deti universidade de aveiro
departamento de eletrónica, telecomunicações e informática

TQS: Product specification report

João Farias [98679], Diogo Monteiro [97606], Marta Fradique[98629], André Freixo[98495]
Final version

1 Introduction

1.1 Overview of the project

DeliverMe is an online platform responsible for delivery services, available anywhere in the country. This is a generic platform where any store that uses online deliveries can use this service by connecting the store to the platform **api**. After this, all clients of registered stores have the possibility to order products/services and utilize the home delivery mark.

The **BookShelf** store responsible for book delivery has its own website available and decided to join the **DeliverMe** service.

1.2 Limitations

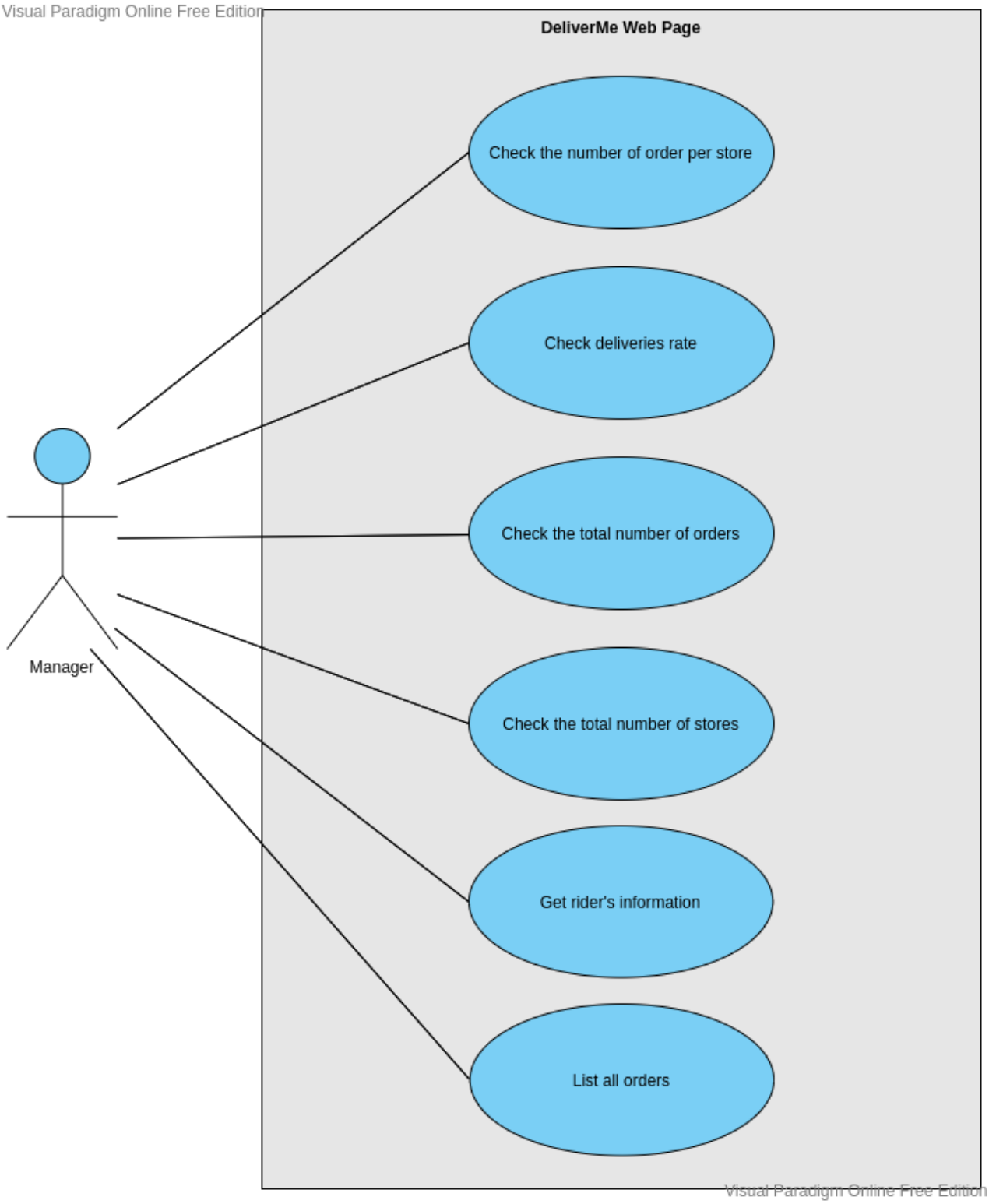
We could not successfully deploy our final solution, as of writing this report only the DeliverMe backend is deployed in the course provided virtual machine, which is accessible only within the university's network.
The delivery engine provides API endpoints to view deliveries' progress and status, but the specific business BookShelf did not use these in any way, in what could have been a tracking feature for clients.
Geographical and location information is sparsely used in the frontend, and no integration with external mapping engines was developed.
The specific business BookShelf does not have a registration feature implemented, having a test account for development and demo purposes.
Some cases of the use cases diagram, represented in the next section, are not implemented

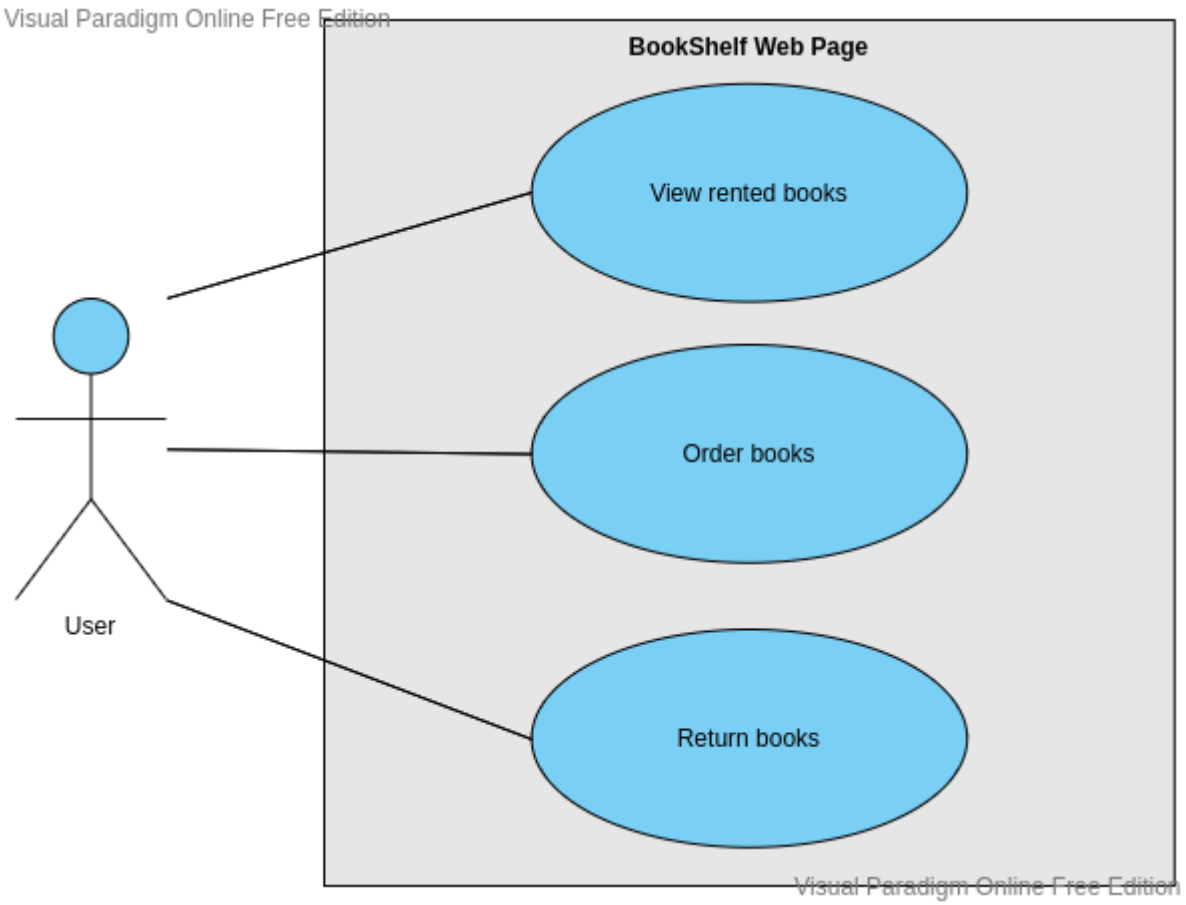
2 Product concept

2.1 Vision statement

This system will be used to make service distribution, the user orders something from a partner store and this order will be assigned to a driver that will collect the order and consequently deliver it to the client. The products available in the service will be provided by these partner stores that need to register in the platform.
Compared with similar delivery systems, while these ones have all stores in the same application, for example, **Uber Eats** has a list of restaurants and the user has to choose a specific one, the application we are presenting has his own platform, specific for each shop. So, this way, shop managers can have a bigger control in management and product selling.
To demonstrate this service, we are going to use **BookShelf**, a library that has the delivery service.

The next two diagrams represent the use cases diagram:


Visual Paradigm Online Free Edition
DeliverMe Web Page
Check the number of order per store
Check deliveries rate
Check the total number of orders
Check the total number of stores
Get rider's information
List all orders
Manager
Visual Paradigm Online Free Edition



2.2 Personas

Vera is 26 years-old, always lived and studied in Lisbon, **Catholic University**, where she finished her master's degree in Business two years ago. She has a leader profile, loves new challenges and beyond all these characteristics, Vera is accustomed to use new technologies which facilitates her adaptability into new business. Overall, Vera is a good manager and is available to face a good challenge.

Nicolau is 50 years-old, married for 25 years and works as a librarian in a library near Aveiro, where he lives with his wife and his two sons: André, 13 years-old and João, 20 years-old. With the online world growing so fast, he set

his business online and needs an external enterprise that delivers the books in clients' houses.

Beatriz is a young girl, 19 years-old, lives with her parents and his younger brother, and her favourite hobby is reading books. As she reads a lot of books, it isn’t economically viable to buy them all, so she often goes to the library and rents the books she desires. To prevent displacement, she would like to order the books in an application and receive them in her house.

Jacinto is 43 years-old, and lives with his wife and his young daughter, Maria. He has got a motorcycle and to avail his free time and earn some extra money to spend on his daughter's course, he would like to associate with a company like **UberEats**.

2.3 Main scenarios

1. **Scenario nº 1: Deliveries Manager**

Vera uses this platform to keep in track not only the service growth (stores information), but also rider's information.

2. **Scenario nº2: Store Manager**

Nicolau, due to competition, is facing some problems selling books. To adjust his business, he subscribes to this service allowing clients to receive the books in their homes and this way, increases his business.

3. **Scenario nº3: Client**

Beatriz, instead of going to the library every time she wants to buy a new book, she orders the book from the web and just needs to wait for it in her house.

4. **Scenario nº4: Courier**

Jacinto uses his motorcycle and distributes the orders from different provenances to people, using the platform available.

2.4 Project epics and priorities

For this project's management, an agile method was planned with weekly sprints. The following Epics were specified and given a priority and score:

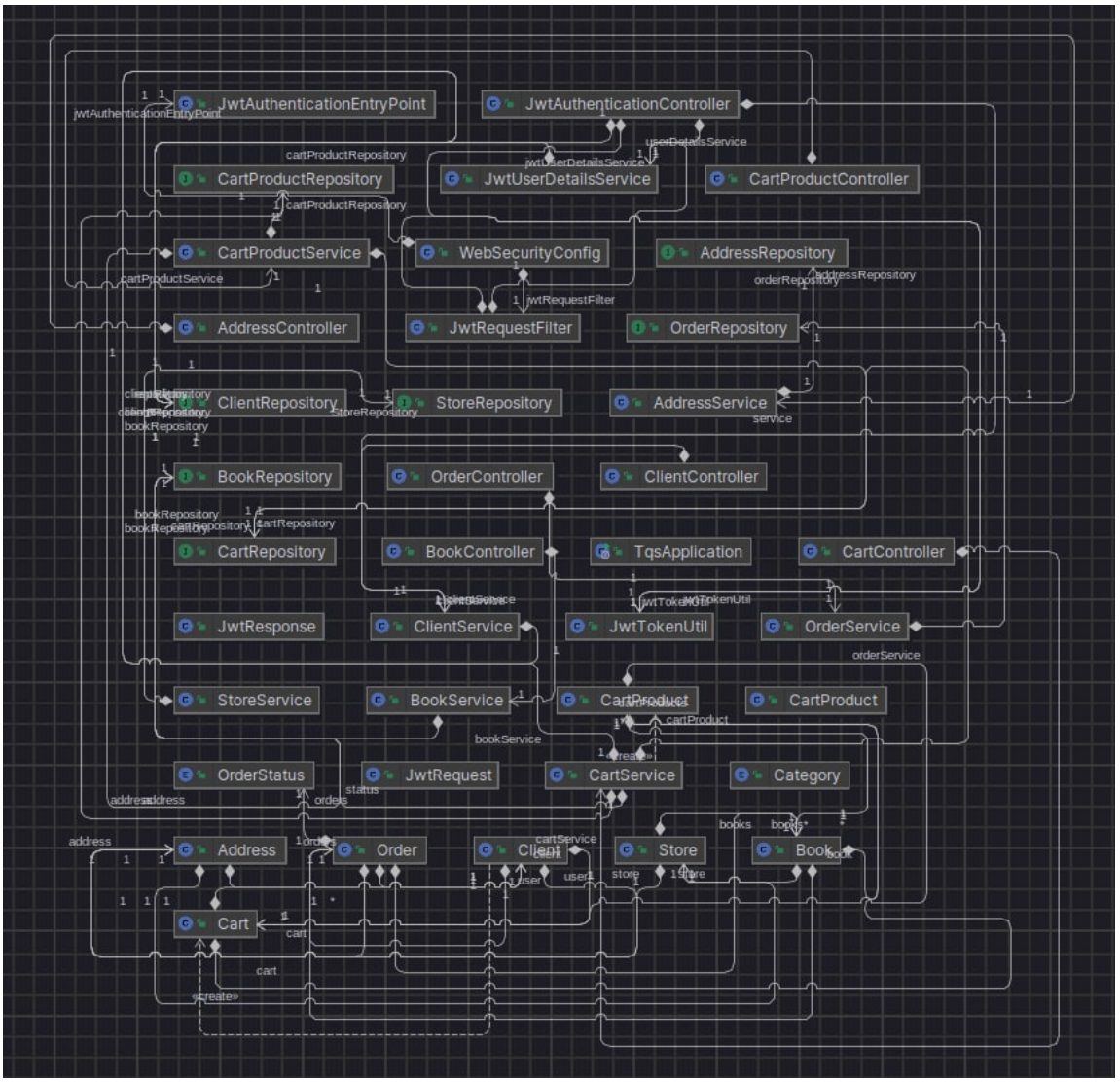
Story points scale (1-8)

- **(High) 5 -** The web applications supports logins (manager and rider for engine, client for specific);
- **(High) 7 -** The specific application can order services from generic app and check their status;
- **(High) 4 -** The rider can accept delivery requests on application;
- **(Low) 2 -** Rider can access his profile and see not only the past orders, but also the reviews associated with them;
- **(High) 2 -** The client can order books from site;

- **(Med) 3 -** The client can view their book orders' status and rate delivery service;
- **(Low) 3 -** The client can manage his profile information;
- **(High) 4 -** The manager can access the web application, check ongoing deliveries, check the rider's information and some extra information (historical data, total deliveries, etc…);
- **(Very Low) 8 -** Push notifications

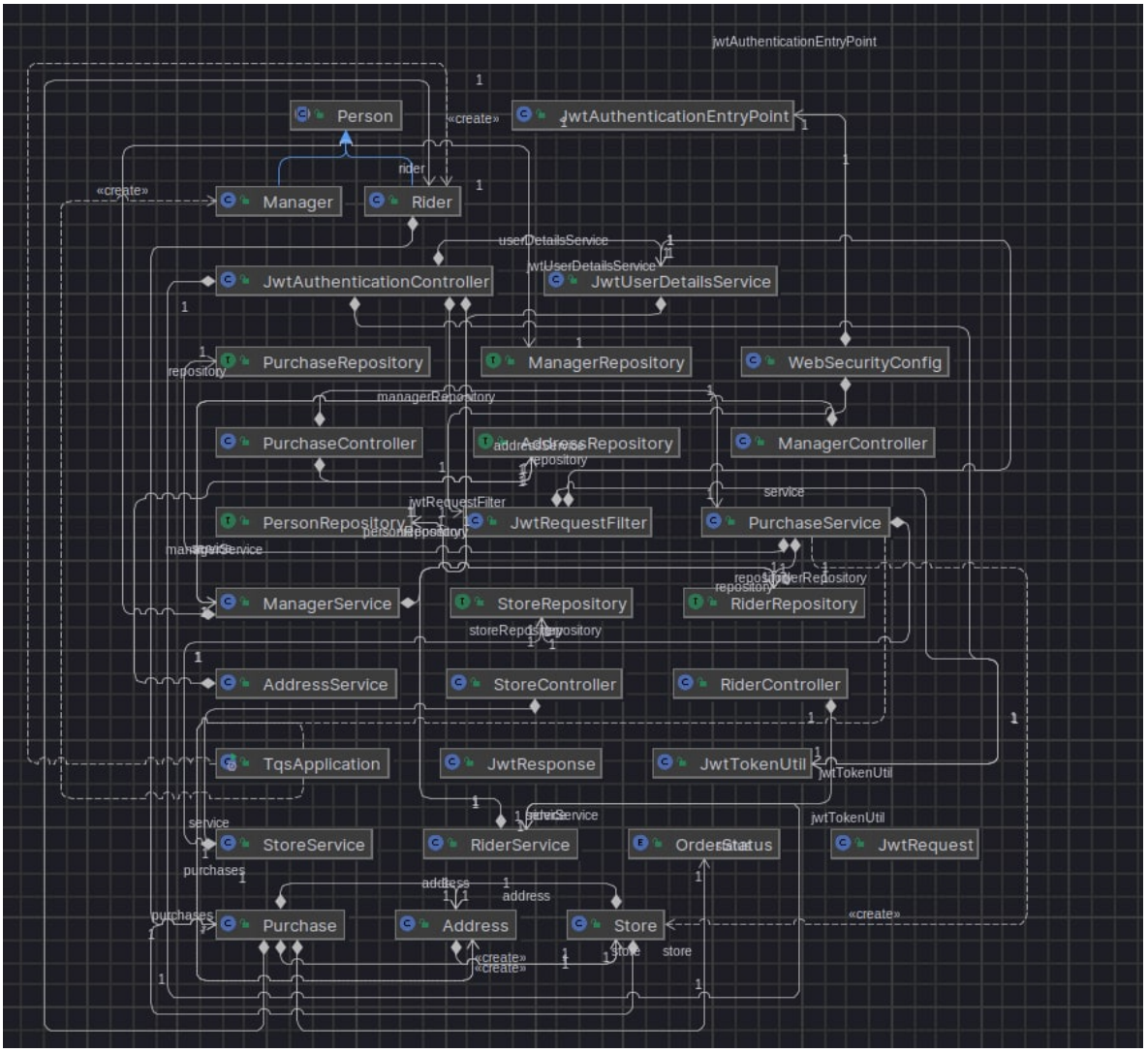
3 Domain model

Bookshelf is the specific service created to integrate with DeliverMe platform. It is composed by 7 entities: **address**, **book**, **cart**, **client**, **orders**, **store** and **cart_product**. As this service is the specific one, there are entities, like **Book,** that belong exclusively to this kind of store. If a music store wants to join to DeliverMe service, its entities and relations are different from this ones.



Bookshelf API Class Diagram

DeliverMe is the deliveries generic service supported by a database in **postgres.** There are 4 main entities: **address**, **store**, **purchase** and **person.** As the main objective of this system is to integrate with other stores, the main characteristics of each entity are a bit generic to facilitate the integration.



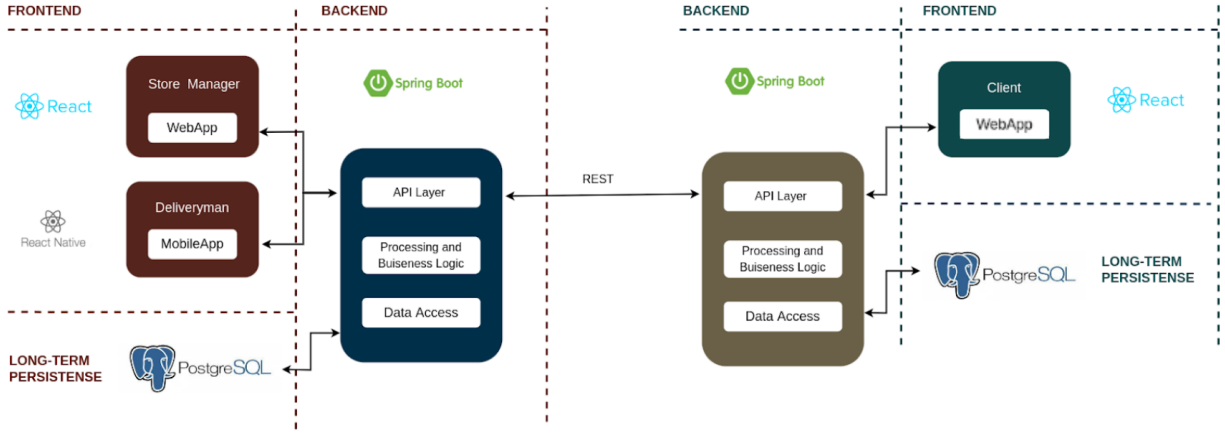
DeliverMe API Class Diagram

4 Architecture notebook

4.1 Architectural view

The solution can be split into two subprojects, the delivery engine DeliverMe and the specific business BookShelf. Both use Spring Boot for their backend APIs, with PostgreSQL databases for persistence. Reactjs is used for web applications, those being the management site for DeliverMe and the online

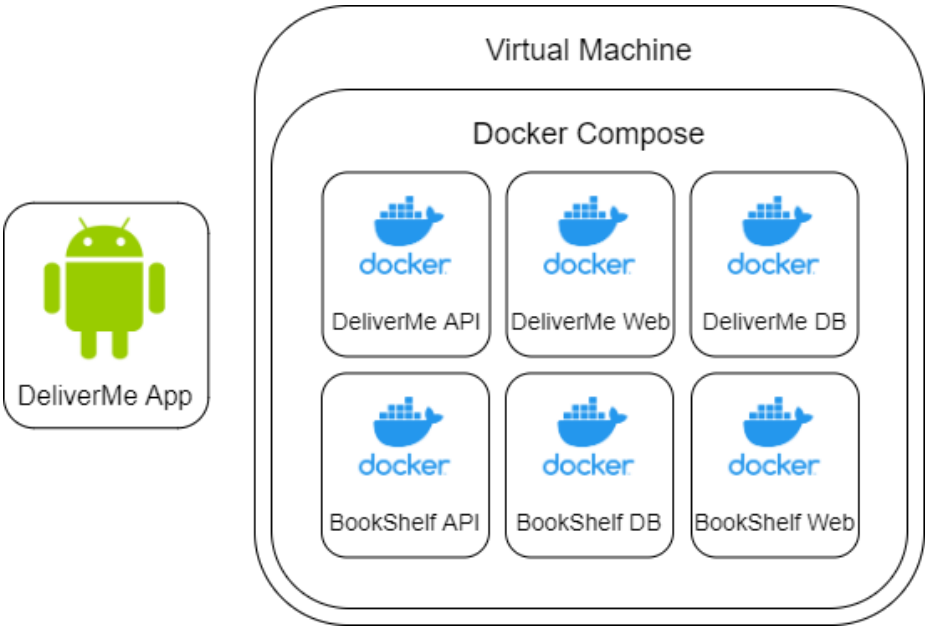
store in BookShelf. Finally, React Native is the technology used for DeliverMe's mobile app for couriers.



The Reactjs and React Native applications interact with their APIs through HTTP requests, and BookShelf communicates with DeliverMe through HTTP requests as well.

4.2 Deployment architecture

To deploy the solution, the APIs, DBs and webapps are containerized with Docker and configured with Docker Compose, which allows easy mapping of ports, setup of inter-module dependencies, and management of virtual networks. The Dockers run on a virtual machine, exposing the service's ports. The mobile app, being developed in React Native, can be built to run on Android and iOS.
For a more realistic scenario, the DeliverMe and BookShelf modules could be deployed on different machines, but we opted to run all of them in the same virtual machine.



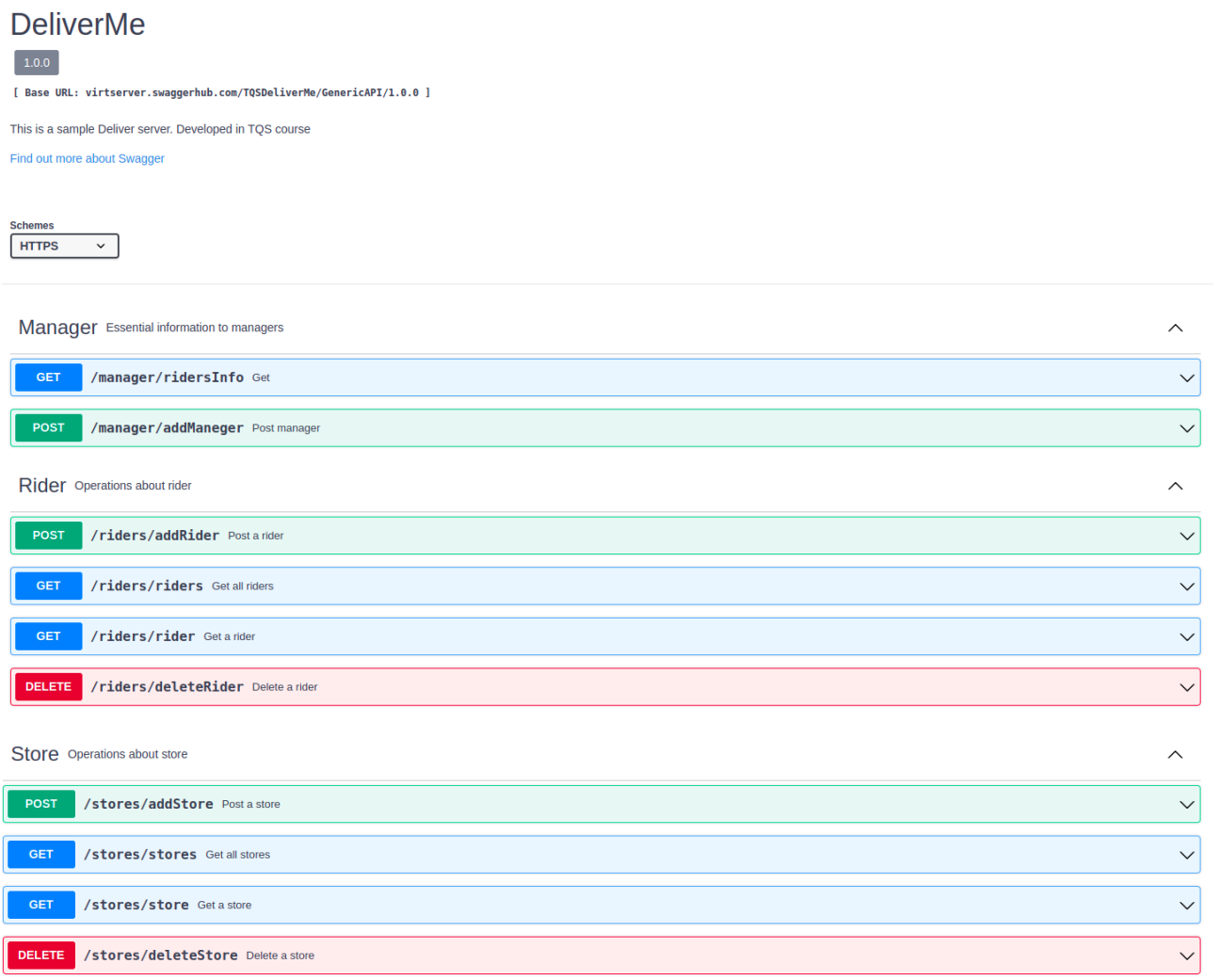
It should be noted that despite our efforts, we could not fully deploy all the modules due to errors we could not resolve. As of this report's writing, only the DeliverMe API and DB are deployed.

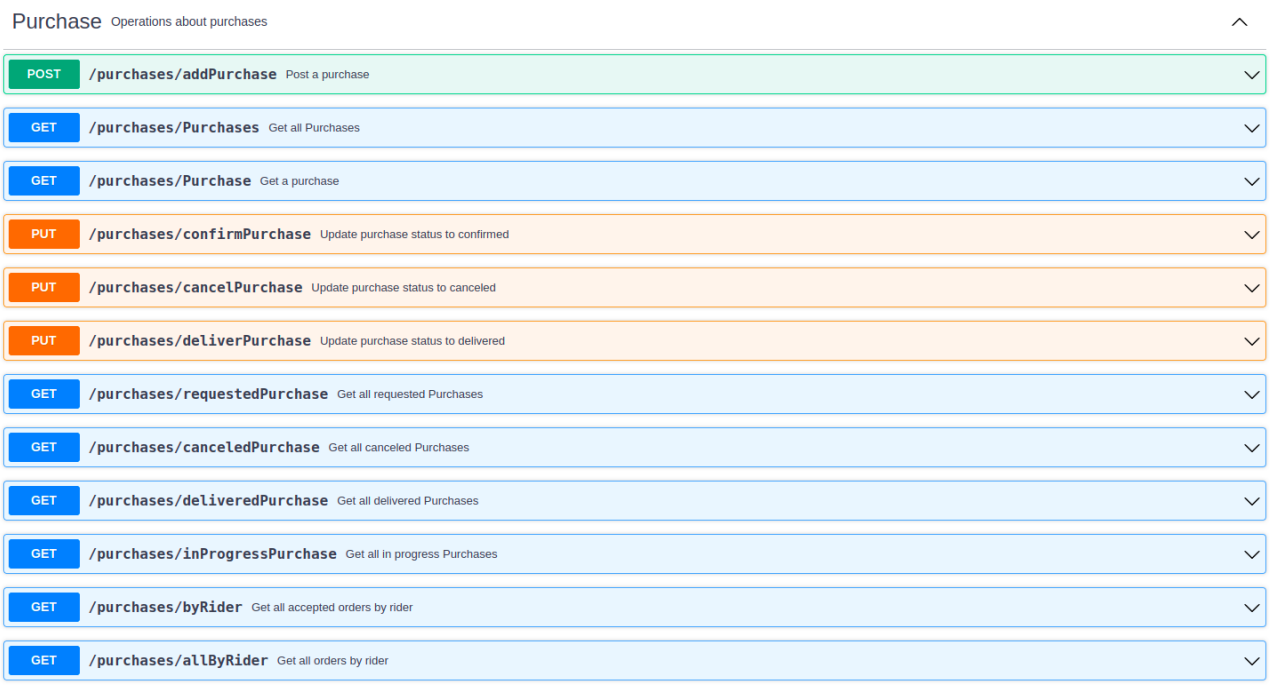
5 API for developers

Regarding API organization both API's , generic service, **DeliverMe**, responsible for home deliveries by riders and Specific, **BookShelf,** responsible for store organization, have their own documentation

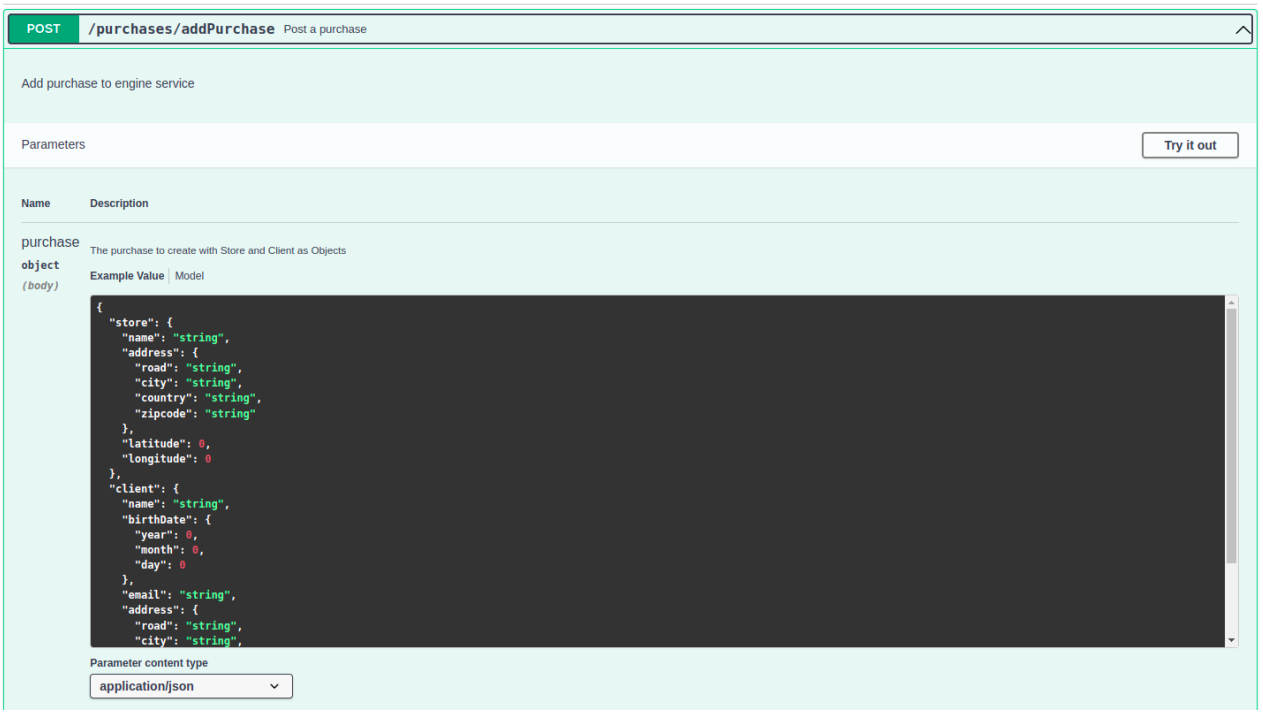
available in swagger. The next two figures represent the documentation, but it is also possible to access it through this link.

- DeliverMe API

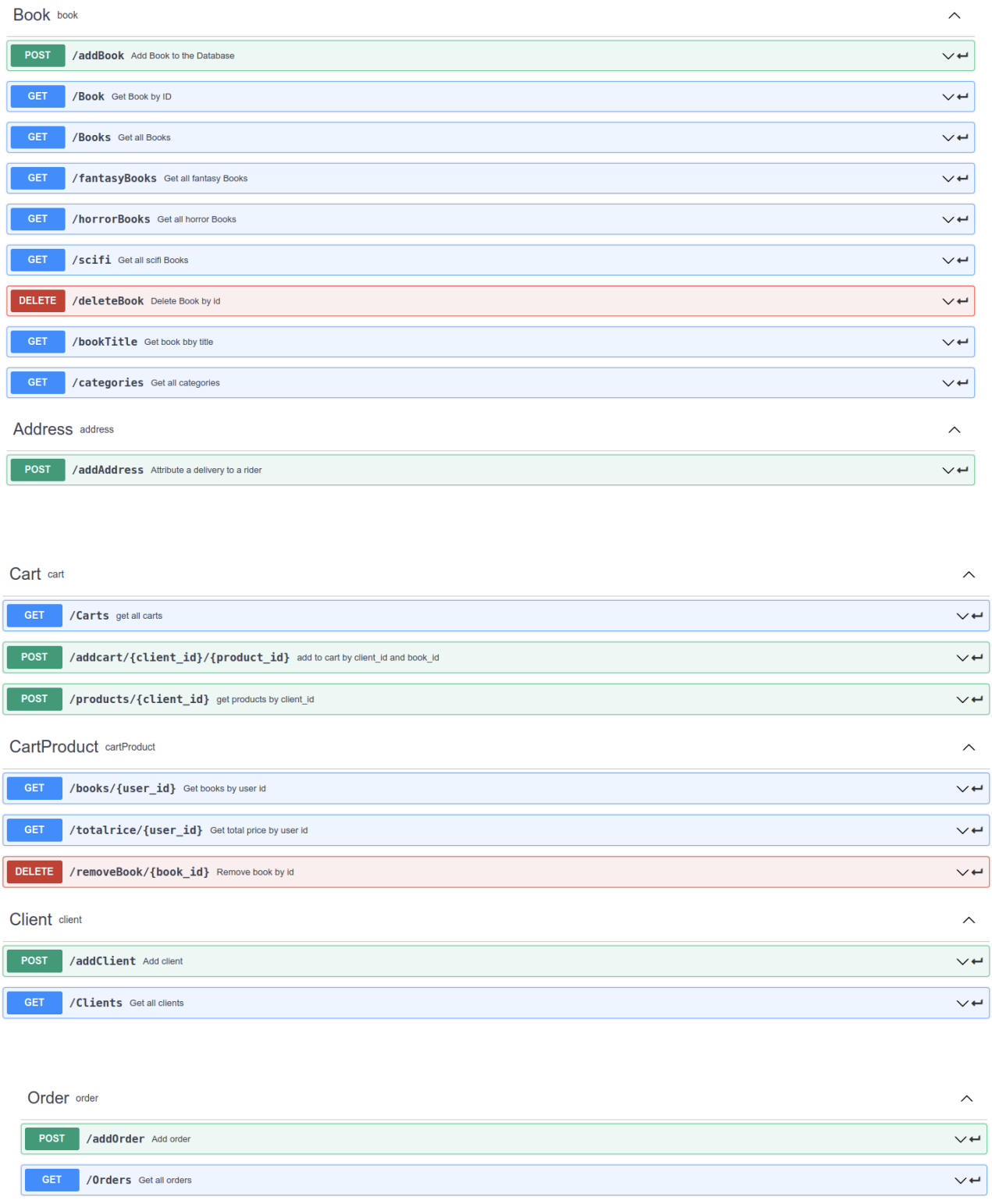




This documentation is important, once any store may use the delivery system. In this case, the store developer just needs to check the endpoints and all methods available to connect the store to the engine service. It is also important to refer that the request bodies are shown in endpoints documentation, as it is possible to discern in the following image:



- Bookshelf API



6 References and resources

Following are references and resources that assisted in development, such as frameworks' documentation and guides:

React Native Documentation
https://reactnative.dev/docs/getting-started

React Navigation Documentation
https://reactnavigation.org/docs/getting-started

“Dockerizing a React App” (Michael Herman)
https://mherman.org/blog/dockerizing-a-react-app/

“Stack overflow”
https://stackoverflow.com/