TQS: Product specification report

Raquel Paradinha [102491], Paulo Pinto [103234], Tiago Carvalho [102142], Miguel Matos [103341]

1 Introduction

1.1 Overview of the project

Nowadays, many people view a simple act like going to a store to buy or pick up a product as something tiresome, uninteresting, or even scary. Many companies are already investing in methods of delivering their products to the homes of their loyal customers to combat these issues. However, a large number of stores, especially those with limited delivery services, are dissatisfied because they are not able to meet the demands of customers who would be willing to purchase if the delivery or pick up process were more convenient.

Another issue in today's society is the lack of people using spaces such as small local cafes or long-standing shops that haven't experienced growth and are losing their customer base. The consistent absence of people frequenting these spaces becomes worrisome as many are forced to close due to this problem.

Our platform, Take It Easy, is a network of Pick-Up Points that, with the support of distribution companies and delivery points, facilitates the delivery and pickup of orders, providing greater convenience and freedom for eStores to send their products to their customers.

Take It Easy is the platform that comes to fulfill these objectives and add necessary functionalities for those who care about the details of their order.

In the web application, users may track their order and make adjustments regarding the delivery and pickup of the product, without any concerns about the shipping company. Additionally, space owners who wish to join the Pick-Up Points network can easily submit their membership request through the application.

1.2 Limitations

Overall, the product has fewer implemented stories than we initially intended, due to the small time window we had to develop the project, alongside other big projects from other classes.

However, we are still adamant that the produced work still displays good testing and quality assurance practices.

2 Product concept

2.1 Vision statement

Problem

Nowadays, many people view a simple act, such as going to a store to buy or pick up a product, as something tiresome, uninteresting, or even scary. Many of today's companies even invest in methods of transporting their products to the homes of their loyal customers in order to combat these issues.

However, a large part of stores, especially those with less extensive delivery services, are dissatisfied because they cannot meet the demands of customers who would be willing to buy if the delivery or pickup were more convenient.

Another problem of modern times is the weak participation in spaces such as small local cafes or old shops that have not grown and are losing their audience. The constant absence of people frequenting these places becomes worrying in that many are forced to close due to this issue.

Objectives

The goal of our platform is precisely to solve the problems presented above. With the use of our system, the most needy stores, which suffer from the same setbacks, have the opportunity to expand their buyer horizons, as well as respond to other needs related to their deliveries.

Furthermore, spaces that want to increase their attendance can use our platform to create new reasons and opportunities for people to visit these places, thus increasing their memberships and, consequently, their sales and profits.

Concept

Our platform, Take It Easy, is a network of Pickup Points scattered throughout the country that connects distributing companies and pickup points, facilitating the delivery and pickup of orders, giving greater abstraction and freedom to eStores to send their products to their customers.

Take It Easy is the platform that fulfills the aforementioned objectives and also adds necessary features for those who care about the details of their order. In the web application, users can track their order and redefine details regarding delivery and pickup of the product, without any concern about the

transportation entity. Additionally, those responsible for spaces that want to join the network of Pickup Points can request membership through the application in a simple and clear way.

2.2 User Types and Stories

For user types, we identified 4 different types of users that can interact with the system: admins, store clients, store owners, and PickUpPoint owners.
For each of these, we layed out the stories associated to each, and prioritized them.

Customer

- (high) Customer can place an order

  As a customer, I want to place an order to buy what I need.

- (high) Customer can check the order status

  As a customer, I want to check the status of my order at the store/carrier to review the delivery details.

- (med) Customer can receive notifications

  As a customer, I want to receive notifications when the order status changes to stay updated.

- (low) Customer can modify delivery information

  As a customer, I want to modify the delivery information (location/date) to better accommodate my availability.

Seller

- (low) Seller can associate with Take It Easy

  As the owner of an online store, I want to associate with Take It Easy as an eStore to start using their services.

Pick-up Point Partner

- (high) Partner can receive and record an order

  As a partner, I want to receive and record an order from the delivery person to confirm that the shipment has arrived.
  When an order arrives at my store:
  - The order should be stored in a secure location.
  - The order number should be noted.
  - In the system, I should be able to:
    - See which orders I need to receive and mark the ones I received.
    - Notify the customer that the order has arrived at my store.

- (high) Partner can deliver an order

As a partner, I want to deliver orders to customers in order to fulfill the shipments.

When a customer requests an order that has already arrived at my store and belongs to them:

- I should deliver it to them.
- In the system, I should be able to:
  - Mark the delivery of the order.
  - Notify the customer that their order has been delivered.

- (med) Partner can check the order status

As a partner, I want to monitor the order status related to my business to stay updated.

- (low) Partner can accept a return

As a partner, I want to accept returns to accommodate customer preferences.

- (low) Non-associated partner can associate with Take It Easy

As a partner, I want to associate with Take It Easy to become a Pick-Up Point.

Take It Easy Administrator

- (high) Administrator can approve a new eStore

As an administrator, I want to approve new eStores to expand the network of sellers.

- (high) Administrator can approve a new Pick-Up Point

As an administrator, I want to approve new Pick-Up Points to expand the network of delivery points.

When a physical store interested in becoming a Pick-Up Point provides all the necessary data for its association with the platform, I should:

Accept that store as a Pick-Up Point partner.

In the system, I should be able to:

Review and accept or reject requests for platform association.

- (high) Administrator can view the order status

As an administrator, I want to view the order status to verify that everything is in order.
I should have constant access to the order status and its details.

- (med) Administrator can view operational statistics

As an administrator, I want to view operational statistics to gather system details.

- (med) Administrator can access Pick-Up Points

As an administrator, I want to access available Pick-Up Points to ensure everything is in order.

- (low) Administrator can access associated eStores

As an administrator, I want to access associated eStores to gather system details.

- (low) Administrator can remove a Pick-Up Point

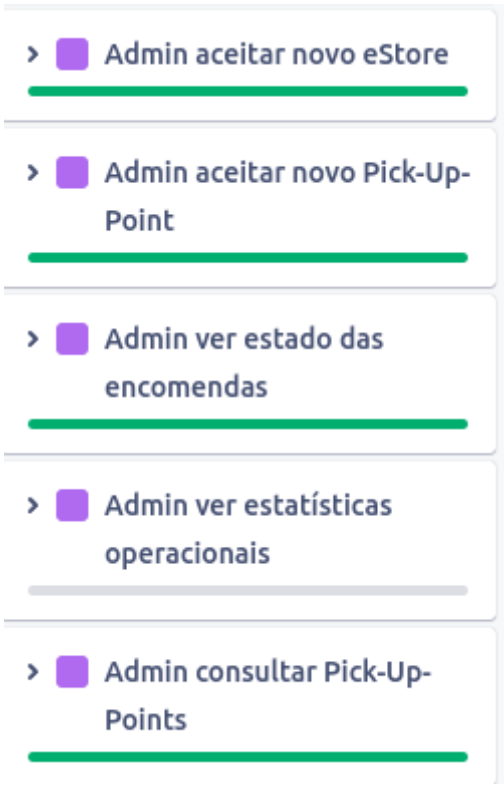
As an administrator, I want to remove a Pick-Up Point to update the network of delivery points.

- (low) Administrator can remove an eStore

  As an administrator, I want to remove an associated eStore to update the network of sellers.

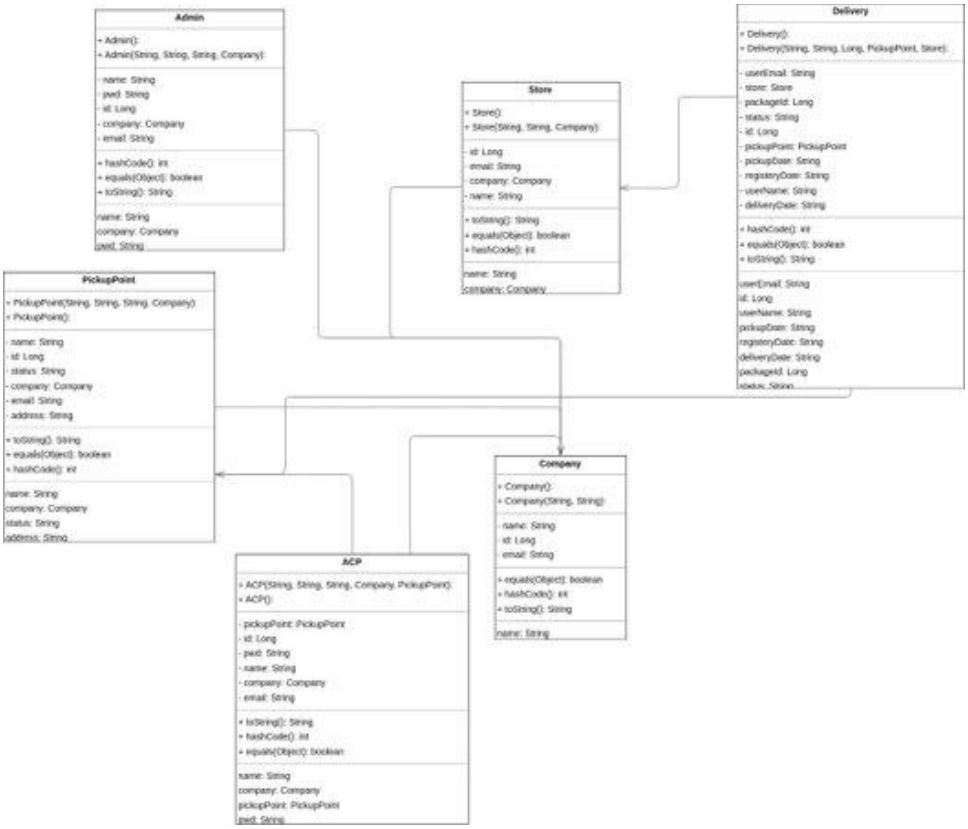
2.3 Project epics and priorities

When collecting the requirements for our application, we made epics, which are high-level user stories that are meant to represent a possible flow of action in our solution.
Each epic had a priority associated to it (low, medium or high), which helped us prioritize epics with bigger impact on the final solution.
Through iterations, the epics we were working on were split into issues, and these were assigned to a develeper to work on.



Above, we can see some of the Epics we had planned for the final solution. Out of the displayed ones, only the 4th one was not implemented; it was the one with the lowest priority and we didn't have the time to spare, so we decided to leave it apart.

3 Domain model



This is the domain model for the TakeItEasy API. The main classes are the Delivery, which the system revolves around of, Store, which represents the stores where the deliveries may come from, and the PickUpPoints, which are the physical places to where the deliveries are shipped.

4 Architecture notebook

4.1 Key requirements and constrains

Non-Functional Requirements:

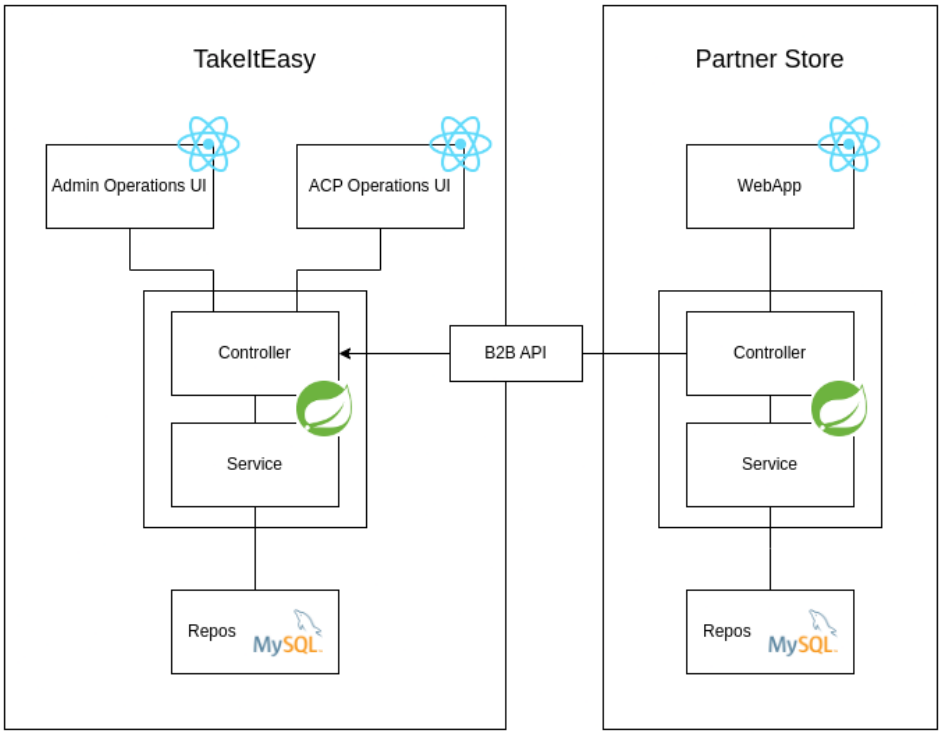
- Scalability: the application should support many users
- Reliability: it shouldn't frequently crash
- Usability: the application should be intuitive
- Availability: the system should be available to the user

Functional Requirements:

- The system should have a varying number of PickUpPoints.
- A new PickUpPoint should be able to become an associated partner PickUpPoint.
- An admin should be able to remove a partner PickUpPoint.
- The system should have a varying number of Stores.
- A new Store should be able to become an associated partner store.
- An admin shoud be able to remove a partner store.
- The system should allow for a client to make a purchase on a partner store and have it delivered to a PickUpPoint.
- The system should keep track of deliveries.
- The system should provide a way for administrators to keep track of the whole system.

- The system should be able to deliver a package, terminating its lifecycle on the system.
- A partner PickUpPoint should be able to see the deliveries associated to its own service.

4.2 Architetural view



The architecture is comprised of two types of modules – the main system and the partner store. Both modules will have their backend layer developed using Spring Boot and their presentation layer developed in React.

The TakeItEasy system makes available an API that other businesses can call and use to access the functionalities provided by the service.

For persistence, we decided to go with a MySQL database implementation, given its versatility and ease of use.

4.3 Deployment

We first attempted to deploy the solution using Google's Cloud Run service; however, we were having some issues with it, so we decided to use the VM made available to us by the university.
We developed the pipelines that deploy these using Docker containers on the VM (docker was installed on the VM to enable the image building process).

5 API for developers

Our main API was subdivided into 3 different controllers. Each controller deals with a part of the data. The delivery controller is responsible for handling the processes related to these entities; it has endpoints to create a new one, update their status, or fetch them. These are meant to be called by the Store's backend and the frontend applications.
Both the store and pickup-point controllers are meant to be used by the administration of the TakeItEasy company, since they provide the means to manage the existing stores and PickUpPoints.

The API documentation can be found at SwaggerUI.

6 Conclusion

Overall, we feel like the project still had a little of work left, in terms of proper CD pipelines and frontend development.
However, the main focus of this project was to work on our quality assurance skills, and we feel like we had the chance to explore the various aspects within these, and actually made effort to put these practices to use.
Overall, we are content with the work done, despite the difficulties.