Ministry of Higher Education and Scientific Research

And Information and Communication Technologies

Générale Direction of Technologies studies

**Higher Institute of Technologies studies of Bizerte**

**Information Technology Department**

**INTEGRATION PROJECT REPORT**

Subject: Development of purchasing and sales applications

**Realized By:** Sprint Squad Team

**Supervising teacher:** Miss Afef Gafsi

**Academic year: 2023-2024 DSI31**

**Summary**

[General Introduction 1](#_Toc150426181)

[Chapter 1: Preliminary study 2](#_Toc150426182)

[Introduction 2](#_Toc150426183)

[1.1. Description of existing equipment 2](#_Toc150426184)

[1.2. Existences Criticisms 2](#_Toc150426185)

[1.3. Proposed Solution 2](#_Toc150426186)

[1.4. Adapted working method. 3](#_Toc150426187)

[a SCRUM project management method 3](#_Toc150426188)

[b Done criteria 4](#_Toc150426189)

[c Sprints Time Box 4](#_Toc150426190)

[1.5. Working tools 4](#_Toc150426191)

[a Frameworks used 4](#_Toc150426192)

[b Software development 5](#_Toc150426193)

[Conclusion 7](#_Toc150426194)

[Chapter 2: Product Backlog planning 8](#_Toc150426195)

[Introduction 8](#_Toc150426196)

[2.1. User’s identification 8](#_Toc150426197)

[2.2. User Stories 9](#_Toc150426198)

[2.3. Application Flows 10](#_Toc150426199)

[2.4. Product Backlog 10](#_Toc150426200)

[Conclusion 12](#_Toc150426201)

[Chapter 3: Sprint 1 13](#_Toc150426202)

[Introduction: 13](#_Toc150426203)

[3.1 Sprint goals 13](#_Toc150426204)

[3.2 Sprint backlog 13](#_Toc150426205)

[3.3 Sprint 1 implementation 17](#_Toc150426206)

[a System description phase 17](#_Toc150426207)

[b Analyze 19](#_Toc150426208)

[c Conception phase 21](#_Toc150426209)

[d Interfaces Models 24](#_Toc150426210)

[3.4 Sprint retrospective 24](#_Toc150426211)

[Conclusion 25](#_Toc150426212)

[Chapter 3: Sprint 2 26](#_Toc150426213)

[Introduction 26](#_Toc150426214)

[4.1 Sprint goals 26](#_Toc150426215)

[Bibliography 27](#_Toc150426216)

**List of figures**

[Figure 1: NodeJS and Express [1] 4](#_Toc150418544)

[Figure 2: Android Native with Koltin [2] 5](#_Toc150418545)

[Figure 3: Spring boot logo [3] 5](#_Toc150418546)

[Figure 4: VsCode logo [4] 6](#_Toc150418547)

[Figure 5: Android Studio logo [5] 6](#_Toc150418548)

[Figure 6: IntelliJ ide logo [6] 6](#_Toc150418549)

[Figure 7: Postman logo [7] 7](#_Toc150418550)

[Figure 8: Github logo [8] 7](#_Toc150418551)

[Figure 9: Application browsing flow 10](file:///C:\Users\Hakim\OneDrive\Bureau\SwapShop\Project%20Documentations\Rapport\Rapport%20Projet%20Intégration%20(2).docx#_Toc150418552)

[Figure 10: Use Case diagram 18](#_Toc150418553)

[Figure 11: System Sequence Diagram 19](#_Toc150418554)

[Figure 12: Participants Class Diagram 20](file:///C:\Users\Hakim\OneDrive\Bureau\SwapShop\Project%20Documentations\Rapport\Rapport%20Projet%20Intégration%20(2).docx#_Toc150418555)

[Figure 13: Domain Model Diagram 21](#_Toc150418556)

[Figure 14: Sequence Diagram "Crud Products" 22](#_Toc150418557)

[Figure 15: Sequence Diagram "Crud Categories" 23](#_Toc150418558)

[Figure 16: Conception Class 24](#_Toc150418559)

**List of Tables**

[Table 1: Product Backlog 10](#_Toc150426217)

[Table 2: Sprint 1 backlog 13](#_Toc150426218)

[Table 3: ToDo for amelioration 24](#_Toc150426219)

# General Introduction

# Chapter 1: Preliminary study

## Introduction

In this chapter, we are going to present the different functionalities of our application, as well as its critics, the proposed solution and a detailed description of the final solution.

## Description of existing equipment

In the current method of selling and buying products, the customer or supplier has to go to a store or company or someone to buy a product on the spot, talk to them, and all transactions between them are done through documents such as purchase orders, and they receive money hand over fist.

## Existences Criticisms

In the current system, to purchase products there is some criticisms like:

* **No availability:** customers need to contact the seller to arrange a meeting when they are available.
* **Wasting time and money:** They must then arrange transportation, and sometimes unforeseen issues may prevent them from completing the transaction, resulting in wasted time and money
* **Scamming and fraudulent activities:** There is no customer service available, which has led to a significant number of scams and fraudulent activities. Customers cannot feel secure because they must carry cash with them to make purchases, putting them at risk of theft.

## Proposed Solution

We are proposing our innovative application for buying and selling products. Our platform empowers users with an array of essential features, enabling seamless transactions and fostering a secure and trustworthy marketplace. With our application, sellers can effortlessly list their products for sale, while buyers can explore, purchase, and even add items to their Wishlist. We've implemented a robust and secure online payment system to ensure a smooth transaction process.

In addition to the core functionalities, we've integrated optional shipping capabilities, providing users with the choice of convenient delivery with tracking. To enhance user trust and reliability, we've introduced verified accounts in fact users must verify their account with their phone number to us our application, with help eliminating fake profiles, spam, and non-serious users from our platform. To enhance user trust and reliability, we've introduced verified accounts in fact users must verify their account with their phone number to us our application, with help eliminating fake profiles, spam, and non-serious users from our platform. Moreover, our application offers badges and a reward system, encouraging sellers to build strong relationships with their customers.

To further enhance user support, our platform now features a real-time chat system, allowing buyers and sellers to communicate seamlessly. Additionally, we've introduced a ticket system, ensuring that customers have a direct channel to resolve any issues and assert their rights. if the problem isn’t solved, buyers can transfer tickets to support team, we take user satisfaction seriously, and in line with this commitment, sellers are held to high standards. With our 'three strikes and you're out' policy, sellers are given three chances to maintain account integrity. After three strikes, their accounts will be subject to removal.

Should any issues or concerns arise, our platform includes a dedicated feature for users to contact either the admin or the seller for prompt assistance. Our application is built using Android with Kotlin, utilizing a microservices architecture with Node.js and Express (part of the MEAN stack technology), as well as Spring Boot, ensuring a high-performance and scalable platform for a seamless buying and selling experience.

## Adapted working method.

### SCRUM project management method

For the development of our projects, we are decided to choose SCRUM like a method of the project management with:

* Miss Afef Gafsi: the Scrum master which she is the responsible to help us to move forward independently, while constantly striving to improve.
* Developers: We are playing the developers roles that we are supposed to developing the different apps and architectures of it.
* Product Owner

### Done criteria

We are supposed that a user story is done, it should be terminated and work fluently.

### Sprints Time Box

We take as an estimation for every sprint 2 weeks.

## Working tools

### Frameworks used

During the development of our project, we are using a variety of Frameworks like:

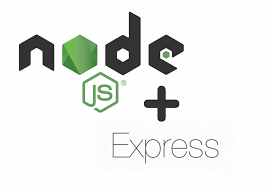


Figure 1: NodeJS and Express [1]

We are using NodeJS and Express for developing micro services, for the web version.



Figure 2: Android Native with Koltin [2]

To developing the mobile version of our application we are using the android native with kotlin as a programming language.



Figure 3: Spring boot logo [3]

We are also using Spring boot for the developing of the micro services for the web version.

### Software development

For the developing of our application, we are using a variety of software programs like:



Figure 4: VsCode logo [4]

For the development of NodeJS and Express, we are decided to use VsCode.



Figure 5: Android Studio logo [5]

For the development of the mobile version we decided to use android studio.

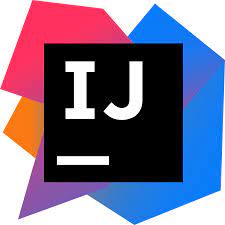


Figure 6: IntelliJ ide logo [6]

For the development for the micro services in spring boot we decided to use IntelliJ ide



Figure 7: Postman logo [7]

We are using Postman for testing the API of the backend which develops with NodeJS express and SpringBoot.



Figure 8: Github logo [8]

We are using Github for colloaborating and for coding as a teams, means everyone have a code deploying in github and he can change whenever he want.

## Conclusion

In this chapter we’ve presented a preliminary study of our application, we’ve presented the organization's need for this application as well as its benefits, we’ve also presented all the system's functionalities.

# Chapter 2: Product Backlog planning

## Introduction

In this chapter, we are going to present the different actors and their user’s stories, then we are going to present the product backlog.

## User’s identification

In this application we’re developing different functionalities for 3 actors:

A seller he has functionalities:

* Managing the products, he intends to sell.
* Reviewing annual activities and profits.
* Displaying featured badges on their profile to build trust with customers: badges are given automatically after replying for some conditions (example : sellers with receive a "**first product sold** "badge after they sell their first product)
* Keep in contact with the admin if there is a problem.
* Display a list of a buyer's past orders for reference.

A buyer has functionalities:

* Search for products based on a range of criteria.
* Add & manage products Wishlist.
* Ensuring Secure Purchases, Whether Online or In-Person.
* Evaluate the services provided by sellers
* Place an order.
* Monitor order history and shipment tracking.
* Maintain open communication with sellers or our support team through a chat or a ticket system to address any product, shipping, payment, or other concerns.
* Enhance the user experience with a 'Recently Viewed' list, providing users with easy retrieval of their past interactions.

And an admin has functionalities:

* View and manage all users
* Manage the categories of the products.
* Keep in contact with sellers and buyers through a chat/ticket system and give users strike or ban their accounts if they violate the policies.

## User Stories

## Application Flows

Figure 9: Application browsing flow

## Product Backlog

Table 1: Product Backlog

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| N° | Description User story | Type | Priority | Complexity | Estimate | Acceptance criteria | Sprint |
| 1 | As a developer, I initialize the project with the necessary dependencies to properly manage the application. | TS | 1 | Low | 1 day | Functional project | 1  Initialization Project, category and product’s seller Crud |
| 2 | As an admin I would like to add a category | US | 1 | Medium | 2 days |  |
| 3 | As an admin I would like to delete a category | US | 1 | Low | 2 days |  |
| 4 | As an admin I would like to modify a category | US | 1 | Medium | 2days |  |
| 5 | As a seller I would like to add a product in my shop | US | 1 | Medium | 2 days | adding successfully |
| 6 | As a seller I would like to delete a product in my shop | US | 1 | Low | 2 days | deleting successfully |
| 7 | As a seller I would like to modify a product detail in my shop | US | 1 | Medium | 2 days | modifying  successfully |
| 8 | As a user I would like to see all the product in the application | US | 1 | Low | 1 day | Showing products successfully |  |
| 9 | As a customer I would like to check a product detail and it price | US | 2 | Low | 1 day | Showing products details successfully | 2  Crud’s products in customer’s  Wishlist and pass orders |
| 10 | As a customer I would like to search for a product using a various criteria | US | 2 | Medium | 2 days | Searching successfully |
| 11 | As a customer I would like to add a product in my wishlist to buy it later | US | 2 | Medium | 2 days | Adding successfully |
| 12 | As a customer I would like to delete a product in my wishlist | US | 2 | Low | 1 day | Deleting successfully |
| 13 | As a customer, I would like to pass an order to a product | US | 2 | Medium | 2 days | Orders passed successfully |
| 14 | As a customer, I would like to verficate the qualities of products before paying | US | 2 | Medium | 2 days | Verification successfully |
| 15 | As a customer, I would like to pay for the product online | US | 2 | Medium | 2 days | System of payment working successfully |
| 16 | As a customer, I would like to track my shipment | US | 2 | Medium | 2 days | Shipment tracked successfully |
|  | As a developer I developing a system of earnings badges to who’s have send 5 successful tracks | US | 3 | Medium | 1 day |  | 3  Authentication  Reviewing, badges system  And user’s statics |
| 17 | As a customer I would like to review the services of sellers | US | 3 | Medium | 1 day |  |
| 18 | As a customer I would like to add strikes if I don’t like a service of sellers | US | 3 | Medium | 1 day |  |
| 19 | As an admin, I would like to verificate the strikes after user’s add | US | 3 | Medium | 1 day |  |
| 20 | As an admin, I would ban sellers if he got 3 strikes | US | 3 | Medium | 1 day |  |
| 21 | As a seller, I want to showcase earned badges to build trust with buyers. | US | 3 | Medium | 1 day |  |
| 22 | As a user, I want to see badges on the profiles of other users or sellers | US | 3 | Medium | 1 day |  |
| 23 | As a user, I want to view the transaction history with real-time tracking data. | US | 3 | High | 2 days |  |
| 24 | As a seller I would like to check my profits in the year | US | 3 | High | 2 days |  |
| 25 | As a customer I would like to contact the seller or the admin if there is a problem | US | 3 | High | 3 days |  |
| 2 | As a user, I would like to authenticate to get an access to my space in the app | US | 1 | Medium | 4 days | Authentication  successfully |  |

In our project we got 3 sprints, the one of it releasing during 2 weeks:

* Sprint 1: Initialization Project, category and product’s seller Crud.
* Sprint 2: Crud’s products in customer’s Wishlist and pass orders.
* Sprint 3: Authentication Reviewing, badges system And user’s statics

## Conclusion

In this chapter, we’ve presented all the user stories related to our projects, also we’ve presented all the user’s identifications, and finally we’ve presented the product backlog classified by sprints.

# Chapter 3: Sprint 1

## Introduction:

In this chapter, we are going to present the sprint backlog, the implementation, and the different diagrams and conceptions.

## Sprint goals

* Adding a system to let the user to manage their products that he would buy.
* Adding a system to let the user to manage the categories of products

## Sprint backlog

Table 2: Sprint 1 backlog

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | User Story | Tasks | Owner | Estimation | Start | In progress | Done |
| 1 | As a developer, I initialize the project with the necessary dependencies to properly manage the application. | Install the developing environments | All members | 15 hours |  |  | X |
| Install the project and all their dependencies | 9 hours | X |  |  |
| Developing the eureka server service using spring boot |  |  |  |  |  |
| Developing the gatway server service using spring boot |  |  |  |  |  |
| 2 | As an admin I would like to add categories for products | Initialize the category management service using spring boot |  |  |  |  |  |
| Initialize the category management service using NodeJS |  |  |  |  |  |
| Adding the add categories functionality to category service using spring Boot |  |  |  |  |  |
| Adding the add categories functionality to category service using NodeJS | Farouk | 15 hours | X |  |  |
| Create a mobile user interface | Jasser | 5 hours | X |  |  |
| Creating an add category functionality in the category controller in kotlin | Hakim | 4 hours |  |  |  |
| 3 | As an admin I would like to delete categories | Adding the delete category functionality to category service using spring Boot |  |  |  |  |  |
| Adding the delete category functionality to category service using NodeJS |  |  |  |  |  |
| Create a delete button in the  mobile user interface | Haythem | 1/2 hour | X |  |  |
| Create a delete category functionality in the category controller in kotlin |  | 23 hours | X |  |  |
| 4 | As an admin I would like to modify a category | Adding the edit category functionality to category service using spring Boot | Farouk | 15 hours | X |  |  |
| Adding the edit category functionality to category service using NodeJS | Ahmed | 8 hours |  |  |  |
| Create a mobile user interface | Bayrem | 1 jour | X |  |  |
| Creating a modify category functionality in the category controller in kotlin |  |  |  |  |  |
| 5 | As a seller I would like to add a product in my shop | Initialize the products management service using spring boot | Hakim | 15 hours | X |  |  |
| Initialize the products management service using NodeJS | Bayrem | 9 hours | X |  |  |
| Adding the add products functionality to products service using spring Boot | Farouk |  |  |  |  |
| Adding the add products functionality to products service using NodeJS |  |  |  |  |  |
| Create a mobile user interface |  |  |  |  |  |
| Creating an add products functionality in the products controller in kotlin |  |  |  |  |  |
| 6 | As a seller I would like to delete a product in my shop | Adding the delete product functionality to product service using spring Boot | Jasser | 1 hour | X |  |  |
| Adding the delete product functionality to product service using spring Boot | 1 hour | X |  |  |
| Create a delete button in the  mobile user interface | 22 hours | X |  |  |
| Create a delete product functionality in the product controller in kotlin |  |  |  |  |  |
| 7 | As a seller I would like to modify a product detail in my shop | Adding the edit product functionality to product service using spring Boot | Hakim | 15 hours | X |  |  |
| Adding the edit product functionality to product service using NodeJS | Bayrem | 8 hours | X |  |  |
| Create a mobile user interface | Ahmed | 1 day | X |  |  |
| Creating an add products functionality in the product controller in kotlin |  |  |  |  |  |
| 8 | As a user I would like to see all the product in the application | Adding the show products functionality to product service using spring Boot |  |  |  |  |  |
| Adding the show products functionality to product service using NodeJS |  |  |  |  |  |
| Create a mobile user interface |  |  |  |  |  |
| Creating an show all products functionality in the product controller in kotlin |  |  |  |  |  |

## Sprint 1 implementation

### System description phase

Une image contenant texte, diagramme, croquis, ligne

Description générée automatiquement

Figure 10: Use Case diagram

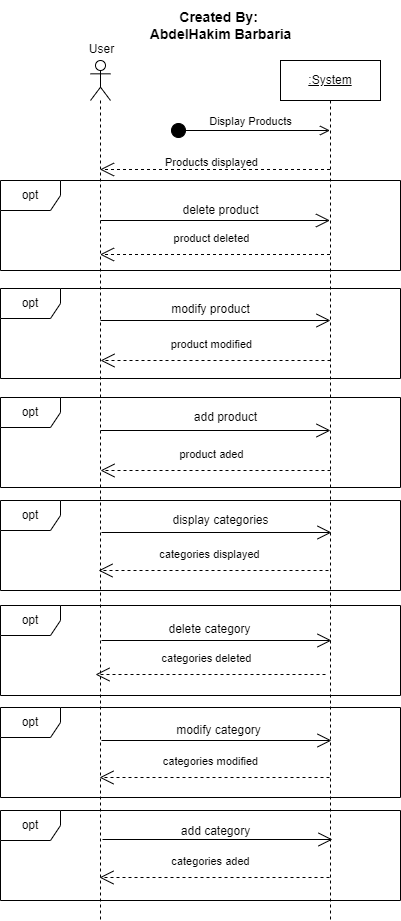


Figure 11: System Sequence Diagram

### b Analyze

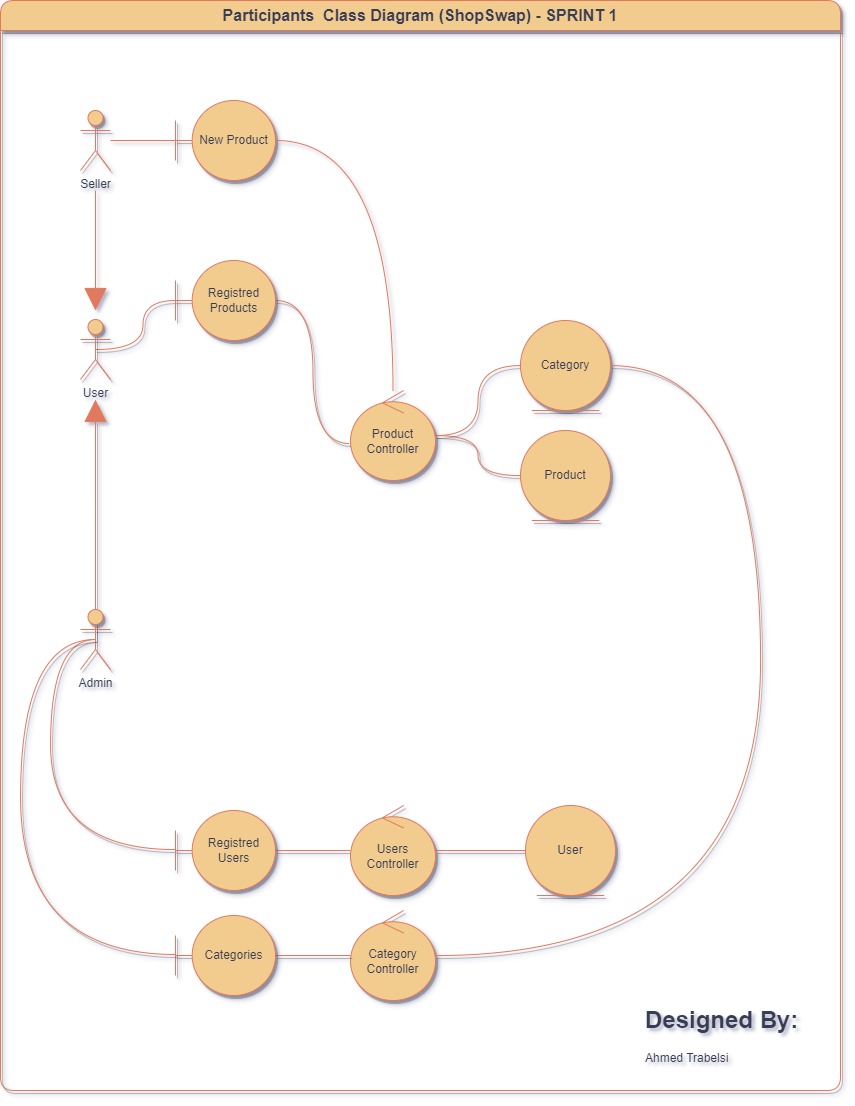


Figure 12: Participants Class Diagram

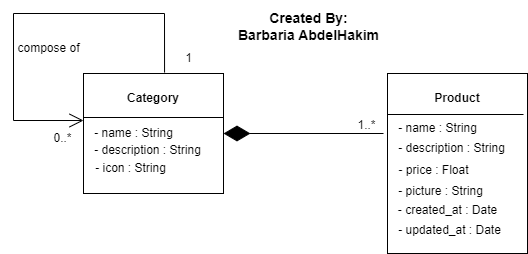


Figure 13: Domain Model Diagram

### Conception phase

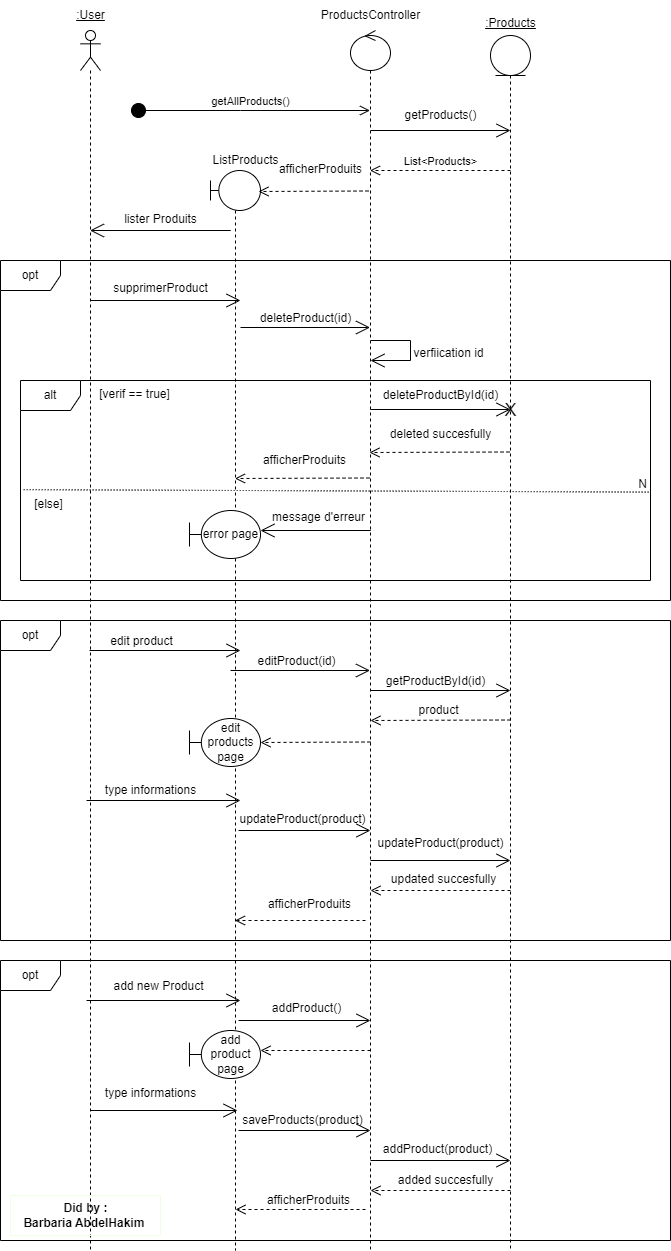


Figure 14: Sequence Diagram "Crud Products"

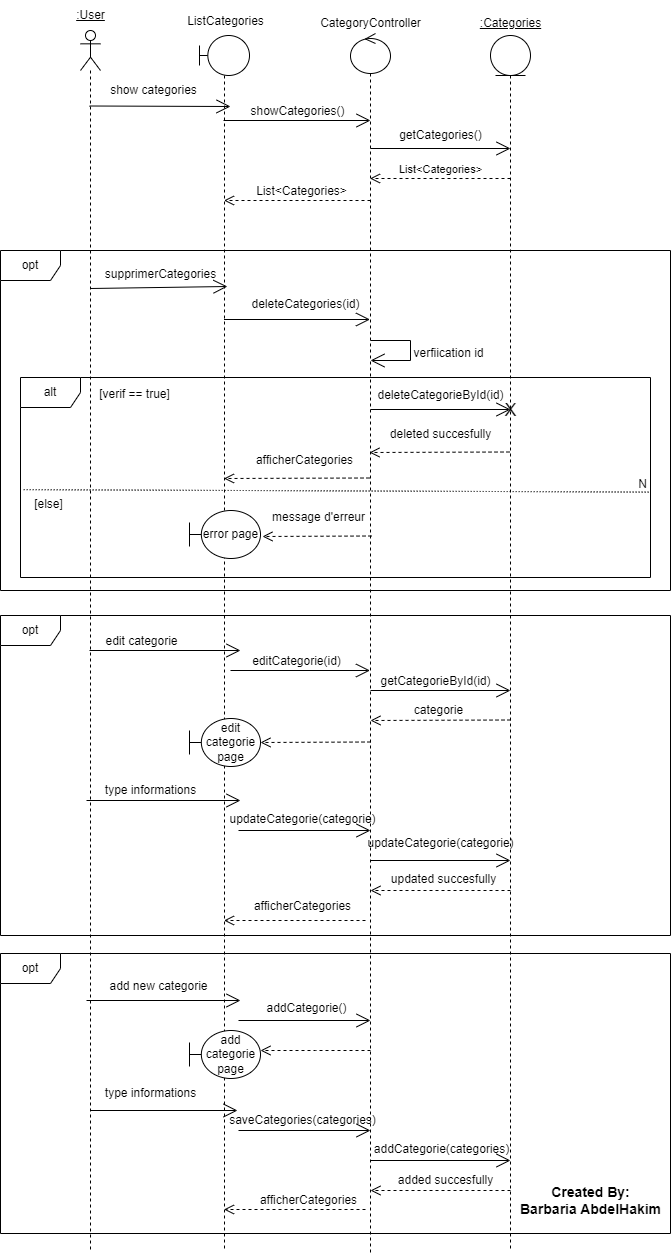


Figure 15: Sequence Diagram "Crud Categories"

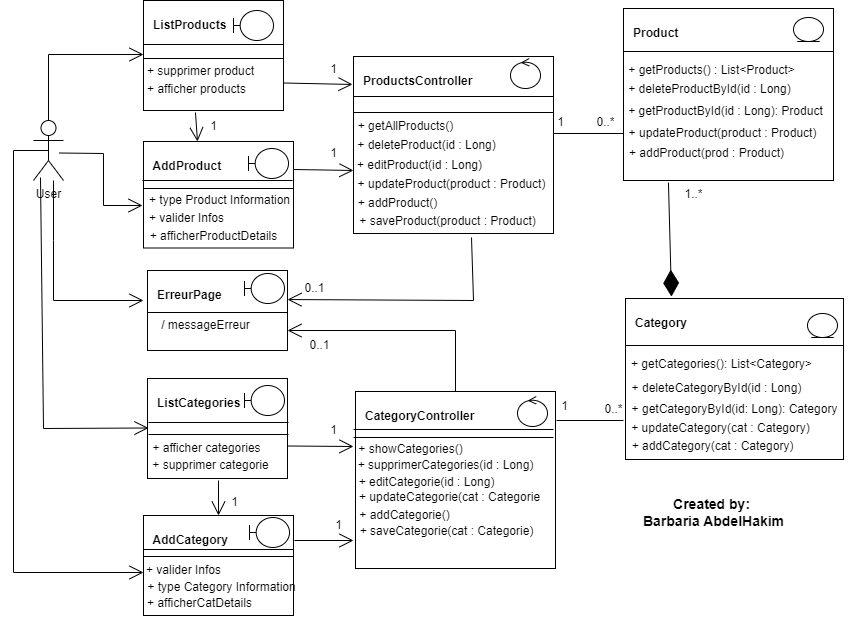


Figure 16: Conception Class

### Interfaces Models

## Sprint retrospective

As a scrum teams, we are planned this meeting to discuss about the problems during this sprint, what’s going well and what we can do to ameliorate.

Table 3: ToDo for amelioration

|  |  |
| --- | --- |
| What’s going well | Doing to ameliorate |
| Technical developpement | Time estimation for tasks |
|  | Affecting tasks to members |

## Conclusion

In this chapter we’ve presented the sprint 1, means we’ve presented the backlog, the sprint objective and the implementations contains the different UML diagram in the MVC structure, and we’ve also presented all the interfaces in mobile or in the web versions.

# Chapter 3: Sprint 2

## Introduction

In this chapter, we are going to present the sprint 2 backlog, the implementation, and the different diagrams and conceptions also the interfaces.

## Sprint goals

# Bibliography

|  |  |
| --- | --- |
| [1] | [En ligne]. Available: https://web-and-mobile-development.medium.com/what-are-the-prime-differences-between-node-js-and-express-js-b560b19b8b33. |
| [2] | [En ligne]. Available: https://www.freecodecamp.org/news/learn-how-to-develop-native-android-apps-with-kotlin-full-tutorial/. |
| [3] | [En ligne]. Available: https://spring.io/trademarks/. |
| [4] | [En ligne]. Available: https://en.m.wikipedia.org/wiki/File:Visual\_Studio\_Code\_1.35\_icon.svg. |
| [5] | [En ligne]. Available: https://techcrunch.com/2017/02/19/why-is-android-studio-still-such-a-gruesome-embarrassment/. |
| [6] | [En ligne]. Available: https://fr.wikipedia.org/wiki/IntelliJ\_IDEA. |
| [7] | [En ligne]. Available: https://www.postman.com/company/press-media/. |
| [8] | [En ligne]. Available: https://1000logos.net/github-logo/. |