider is designed to manage all patient-related data (name, surname, CF, age, address...).

ENDPOINTS

[GET, POST]
http://localhost:8080/prs/patientData

[GET, PUT, DELETE]
http://localhost:8080/prs/patientData/{cf}

DOCTOR RECORD SERVICE

The DRS provider is designed to manage all doctor-related data (name, surname, specialization).

ENDPOINTS

[GET, POST]
http://localhost:8082/drs/doctorData

[GET, PUT, DELETE]
http://localhost:8082/drs/doctorData/{id}

MEDICAL HISTORY SERVICE

The MHS provider is responsible for storing detailed medical histories of patients, including past diseases or treatments.

ENDPOINTS

[GET, POST]
http://localhost:8081/mhs/medicalRecord

[GET, PUT, DELETE]
http://localhost:8081/mhs/medicalRecord/{cf}

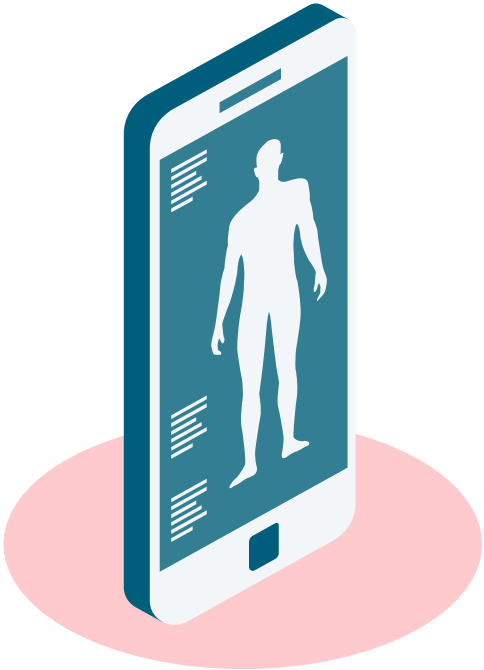
APPOINTMENT SCHEDULING SERVICE

The ASS prosumer is responsible for scheduling appointments between a patient and a doctor. It's possible to create, update, delete and retrieve appointments.

ENDPOINT

http://localhost:8083/appointmentSchedulingService

HEALTCARE MANAGEMENT SERVICE



The HMS prosumer contains three modules:

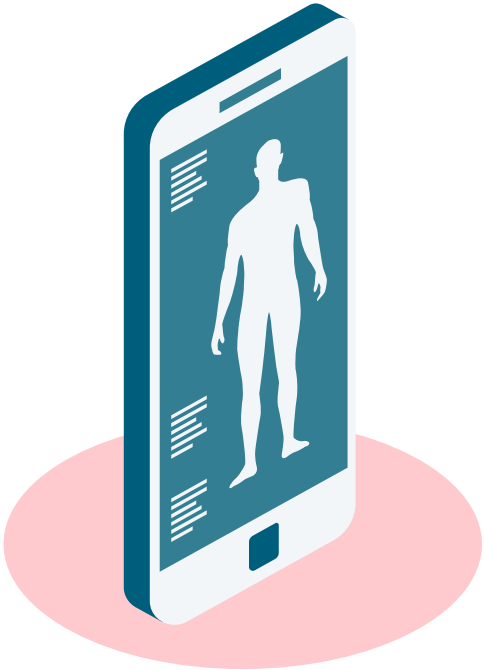
1. **InsertModule** to allow the client to communicate with the MHS provider.

ENDPOINTS

[POST] http://localhost:8084/hms/medicalRecord

[PUT] http://localhost:8084/hms/medicalRecord/{cf}

HEALTCARE MANAGEMENT SERVICE



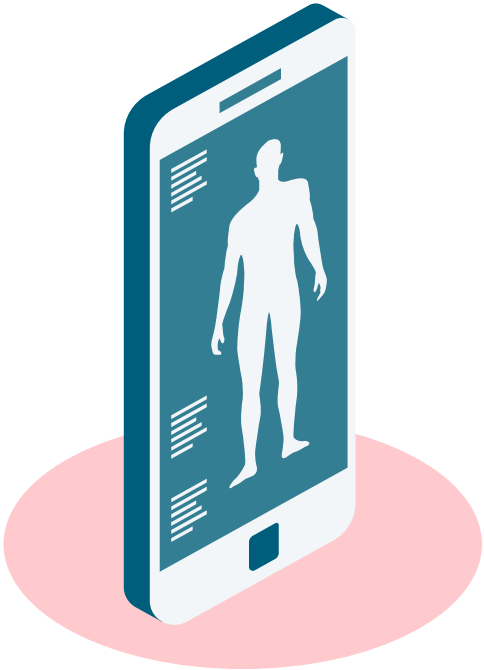
The HMS prosumer contains three modules:

2. **StatisticalModule** to perform some analysis on the data of all patients, such as which is their average age or which is the percentage of patients without diseases.

ENDPOINTS

- [GET] http://localhost:8084/hms/statistical/average-age
- [GET] http://localhost:8084/hms/statistical/total-patient
- [GET] http://localhost:8084/hms/statistical/gender-percentage
- [GET] http://localhost:8084/hms/statistical/nodisease-percentage

HEALTCARE MANAGEMENT SERVICE



The HMS prosumer contains three modules:

3. **AggregateModule** to aggregate all the data regarding a patient, including the administrative informations provided by PRS and the medical informations contained in MHS.

ENDPOINTS

[GET] http://localhost:8084/hms/aggregateData

GATEWAY

Directs incoming requests to the appropriate service provider (PRS or DRS).

ENDPOINTS

[GET, POST] http://localhost:9000/prs

[GET, PUT, DELETE] http://localhost:9000/prs/{cf}

[GET, POST] http://localhost:9000/drs

[GET, PUT, DELETE] http://localhost:9000/drs/{id}

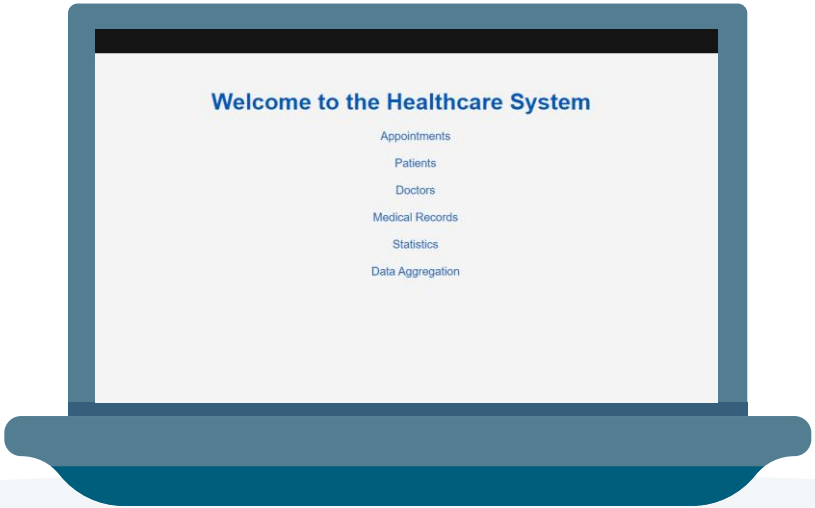
LOAD BALANCER

Ensures the efficient distribution of incoming requests across multiple instances of a microservice.

03.

HOW DOES IT WORK?

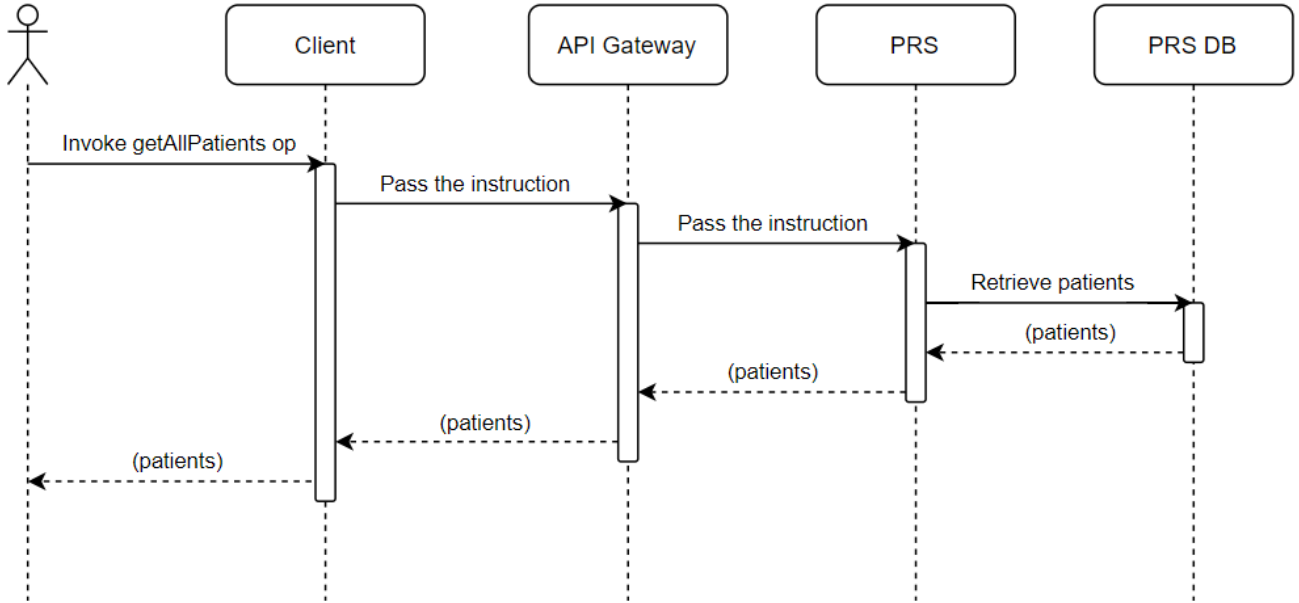
CLIENT



A client web application was built with python.

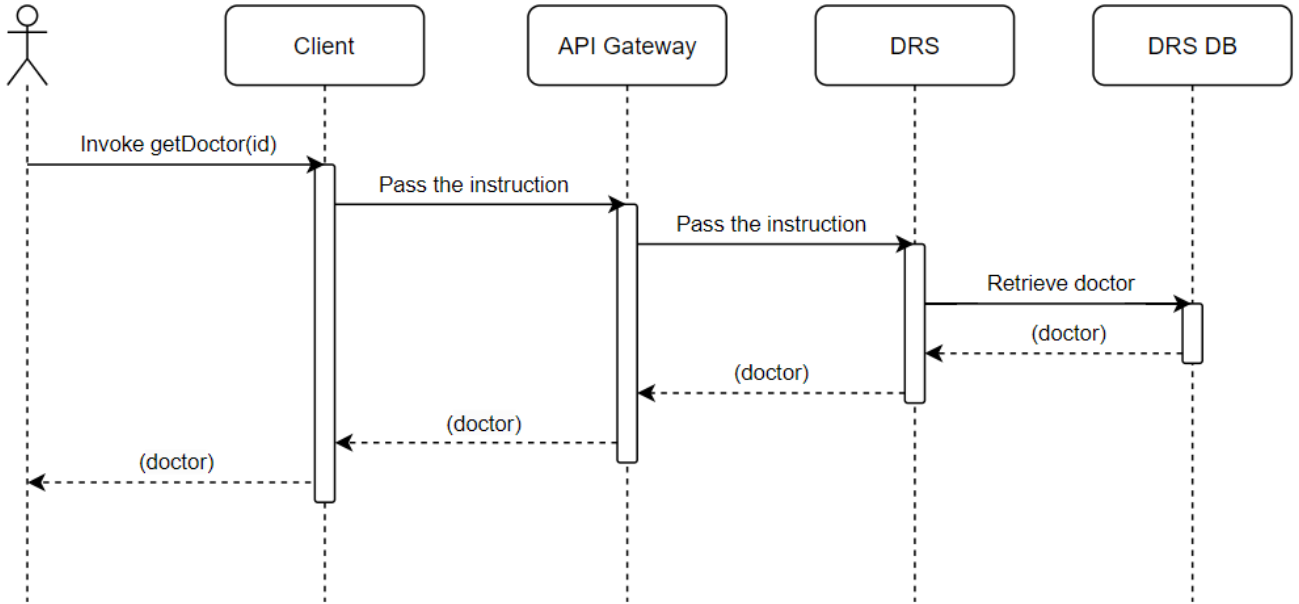
INTERACTION CLIENT - PRS

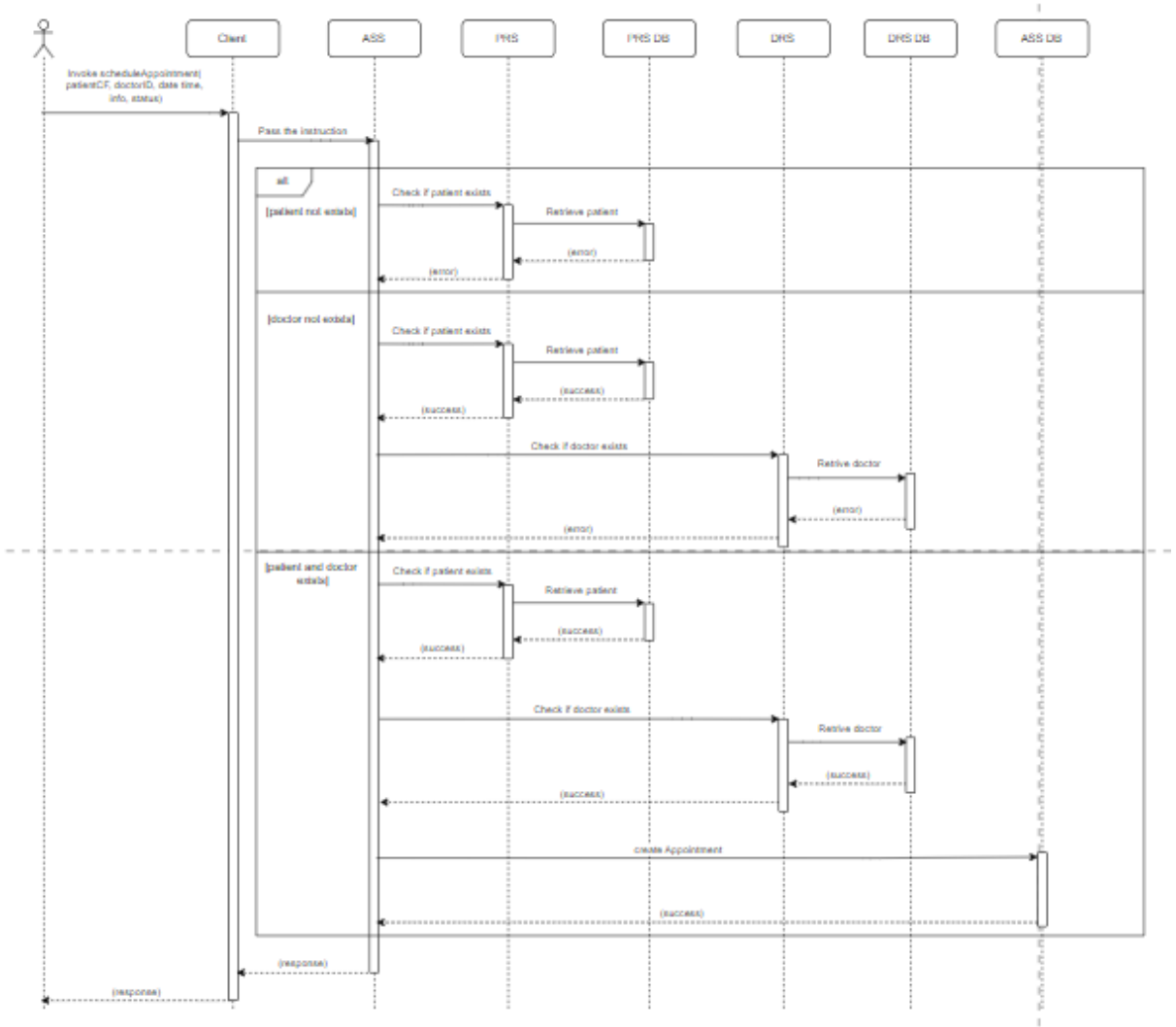
The receptionist wants to visualize all the patients registered in the system



INTERACTION CLIENT - DRS

The receptionist wants to visualize the details of a single doctor



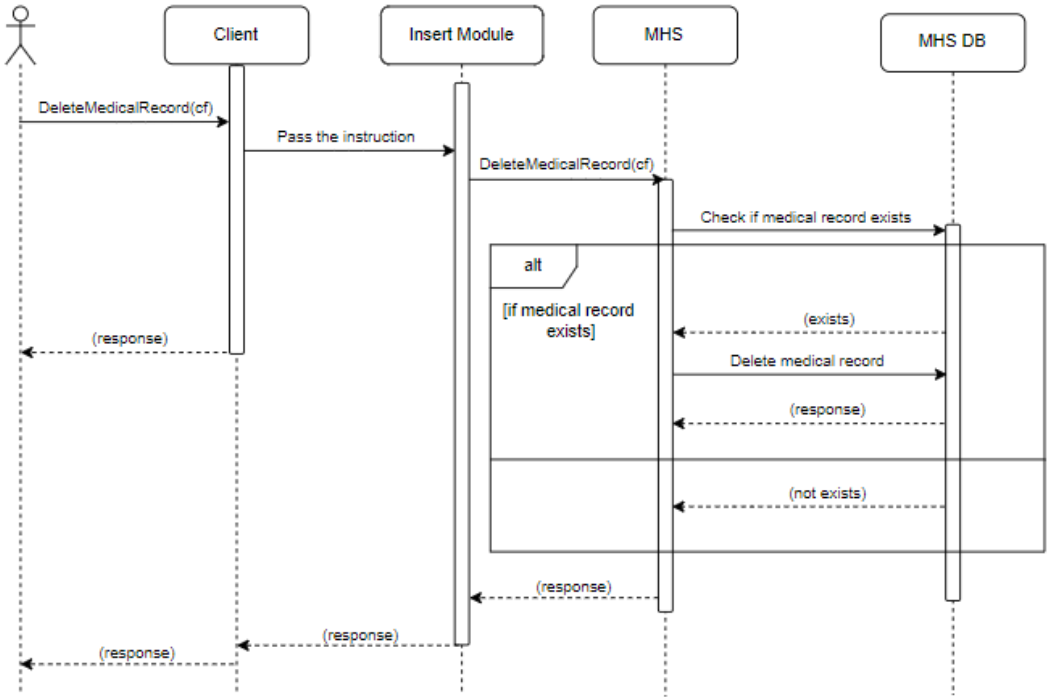


INTERACTION CLIENT - ASS

The receptionist wants to schedule a new appointment

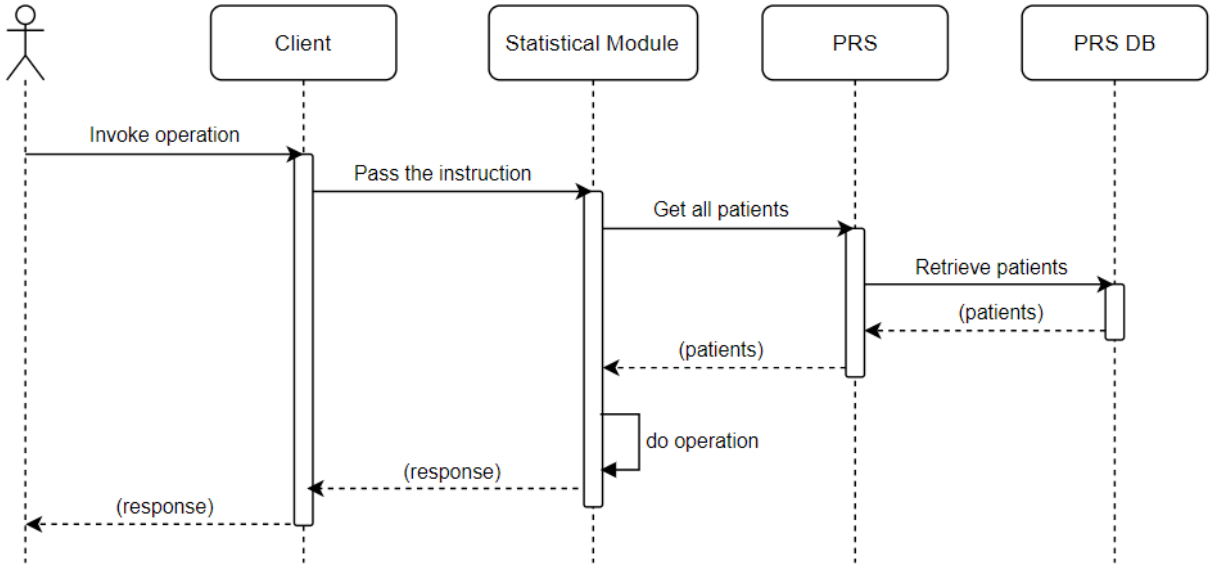
INTERACTION CLIENT – HMS (INSERT)

The receptionist wants to delete a medical record



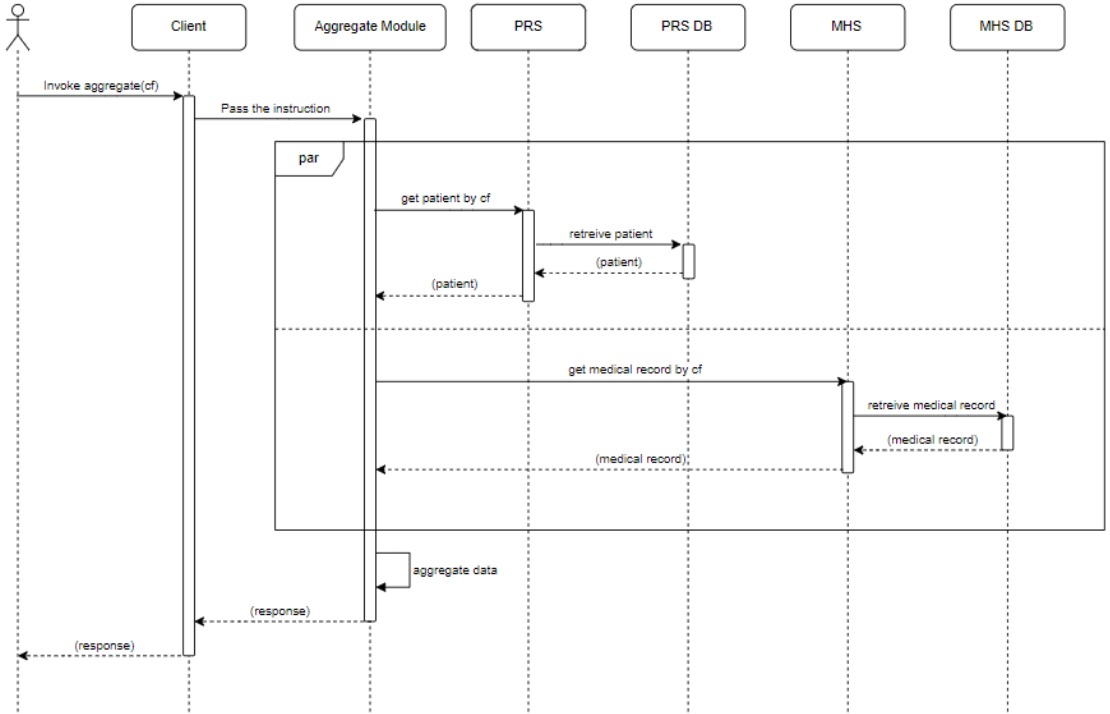
INTERACTION CLIENT – HMS (STATISTICAL)

The receptionist wants to visualize a statistic

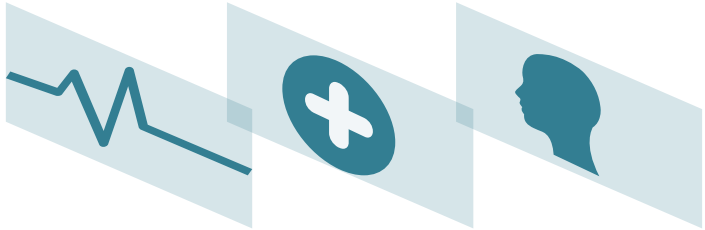


INTERACTION CLIENT – HMS (AGGREGATE)

The receptionist wants to aggregate all the informations regarding a patient



04.
PROJECT DEMO



https://github.com/itsrocchi/HealthCare-Sose

05.
CONCLUSION

MEETING THE SPECIFICATIONS

The system comprise REST and SOAP services and microservices developed with Apache CXF and Spring Boot.

It is composed of:

Providers

- PRS
- DRS
- MHS

Prosumers

- HMS
- ASS

Client

MEETING THE SPECIFICATIONS

- The interaction client-to-service pass through an API gateway.
- The client interacts with two prosumers (ASS and HMS) and two providers (PRS and DRS).
- One module (AggregateModule) is developed in an asynchronous way and allows two providers (MHS and PRS) to execute their job in parallel.
- Microservice (PRS) can be deployed in multiple instances and their interaction is load-balanced.
- The project is Mavenized and Dockerized.
- The architecture and interaction scenario are shown as component and sequence diagrams.
- All the REST services are equipped with OPEN API and Swagger documentation.

THANK YOU!