



End of Studies

Design & Development of a Fullstack Web-Based platform for resource management to handle online appointments and reservations within different facilities

Selim HORRI



TABLE OF CONTENTS

- 01 Introduction
 - 02 Context model
 - 03 Product Backlog
 - 04 Design
 - 05 Implementation Stack
-



Introduction



Most of service providers need to manage their resources and reservations within time slots in order to fulfill their customer needs.

In this context, we are planning to develop a resource management platform that can be used in various fields, including medical offices, cosmetic saloons etc...

This platform will be accessible via a Web application. (*for now*)



Context model

Actors



Customer [*Visitor*]

...



Worker

...



Manager

...



Owner

...



Dive into

User Authorities & Usecases

Use cases

Visitor service

Visitor is authorized to consult :

01 Branches

02 Facilities/Saloons



Use cases

Customer service

After being authenticated, a customer should be authorized to manage :

01 Favourites

02 Reservations

03 Ratings



Use cases

Owner service

After being authenticated, an owner should be authorized to manage :

01 Facilities



Use cases

Manager service

After being authenticated, a manager should be authorized to manage :

- 01 Categories
- 02 Service details
- 03 Workers
- 04 Appointments



Use cases

Worker service

After being authenticated, a worker should be authorized to manage :

01 Timetable

02 Tasks/Todos



Product Backlog

Backlog & Releases

...

01

App Configurations

Sprint 1:

- Setup Backend & Frontend apps

Sprint 2:

- Infrastructure as a Service (IaaS)
- Configs (env vars, secrets...)
- CI / CD pipeline conf
- Deploy Poc(s) as Images

02

Auth Security process

Sprint 1:

- Authentication & Authorization

Sprint 2:

- Registration

03

Customer service

Sprint 1:

- Consult Facilities & Locations
- Consult Topics & tags

Sprint 2:

- Manage Favourites & Reservations

Sprint 3:

- Rate a worker

04

Worker service

Sprint 1:

- Checkout timetable
- Manage Todos

05

Manager service

Sprint 1:

- Manage facility categories & services

Sprint 2:

- Manage Workers & Reservations

06

Owner service

Sprint 1:

- Manage facilities

Design

Use case :

https://github.com/SelimHorri/atic-backend-app/blob/master/docs/doc/cita_usecase_diag.drawio.pdf

Entity Relationship (ERD) :

https://github.com/SelimHorri/atic-backend-app/blob/master/docs/doc/cita_erd_diag.drawio.pdf

App Architecture :

https://github.com/SelimHorri/atic-backend-app/blob/master/docs/doc/cita_component_complexity.drawio.pdf

Implementation Stack

Datastores



Backend develop



Frontend develop



Code Quality Measurement



Version Control



Azure DevOps



git

CI/CD pipelines

Continuous Integration – Continuous Delivery



GitHub Actions



Azure Pipelines



Jenkins

Container Runtime



kubernetes



docker®

Cloud Provider

...



Monitoring

...





THANK YOU

...



+216 22 12 51 44



github.com/SelimHorri



hub.docker.com/u/selimhorri



dev.azure.com/SelimHorri



selim.horri@ooredoo.tn

