universidade de aveiro

Event Management System

M4: Final Presentation and Results

Projeto em Informática 2023/24

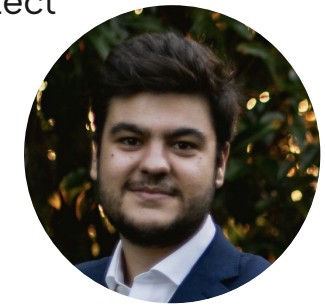
Roberto Castro - 107133
Rafaela Abrunhosa - 107658
Marta Inácio - 107826
Tiago Gomes - 108307
Sara Almeida - 108796

Our team

Marta Inácio
Software Architect

Sara Almeida
Team Manager

Rafaela Abrunhosa
DevOps Master



Tiago Gomes
Product Owner

Roberto Castro
DBA / DevOps

Guiding Teacher

Joaquim Sousa Pinto

Table of contents

01 Context and State of The Art

02 Problem

03 Actors and Requirements

04 Data Model and Architecture

05 Implementations

06 Testing and Results

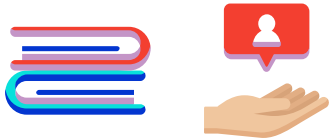
07 Deployment and Future Work

08 Demo

01

Context

The event management industry faces constant challenges.



Develop an integrated management system for the events: Participants info, medical info, entry and exit system, etc ...

Able to simultaneously deal with large audiences in different activities of the same event.

01

State of The Art

	360 CITY	eventify	cvent	FLEXIGATHER
Access Management	✓	✓	✓	✓
Balance Management	✓	✗	✗	✓
Payment Methods (QR Code)	✓	✗	✗	✓
Roles Management	✓	✓	✓	✓
Purchase Products via QR Codes	✓	✗	✓	✓
Account Loading for Cashless Transactions	✓	✗	✗	✓
Medical Atention Management	✗	✗	✗	✓

02

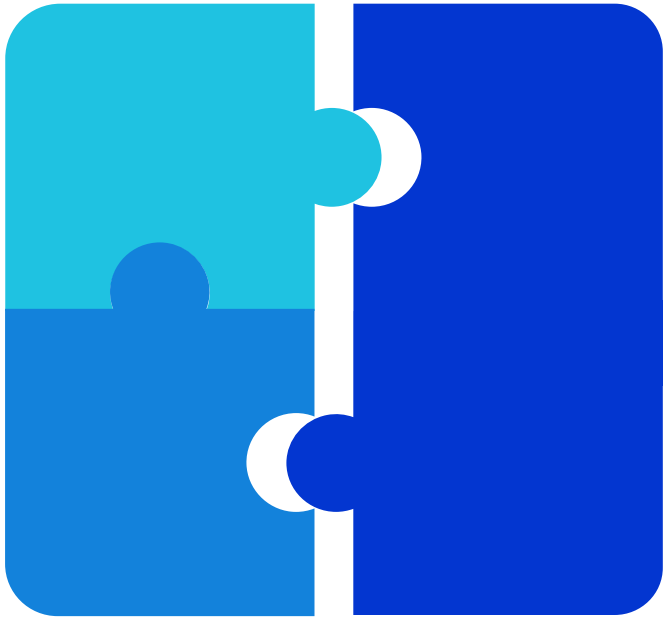
Our domain based system

Medical Center

With pre-filled medical files to better attend **emergency** situations.

Events

Access and Kiosks logistics common to all this kind of events.



Theme

The theme of the event can be changed to **meet** the client **expectations** (i.e Scouts Event)

03 Actors and Requirements

Participants

- Access personal data
- Enter and leave the premises
- Add money to theis accounts

Medical Staff

- Access medical information
- Create treatment files
- Access treatment files history

Administration

- Allow registration of participants
- Update product stock
- See the event participants flow

Sales Staff

- Registration of sales
- Allow money to be load into participants accounts

Entrance Staff

- Record entries of the participants
- Record exits of the participants

03 Non-functional Requirements

Functionality	Performance	Security	Availability
The system should be able to read QRCodes/NFCCodes.	The system should handle large volumes of audiences (data capacity).	Only medical and administration staff can have access to the participants health information.	All the interfaces (except the kiosk) should be available 24/7, this including during the night.
The system should allow the use of more than one browser.	The system should warn the responsibles chiefs in less than 3 seconds whenever a child on their charge entries in the medical center (response time).	Only sales staff can make a sale or exchange.	If the system crashes, the data should not be lost.

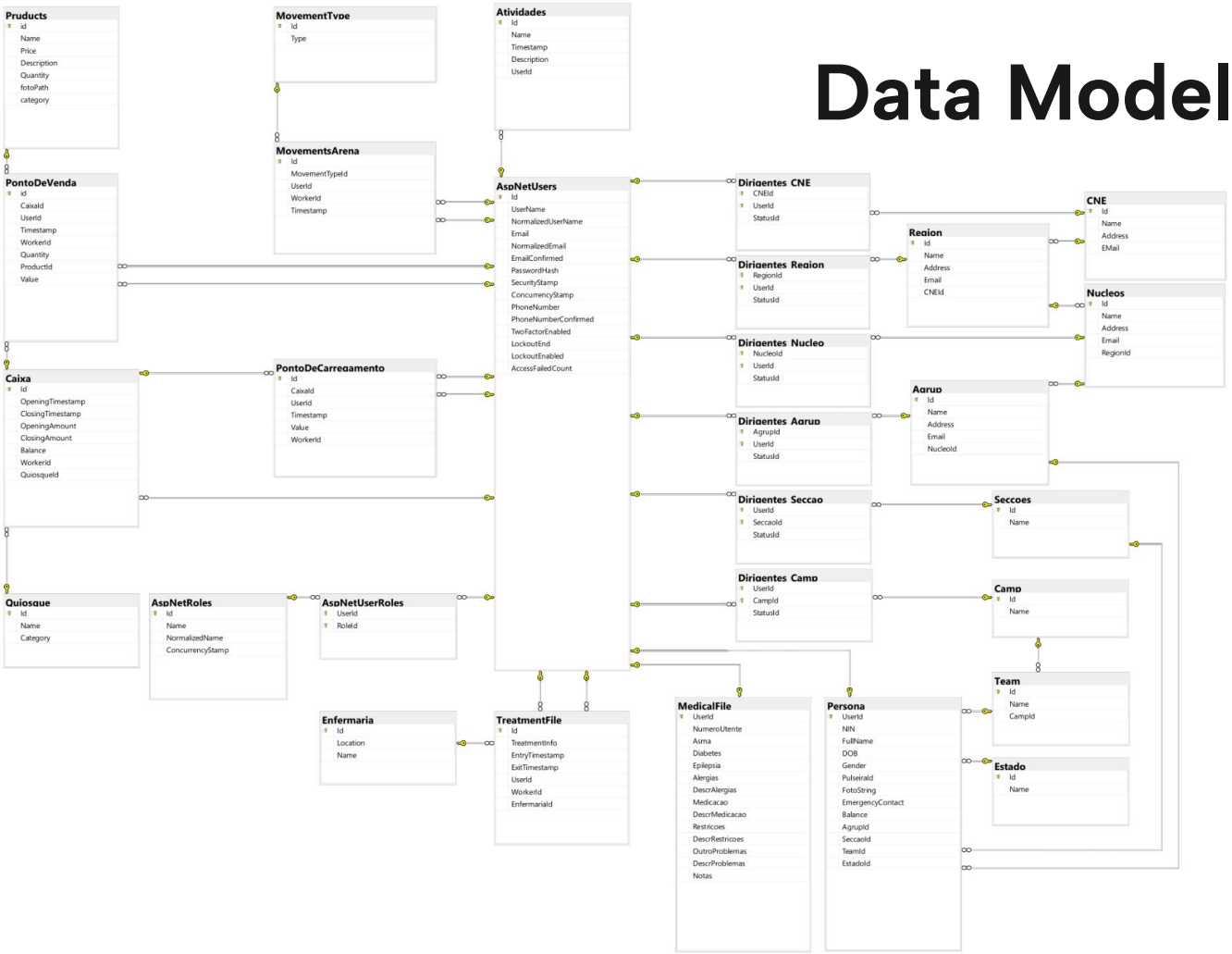
03

User’s Identification

- NFC bracelets
- QRCodes

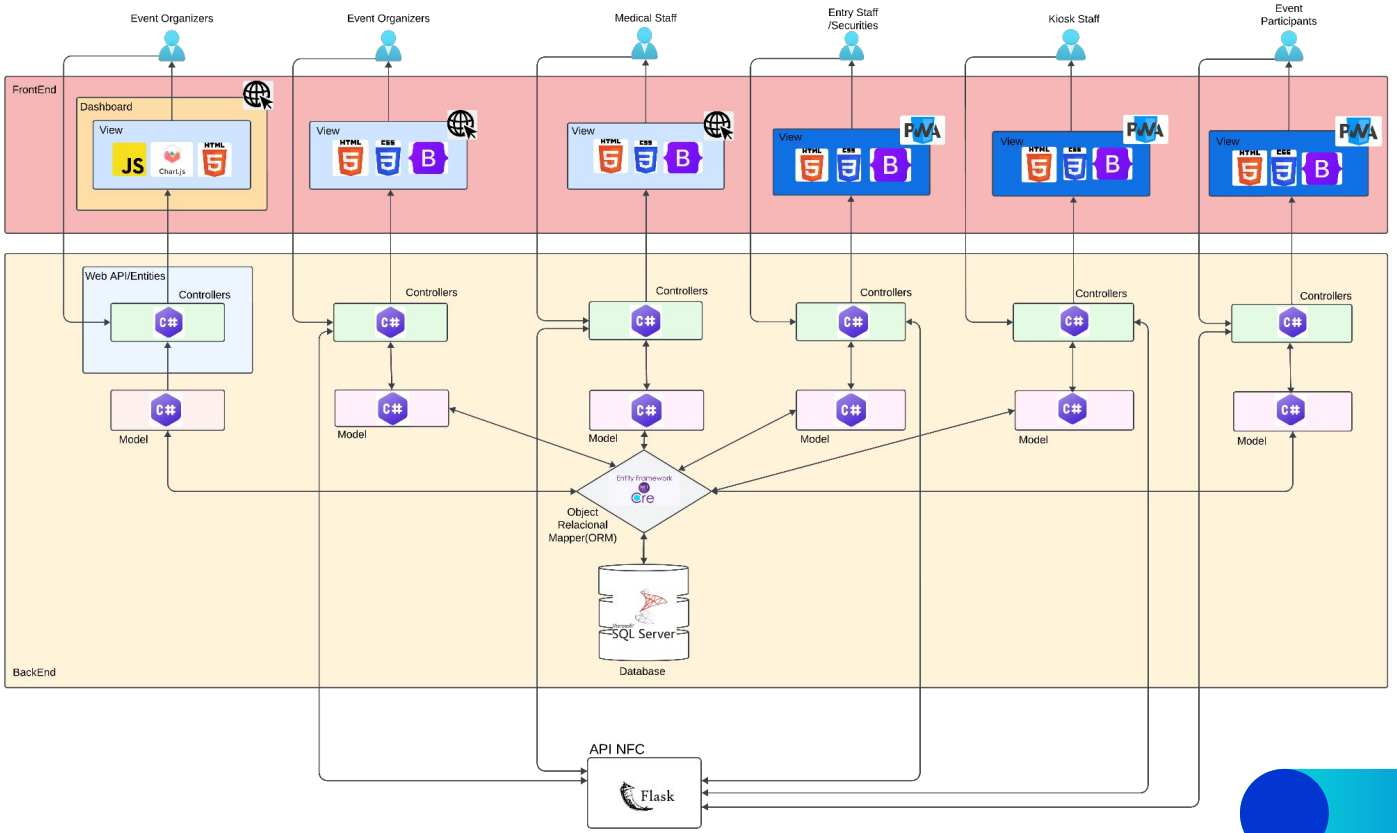
04

Data Model



04

System Architecture



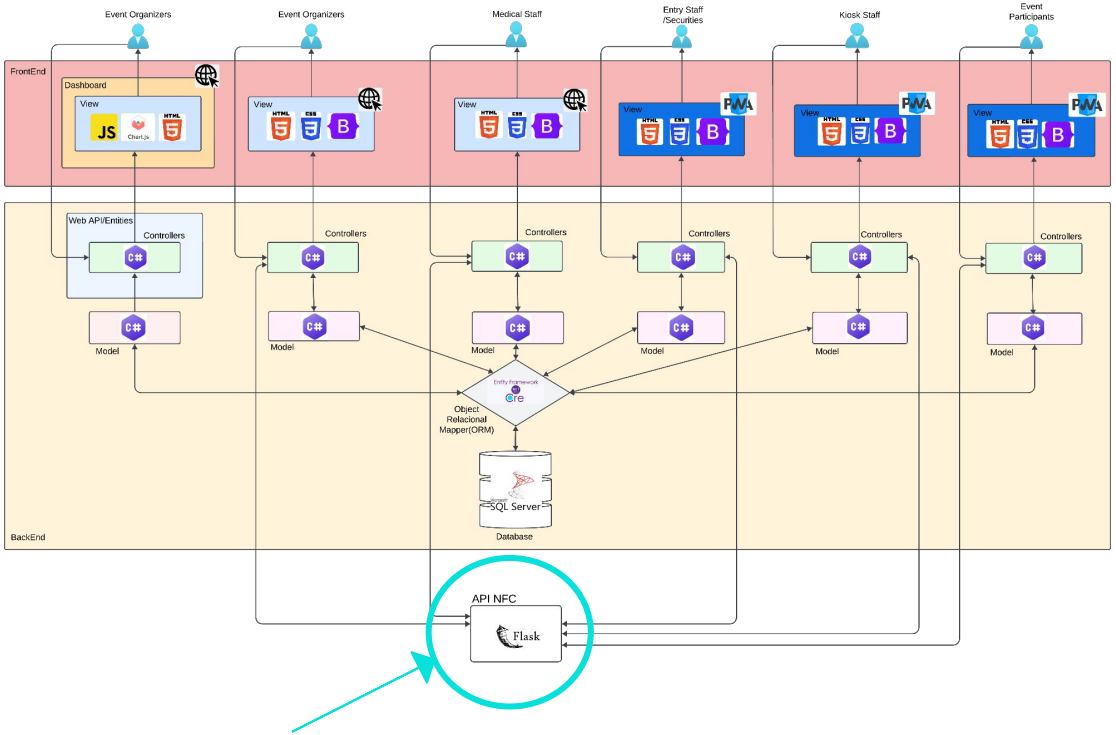
Implementations

05

NFC Integration

API development

GET: /api/nfc/read



Implementations

05

Participants photo identification

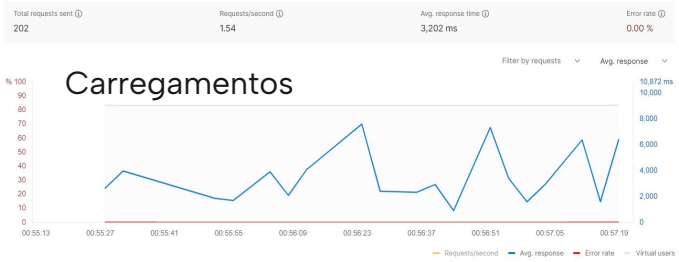
The images identifying the participants will be stored in the wwwroot folder.

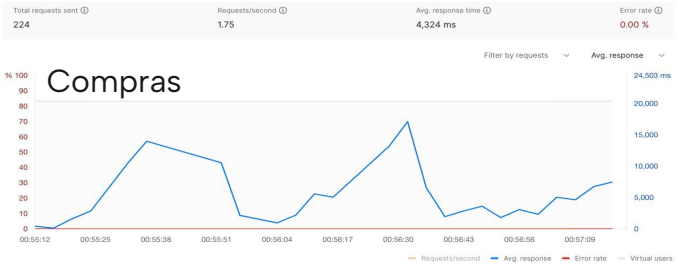
Notifications

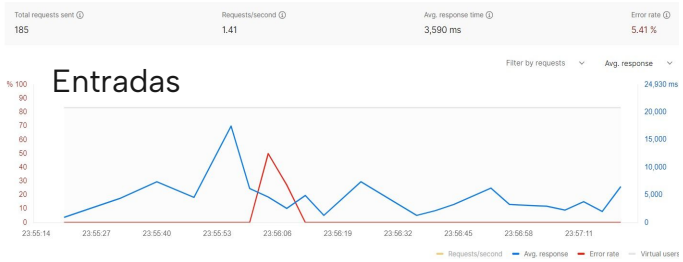
The notifications are meant to alert the **responsible** of the children for their entry and movement in the medical center.

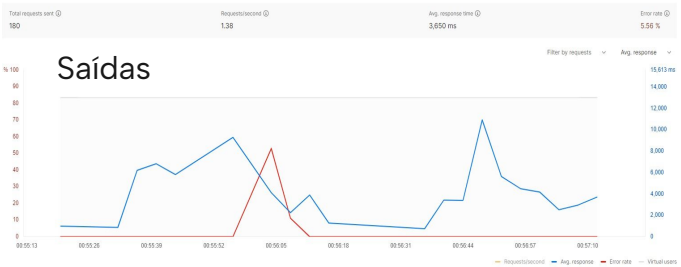
06

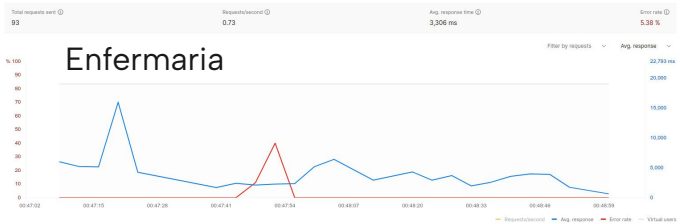
Testing and Results

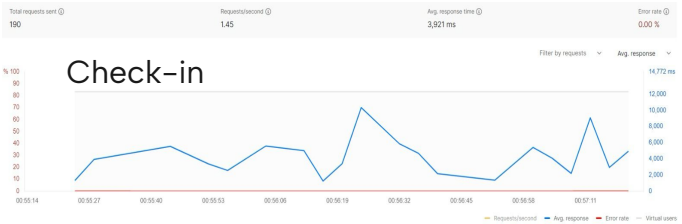






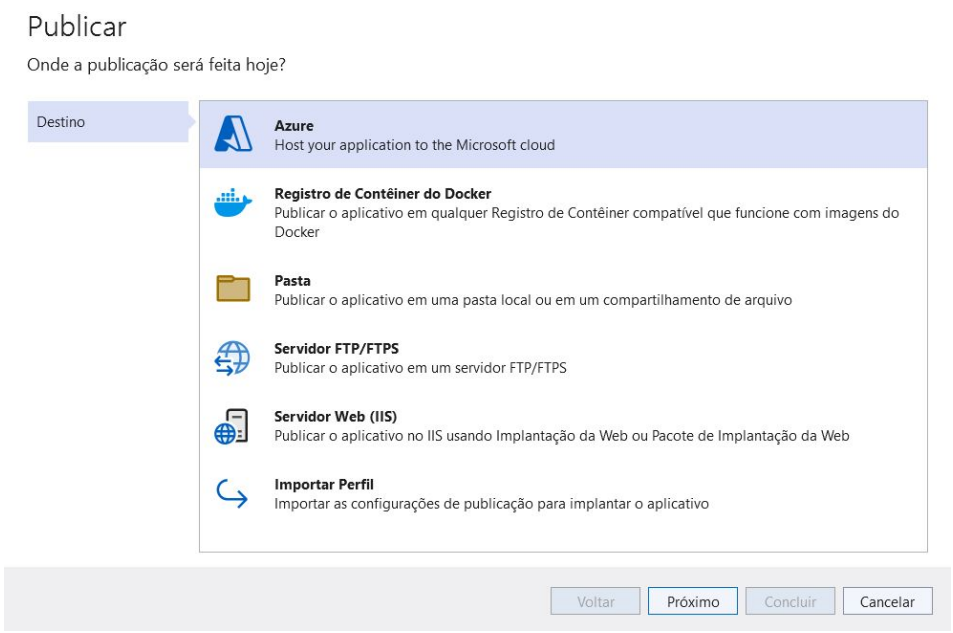






07

Deployment




```
{
  "ConnectionStrings": {
    "DefaultConnection": "Server=192.168.160.27;Database=FlexiGather;User Id=FlexiGather;Passwor
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*"
}
```

AppSettings Release

```
{
  "DetailedErrors": true,
  "ConnectionStrings": {
    "DefaultConnection": "Server=localhost;IntegratedSecurity=true;MultipleActiveResultSets=true;Trust
  },

  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  }
}
```

AppSettings Development

07

Future Work

Asynchronous messaging

Like Kafka to help with traffic and scalability.

Notifications

Implement proper system notification.

Payment methods

Add the possibility of charging bracelets without physical money.

08

Demo

Thanks!