OnYourWay to 24Meds: Product specification report

Paulo Pereira [98430], Mariana Rosa [98390], Artur Romão [98470], Diogo Cruz [98595]

v2022-05-13

1 - Introduction

1.1 - Overview of the project

The objective of this assignment is to implement a digital marketplace project split into two sub-projects:

- A deliveries platform, that handles the system's logic behind orders and riders matchmaking and that contains a performance dashboard with the most relevant statistics (such as new orders, clients reviews, user registrations, riders reviews, etc): OnYourWay
- A market application, that should integrate with the deliveries platform, could be any kind of marketplace, in our specific case, it is a parapharmacy implemented with an end-user mobile app: 24Meds

Bearing these requirements in mind, we came up with the idea of both OnYourWay, the deliveries engine, and 24Meds, an online parapharmacy market store.

24Meds' goal is to hand over non-prescription medicines and self-care products at home, 24 hours per day, 7 days a week, prioritizing the client's accommodation. For that purpose, they hired OnYourWay to handle and manage their products' distribution.

1.2 - Limitations

This project had quite some limitations, we did not achieve the results we wanted, although all the effort we put into it. We couldn't make the apps communicate with each other, nor implement the user stories where a purchase was made and monitoring statistics of the app.

At first, our domain model was confusing and we think we made it way more confusing than it should be. Therefore, during the implementation we had some issues putting it all together.

2 - Product concept

2.1 - Vision statement

Our application will focus on providing support and comfort to users wanting to get their needed medication without having to move outside their homes. It will be specially designed for people that can't leave their house due to their illness these users will be able to count on us 24/7 instead of having to resort to family and friends that might not always be available for their needs. This is a fairly unused aspect of courier services and we feel like our application will fit this gap in the market due to its high reliability and readiness.

At the start, we had plans to make the application also able to read doctor's prescriptions on delivery in order to reach even more users.(caso não dê preencher este paragrafo).

Systems like Glovo also provide this kind of service but they only allow for scheduled deliveries when it's late at night, we instead intend to have a more user-focused approach of always being available by counting on late-night riders to deliver the needed medication to users.

To develop the initial requirements and to develop the initial concept we recurred to close relatives and friends in the medical and pharmaceutical industries to provide us with a bigger depth of knowledge.

2.2 - Personas

Client:

Steven Stifler is a married 33 years old mechanic, born and raised in Aveiro. He currently works in his father's service shop located near Aveiro Forum. When he was a kid he wanted to be a chef but helping his dad during his teenage years made him grow a strong love for the family business. The more time passes, the more his workload grows, as his father gets older. Because of his age, he needs constant attention and care, mostly because his immune system is getting weaker, making him vulnerable to several problems and diseases. Stiffler needs to manage work and his dad's health.

Motivation: Because of his workload and taking care of his father he doesn't have much time to go to pharmacies or stores to get his dad's medication. He would like a service that could deliver him the products he needs in a quick and easy way.

Rider:

Joan Weah is a 25 years old student, currently taking her master's degree in Accounting. She was born in Viana do Castelo and currently studies in Porto. She is a very active person and likes to make plans and trips with her friends. She would like to save up some money for the summer so she can travel and party.

Motivation: To save up some money she would like to start a job as a delivery rider at OnYourWay.

Admin:

Megan Price is a 54 years old businesswoman, born in Vigo, Spain. However, her parents traveled a lot for work and ended up moving to Porto, Portugal. She grew up there and studied Economics in Lisbon. She was always interested in start-up businesses and she started her deliveries engine service in 2014 called **OnYourWay**, which aims to provide pharmaceuticals pickup and delivery services for clients. She sees her business as a success but wishes to continue growing it and reach more stores.

Motivation: She wishes to see her company reach more stores and cities, while always being able to see how the system is doing, see the performance of the riders, manage the stores associated and have a general overview of the system's global statistics so she can focus on the important task of growing her company.

2.3 - Main scenarios

Scenario 1: Client

His father has some headache issues and Steven doesn't have time to pick up the meds at a pharmacy. Therefore, he opens the 24Meds app and orders a box of paracetamol to his address.

Scenario 2: Rider

Joan is free at the moment so she opens the app and sets herself as available to work, so she can start delivering purchases.

Scenario 3: Admin

Megan Price is unsatisfied with **OnYourWay** growth and revenues lately so, to figure out what is actually happening, she consults the business statistics in her Admin Dashboard.

2.4 - Project epics and priorities

Since in Agile software development the project requirements and desires are usually expressed through epics, we followed a Mission-Based Prioritization which consists in ranking our epics' priorities from 1 to 4, in the following table.

Sprint 1 (19/05 - 26/5)

Task	Priority
Create Prototypes Balsamiq Web App 24 Meds	2
Create Prototypes Balsamiq App 24 Meds	2
Create Prototypes Balsamiq Web App On Your Way	2
Create Prototypes Balsamiq App On Your Way	2
Define Domain Model	4
Define Architecture	4
Build Prototype Web App 24 Meds	3
Build Prototype Web App On Your Way	3
Build Prototype App 24 Meds	3
Build Prototype App 24 On Your Way	3
Backlog management system set up in Jira	4

Sprint 2 (26/05 - 02/06)

Epic	Priority
Mobile application has functional GPS	1
A rider can register and login in deliveries engine	1
A market app client can see the list of products	3
A client can make an order	4
Rider has a browsable mockup app	4

Sprint 3 (03/06 - 09/06)

Epic	Priority
A rider can read a client’s QR code	2
A client can access the QR code connected to his order	2
A market app client can add products to the cart	4
A client can make an order	4

Sprint 4 (10/06 - 17/06)

Epic	Priority
View statistic about the app	2
Choose which rider is going to delivery the order	3
Connecting both systems	4

3 - Domain model

For this project two domain models were elaborated and supported with a MySQL database. One for **24Meds** (Fig. 1) and another for **OnYourWay**(Fig. 2).

On **24Meds** we have a Coordinates class that saves the location of the Pharmacy and of the Client. A Pharmacy has products that form a Cart. A Client makes a purchase and when it's accepted it changes to Delivery.

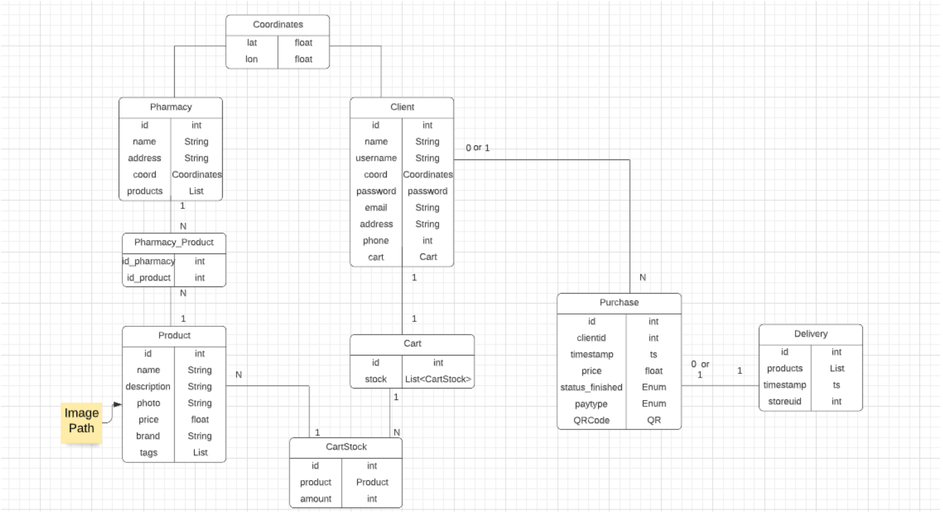


Fig. 1 - 24Meds Domain Model

On **OnYourWay** there is a class *Admin*, who runs the Deliveries Platform. On the deliveries Platform there are *Stores* signed on it. A *Ride* is composed of a *Package* and a *Rider.* There is also a *Coordinates* class to keep track of the location.

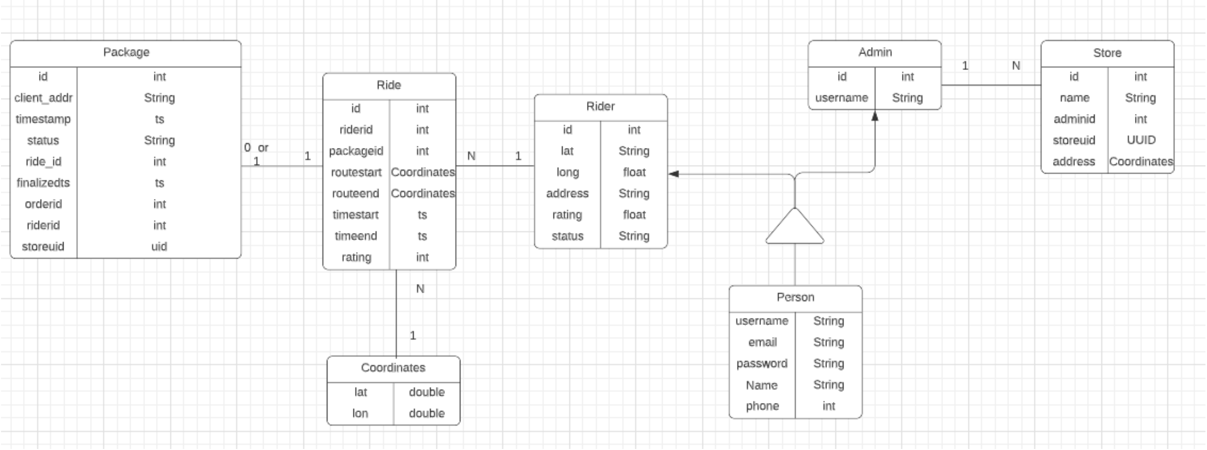


Fig. 2 - OnYourWay Domain Model

4 - Architecture notebook

4.1 - Key requirements and constraints

Our architecture was designed with the following requirements in mind:

- The delivery system should be independent and able to host multiple stores with different products while being able to receive concurrent requests from these stores.
- The system will need a managing platform used to control riders and see system reports and statistics.
- The system should be deployed in a virtual machine.
- The delivery system should notify riders of their rides' details.
- The delivery system should have an algorithm that designates a rider to delivery and if the rider refuses it should notify another one.
- The system should have an authentication system.

With this in mind, our system will have two main modules that interact via an abstract API.

The first module which is called OnYourWay is a delivery system that upon receiving a request for delivery will designate a rider to it. This module will consist of a logical backend with two integrated modules, an algorithm module that will decide the rider of the delivery, and an API capable of receiving packages from any store that uses the OnYourWay system. This first module also has a web app designed for managers that will be able to control different statistics about the rides. Alongside this, the system will provide a mobile app for riders where they can access and be notified of their deliveries.

The second module will be 24Meds which is a service that centralizes pharmaceutical products and allows its users to make orders. It will have a logical backend as well as an API that will help the frontend app display the products. Besides this, it will have a web app and a mobile app where users can make their orders.

Both modules will have their designated database. In the 24Meds' module, we will have a database where we store the users and the products as well as the pharmacies that provide them and the deliveries. In the second module, we have a database that will store the riders and the deliveries that were requested and made.

4.2 - Architectural view

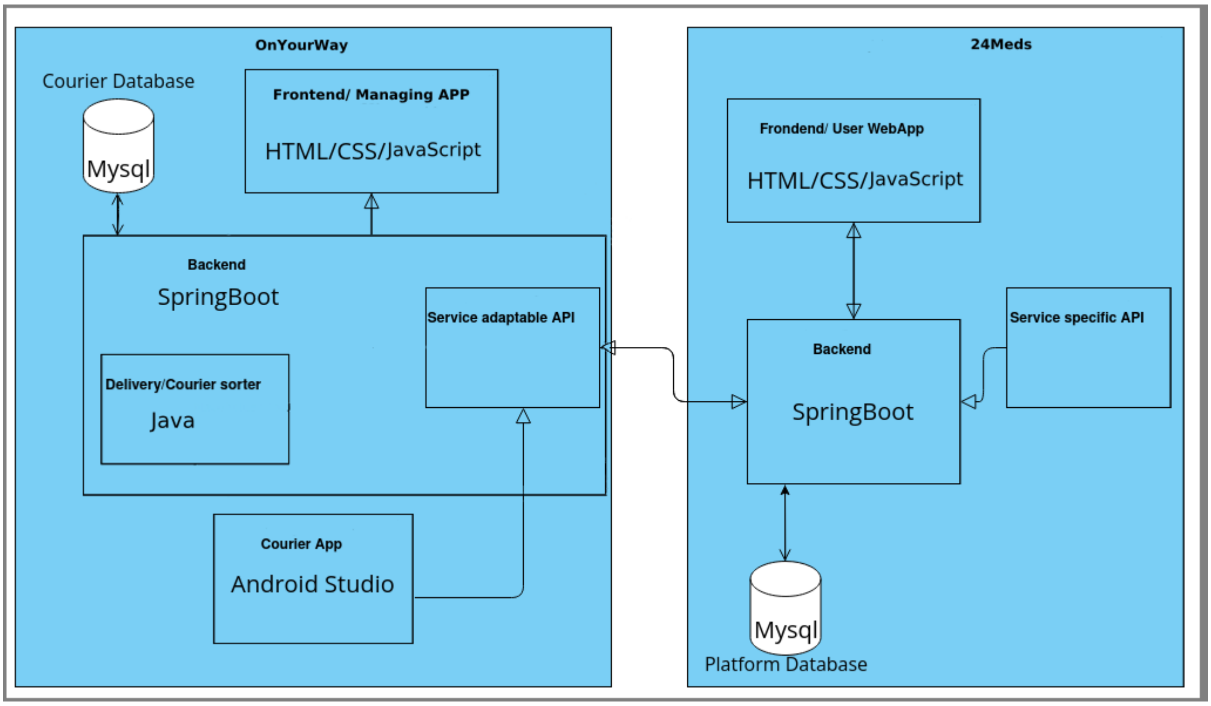


Fig. 3 - Architectural view

The idea behind our architecture is that there are two main modules that work independently, what we mean by that is, that our delivery platform is able to work with any specific platform and is only connected to the current service by an API that abstracts it's contents to the point that any service works with it.

With this said our deliveries platform must have a database that stores our riders and an algorithm module in python that will sort these riders. We also need an API that is able to receive packages from any kind of service and

4.3 - Deployment architecture

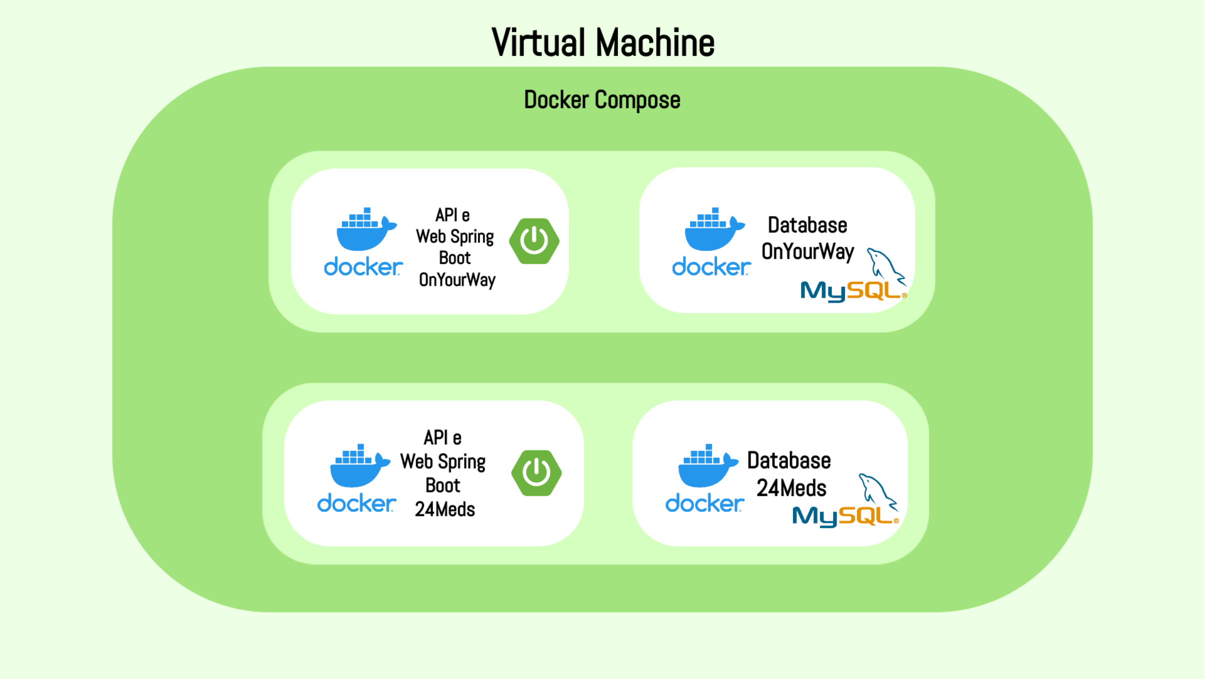


Fig. 4 - Deployment Architecture

For our deployment for each application the API and Web SpringBoot Application were containerized with Docker and configured with Docker Compose. In total, we have four containers, each one with ports and its own dependencies. The docker is running on a Virtual Machine and can be accessed on these links:

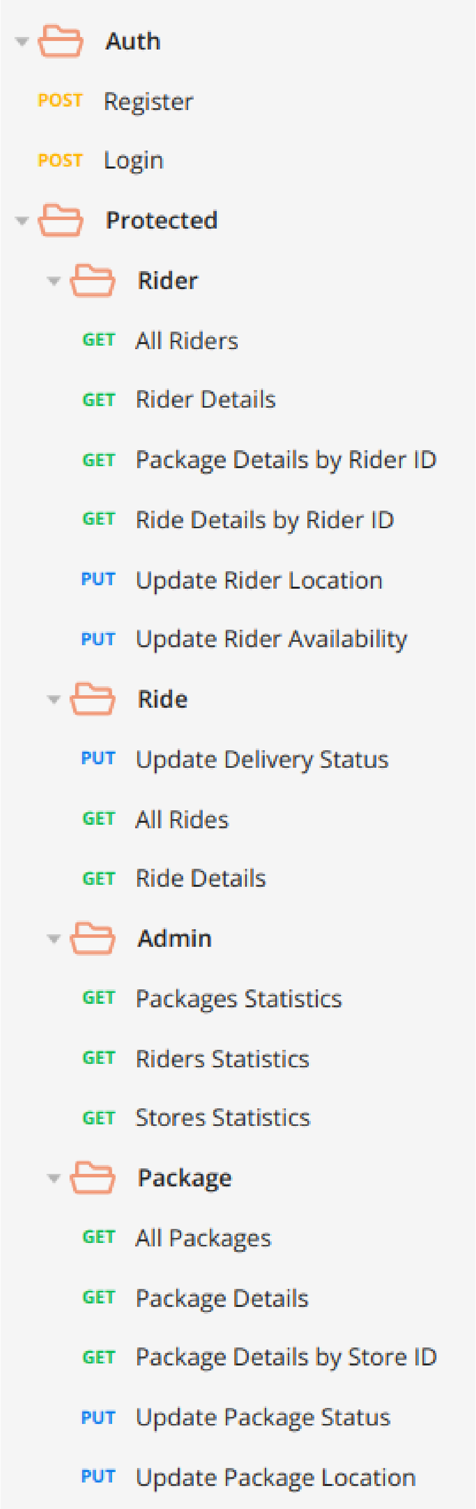
- http://deti-tqs-17:6767/ - OnYourWay
- http://deti-tqs-17:6768/ - 24Meds

5 - API for developers

For the APIs we documented each one on Postman.

OnYourWap API

Postman documentation

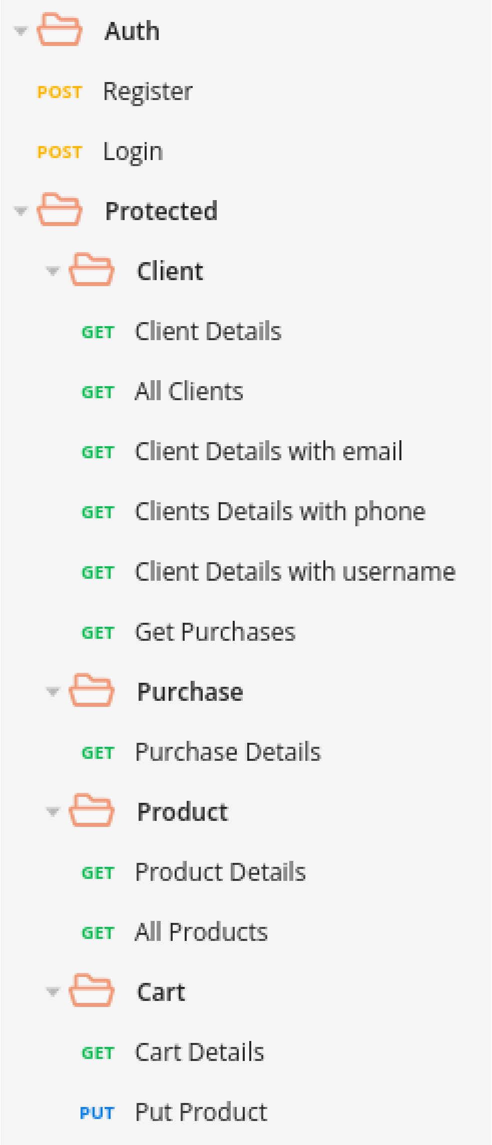


```
Auth
  POST Register
  POST Login
Protected
  Rider
    GET All Riders
    GET Rider Details
    GET Package Details by Rider ID
    GET Ride Details by Rider ID
    PUT Update Rider Location
    PUT Update Rider Availability
  Ride
    PUT Update Delivery Status
    GET All Rides
    GET Ride Details
  Admin
    GET Packages Statistics
    GET Riders Statistics
    GET Stores Statistics
  Package
    GET All Packages
    GET Package Details
    GET Package Details by Store ID
    PUT Update Package Status
    PUT Update Package Location
```

Fig. 5 - Endpoints on OnYourWay

24Meds API

Postman documentation



- Auth
  - POST Register
  - POST Login
- Protected
  - Client
    - GET Client Details
    - GET All Clients
    - GET Client Details with email
    - GET Clients Details with phone
    - GET Client Details with username
    - GET Get Purchases
  - Purchase
    - GET Purchase Details
  - Product
    - GET Product Details
    - GET All Products
  - Cart
    - GET Cart Details
    - PUT Put Product

Fig. 5 - Endpoints on 24Meds

6 - Conclusion

In conclusion, we did not achieve the expected results, as previously said in the *Limitations* section. In the beginning, the project was confusing to understand and it was hard to come up with a solution. With the help of our teacher with got to one and implemented it as well as we could. Although the final project was not the best version of our work, we learned a lot about project management and improved our technical and soft skills even more.

7 - References and resources

The documentation that helped us a lot during this project was the posts on Baeldung and also the practical guides and theoretical resources that Professor Ilídio gave to us.