# Abstract

My graduation project comprises developing a mobile application for searching and visualizing of natural, bio, vegan, eco, and allergen free products. Based on personal preferences such as skin type, allergies, and personal search history, the app users can search for their desired products.

I developed this project in 2 sprints. First, a mobile application using Flutter, the open-source Google’s UI toolkit, that manage different screen interfaces for the app. And second, I created a back-end web server with NodeJS in order to manage the client search requests and others different functionalities.

Working in this project started from interviewing natural and allergen free real consumers, product manufacturers, product retailers, and consultants to gather information about their right needs in the eco-friendly environment and to optimize their user-experience, then the development and implementation of the requested solution and finished with ambition hopes to optimize this project for better user-experience of the app users and further features.

I analyzed, designed, and developed this project within L’Atelier, the Marketing, training and merchandising division of Kilani Group.

© 2021  
Sadok Laouissi   
ALL RIGHTS RESERVED

# Dedication

Sadok Laouissi

# Acknowledgements

I would like to express my deep gratitude to our corporate supervisor Mrs. Ridha Leila and my faculty supervisor Mr. Ben Ahmed Zied, for their patient guidance, enthusiastic encouragement and useful critiques of this work.

I would like also to extend my thanks to the product consultant of L’Atelier Mrs. Dhahri Olfa for her help in providing the professional information about the skin type, ingredients and allergies and also helping in the interview process.

Finally, I wish to thank ARGANIA IT Support Division for their help during this project.

# Table of Contents

[Abstract i](#_Toc71626096)

[Dedication iii](#_Toc71626097)

[Acknowledgements iv](#_Toc71626098)

[Table of Contents v](#_Toc71626099)

[List of Figures vii](#_Toc71626100)

[List of Tables ix](#_Toc71626101)

[General Introduction 10](#_Toc71626102)

[Chapter I: Situation analysis of the project 13](#_Toc71626103)

[1. Introduction 13](#_Toc71626104)

[2. Presentation of the host organization 13](#_Toc71626105)

[3. Presentation of the organization 13](#_Toc71626106)

[4. Study of the existing 19](#_Toc71626107)

[5. Conclusion 22](#_Toc71626108)

[Chapter II: Specifications and methodology 24](#_Toc71626109)

[1. Introduction 24](#_Toc71626110)

[2. Modeling Language 24](#_Toc71626111)

[3. Software Requirements Specifications 25](#_Toc71626112)

[4. Global Use Case 26](#_Toc71626113)

[5. Software Development Methodology 26](#_Toc71626114)

[6. Project Management 32](#_Toc71626115)

[Chapter III: State of the art 38](#_Toc71626116)

[1. Introduction 38](#_Toc71626117)

[2. Conclusion 38](#_Toc71626118)

[Chapter IV: Sprint Zero - Work Environment 40](#_Toc71626119)

[1. Introduction 40](#_Toc71626120)

[2. Work Environment 40](#_Toc71626121)

[3. Application Architecture 49](#_Toc71626122)

[4. Conclusion 51](#_Toc71626123)

[Conclusion & Perspectives 53](#_Toc71626124)

[Reflection 54](#_Toc71626125)

[Table of Acronyms and Abbreviations 55](#_Toc71626126)

[References 56](#_Toc71626127)

# List of Figures

[Figure 1: Kilani Groupe Logo 16](#_Toc71625987)

[Figure 2: Kilani Groupe Organizational chart 18](#_Toc71625988)

[Figure 3: Kilani Groupe Missions & Activities 19](#_Toc71625989)

[Figure 4: L'Atelier’s Activities 21](#_Toc71625990)

[Figure 5: Yuka Logo 22](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625991)

[Figure 6: INCI Beauty Logo 22](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625992)

[Figure 7: PharmaPocket Logo 23](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625993)

[Figure 8: Think Dirty App Logo 23](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625994)

[Figure 9: UML Diagrams Overview [5] 27](#_Toc71625995)

[Figure 10: Global Use Case 28](#_Toc71625996)

[Figure 11: Agile Methodology workflow 29](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625997)

[Figure 12: Scrum Methodology Overview 30](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625998)

[Figure 13: Scrum Values 31](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71625999)

[Figure 25: Android Studio Logo 43](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626000)

[Figure 26: VS Code Logo 43](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626001)

[Figure 27: Postman Logo 43](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626002)

[Figure 28: WampServer Logo 43](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626003)

[Figure 29: Visual Paradigm Online Logo 44](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626004)

[Figure 30: Adobe XD Logo 44](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626005)

[Figure 31: Adobe Photoshop Logo 44](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626006)

[Figure 32: GitHub Desktop Logo 45](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626007)

[Figure 33: TeamGantt Online Logo 45](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626008)

[Figure 35: Trello Logo 45](#_Toc71626009)

[Figure 36: Java Logo 46](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626010)

[Figure 37: MySQL Logo 46](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626011)

[Figure 38: J2EE Logo 47](#_Toc71626012)

[Figure 39: Apache Camel Dependency 47](#_Toc71626013)

[Figure 40: JClouds Users 48](#_Toc71626014)

[Figure 41: OpenStack4j Users 48](#_Toc71626015)

[Figure 42: Maven Logo 48](#_Toc71626016)

[Figure 43: Spring Logo 49](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626017)

[Figure 44: Hibernate Logo 49](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626018)

[Figure 45: JSON Logo 50](file:///C:\Users\Sadok%20Laouissi\Desktop\PFE%20Documents\PFE-Finale.docx#_Toc71626019)

[Figure 46 : Bootstrap Logo 50](#_Toc71626020)

[Figure 47 : Application's Physical Architecture 51](#_Toc71626021)

[Figure 48 : Design Pattern MVC 52](#_Toc71626022)

# List of Tables

[Table 1: The Scrum Team 30](#_Toc71671621)

[Table 2: Product Backlog 36](#_Toc71671622)

[Table 3: Hardware Environment Characteristics 40](#_Toc71671623)

# General Introduction

Nowadays, many people are thinking about environmental issues and the environmental condition of the Earth. People understand that their irresponsibility hurts the natural environment. Our planet suffers from many problems, which result from excessive human activity.

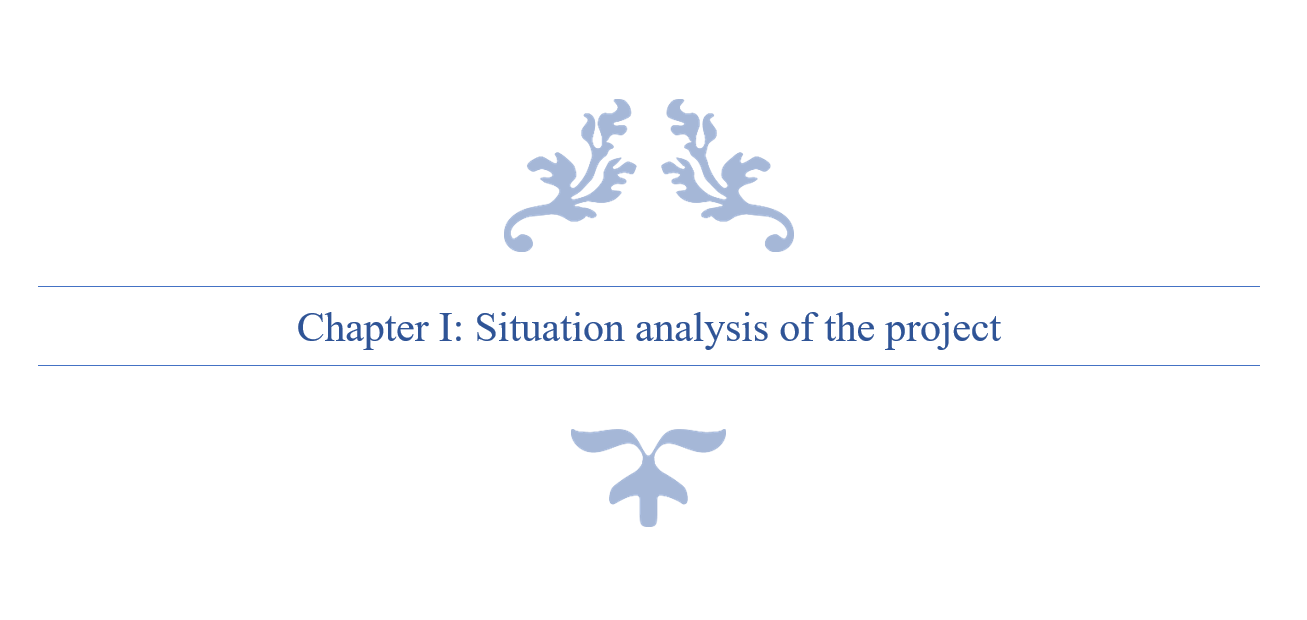
The entire planet is enduring contamination, worldwide warming, deforestation, and imperiling natural life species. These issues are exceptionally pertinent and require quick and comprehensive arrangements. The arrangement for these problems is the alter within the demeanor of humankind towards nature and the natural assets that are being used unrestrictedly. Individuals got to esteem the environment and nature for their survival. Put, individuals ought to “go green” to spare the Soil.

More and more people are “Going Green” in recent years. The primal inspirations for this development have been the diminished presentation to chemicals additionally natural, bio, and allergen-free items are better for the environment. Hence, this new concept has developed the way buyers see the items they use and purchase. Products producers are driving toward a supportability aim and getting to be “Green Manufacturer”. Product Buyers needs more transparency about the ingredients used in making each product and are more willing to advise approximately this rising concept. Thus, taking part in this global movement comes with a part of struggles, the information about allergies, ingredients, allergen... A few customers brag about the product’s quality and harmful ingredients, and others brag about the diversity of products.

The overarching goal of this project is to set up an innovative solution for searching and visualizing of natural, bio, vegan, eco, and allergen free products based on personal preferences such as skin type, allergies, and personal search history.

During the realization of this project, I used a method of fairly efficient development, resulting from the Agile method, namely the SCRUM. I will try through this report to highlight the steps taken, in which I have used the advantages of said method, in particular the plan of productivity and efficiency.

I composed this report of six major chapters. First, I am going to present the situation analysis of the project. Then, the second is about specifications and method. The third chapter is the state-of-the art. The fourth chapter describes the work environment, which is Sprint Zero. And the last two chapters are the sprints implemented to develop this application.



# Chapter I: Situation analysis of the project

## Introduction

In this chapter, I am going to present the host organization first. Then I am going to expose the subject of the project and the work environment. And finally, I will conduct a solution for this problematic.

## Presentation of the host organization

### Introduction

This is a report of the graduation project within the engineer degree majoring in Web & Mobile Software Development at the Private Higher School of Engineering and Technology (ESPRIT). I carried this project out within Kilani Group, the Marketing, training and merchandising department L’Atelier.

## Presentation of the organization

A leading actor in the fields of health, beauty, well-being and hygiene in Tunisia and abroad, Kilani Group is driven by the passion of the profession, the ethics as well as the satisfaction of customers and partners.

Since its creation, Kilani Group has expanded its activities around the world of pharmacy and drugstore including the manufacturing of pharmaceuticals, the import and distribution of healthcare, cosmetics & hygiene brands and medical and paramedical devices. Within a spirit of synergy, a range of service activities have been developed for partners operating in the fields of health & beauty.

The Group is thus well positioned in all sectors: industry, wholesale, direct distribution, retail and services.



Figure 1: Kilani Groupe Logo

### Subsidiaries of Kilani Groupe

Kilani Group started its activities in the distribution sector in 1986. Today, the Group is operating in the import, wholesale and direct distribution. Through dedicated sales teams specialized by sector and a logistic network extended all over Tunisia, the group is specialized in pharmaceuticals, paramedical products, surgical equipment and accessories, reagents and laboratory robots, cosmetics, dermo-cosmetics as well as hygiene products.

* ARGANIA
* KIPROPHA
* PROCHIDIA

Leading actor in Tunisia in the pharmaceutical industry, Kilani Group is a drug manufacturer since 1996 with production sites in Tunisia and abroad. The group innovates by constantly investing in research and cutting-edge technology in order to be at the service of health and participate in the development of the pharmaceutical sector.

* TERIAK

Kilani Group is a leader in retail sales in the beauty sector through a network of self-service outlets throughout Tunisia under its own brand. The Group then expanded into the retail of natural beauty and fashion through franchised brands.

* FATALES
* L’OCCITANE EN PROVENCE
* NINE WEST

KILANI Group also offers a range of complementary services to the medical industry, the distribution and the retail channels. One company is specialized in medical detailing, the management of proprietary and partner brands of drugs and the second company offers consulting in Retail marketing, shops layout; Trade Marketing Activation and Digital Marketing for Retail, entities and brands.

* L'Atelier
* Medicis

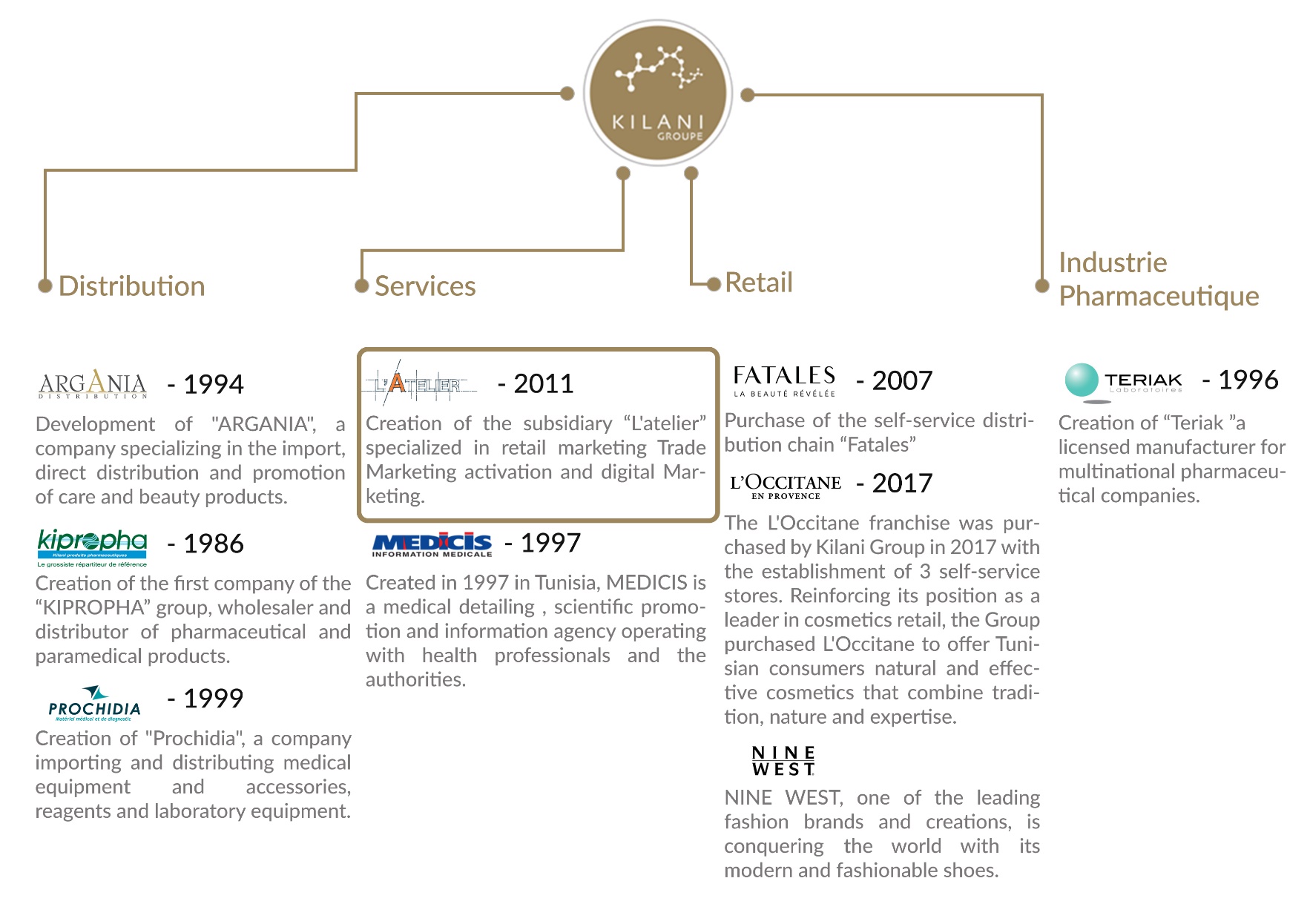


Figure 2: Kilani Groupe Organizational chart

### Missions of Kilani Groupe

Providing well-being and better health to everyone through the development of our brands and those of our partners in drugs, medical devices, care, beauty, hygiene and well-being as well as through our support services for healthcare professionals.

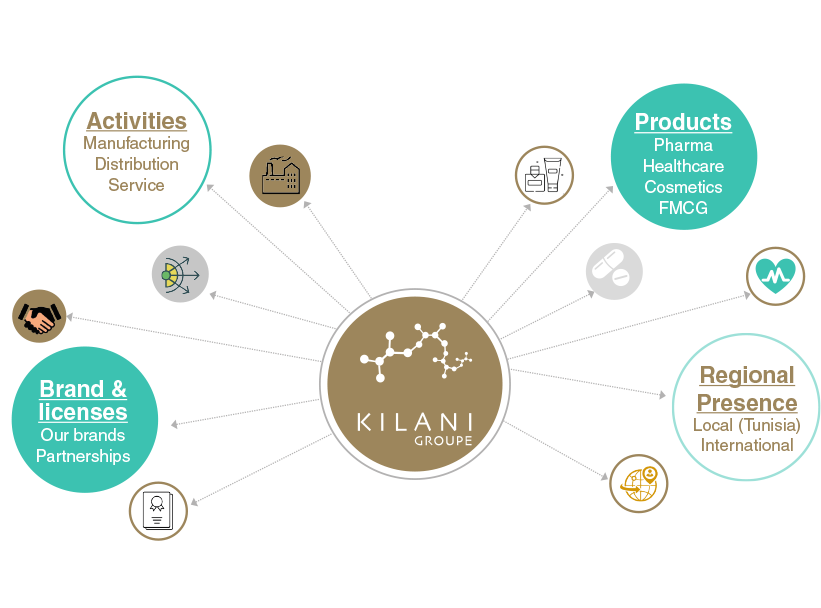


Figure 3: Kilani Groupe Missions & Activities

### Core values of Kilani Groupe

We are driven by a passion, the passion of life. A better life for everyone. Kilani Group has successfully passed on this passion to its subsidiaries and employees who endeavor every day to satisfy all their partners and customers by offering or proposing the best brands, products and services in the fields of health, beauty and well-being.



We adhere to the highest ethical standards. This ethics is that of well performed work and a product that meets the highest quality and safety standards. Ethics that is also based on respect and a sense of responsibility towards our partners, our employees and our customers.

As our main ambition is to satisfy our partners and customers, we do our utmost to innovate through cutting-edge processes to bring out the best in all our fields of activity.



Excellence is our leitmotiv and is embodied through our shared vision with our employees and partners to deliver high quality products and reliable services that meet the requirements of our customers.

### L’Atelier’s Presentation

L'Atelier, a subsidiary of Kilani Group, is specialized in the marketing and communication of brands and outlets.

Composed of a **team of experts**, it supports brands and outlets at all the stages of their projects **from ideation, to launching and marketing activation, in store, out of store as well as digitally.**



Figure 4: L'Atelier’s Activities

L'Atelier's areas of expertise include Marketing Concept Ideation, Outlet Layout and Optimization, Trade Marketing Activation for brands, Marketing and Digital Communication for brands and entities, and Customer Experience Training.

## Study of the existing

### Problematic

After several interviews with customers, producers and retailers in nutriment and cosmetic fields which use, sell, buy or produce natural or bio products, we concluded that currently, there is a strong diversification of organic consumers. Thus, distributors must put in place marketing strategies in order to remain competitive in their market.

The multiple food crises have encouraged the development of the organic market. Indeed, in recent years, demand has grown steadily. Since the early 1990s, the behavior of these consumers has changed dramatically. Combined with a feeling of uncertainty about the quality of the products they buy, they are now turning to certified products, including organic products.

### Presentation of the existing

Based on running tests on the coming concurrent apps, we realized few problems for each one of them.

#### Yuka

Yuka decrypts the labels of your products food and cosmetics and their impact on health.

* Doesn’t have offline mode
* Not suitable for people with special needs such as allergies or diabetes

Figure 5: Yuka Logo

#### INCI Beauty

INCI Beauty is an app that allows you to search and scan a product and consult its ingredients.

Figure 6: INCI Beauty Logo

* Fetching data errors
* Unreliable sources and result not based on scientific background
* Unidentified products
* Complicated to use

#### PharmaPocket

The app allows you to scan the barcodes of cosmetic products and decrypt their composition using simple pictograms.

Figure 7: PharmaPocket Logo

* Last update 2017
* App doesn’t start

#### Think Dirty

Think Dirty app allows us to learn ingredients in our beauty, personal care and household products.

Figure 8: Think Dirty App Logo

* Limited Library
* Exploit the lucrative potential of its proposed service by “Verified Brand”

### Project Solution

Considering this problematic, I propose to develop a mobile application to resolve all of these challenges.

The mission of this project is to provide a responsive mobile app that allows its users to search and filter natural, bio, vegan and allergen-free products with a friendly user interface to inform the users about various information. Then, create a simple and easy form to identify the skin type of the user with a setting page to select its allergies. And last is to implements an algorithm that assist the user in further researches based on his selected preferences.

## Conclusion

My graduation project scope is to develop a mobile application to assist products consumers in their research for the perfect product based on their preferences.



# Chapter II: Specifications and methodology

## Introduction

In this chapter, we’re going to tackle in the first section the modeling language UML, software requirements specifications in the second part, and in the last part, the project management.

## Modeling Language

### Introduction

UML is a standard language for writing software blueprints [4]. UML may be used visualize, construct, and document the artifacts of software. UML is appropriate for modeling systems ranging from enterprise information systems to distributed Web-based applications.

The vocabulary of the UML encompasses three basic terms: **Things**, **Relationships** and **Diagrams**. Things are the abstractions that are firstly build in a model; relationships tie these things together and diagrams group collections of things.

### Unified Modeling Language

A modeling language is a way of expressing building a model, which has been produced during the developing process.

Modeling language defines a collection of model elements. UML, the Unified Modeling language, is the most popular -that will be either by hand or drawing by the tool- needs a number of things that a modeling tool can provide; such as: reporting, integrating with other process model, synchronizations of models and code. Today there are several UML tools on the market that describe the semantics, notations and constructs of UML.

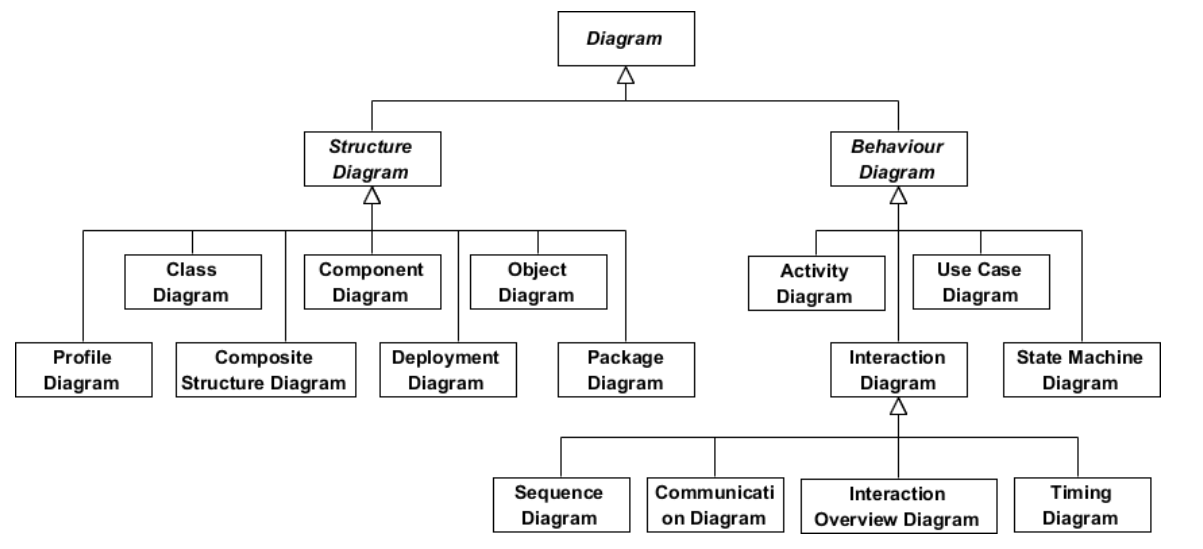


Figure 9: UML Diagrams Overview [5]

## Software Requirements Specifications



### Non-Functional Requirements

Non-functional requirements (NFR’s) cover all the remaining requirements which are not covered by the functional requirements. They specify criteria that judge the operation of a system, rather than specific behaviors.[6]

In other words, NFS define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

* Reliability and synchronization
* Extensibility
* Friendly user experience

### Functional Requirements

The projects functional requirements features are shown as below:

* + Users’ management
  + Preferences and allergies Management
  + Authentication Management
  + Claims Management
  + Management of products and their ingredients
  + Search History Management
  + Product suggestion algorithm
  + Skin Type Deduction Form

## Global Use Case

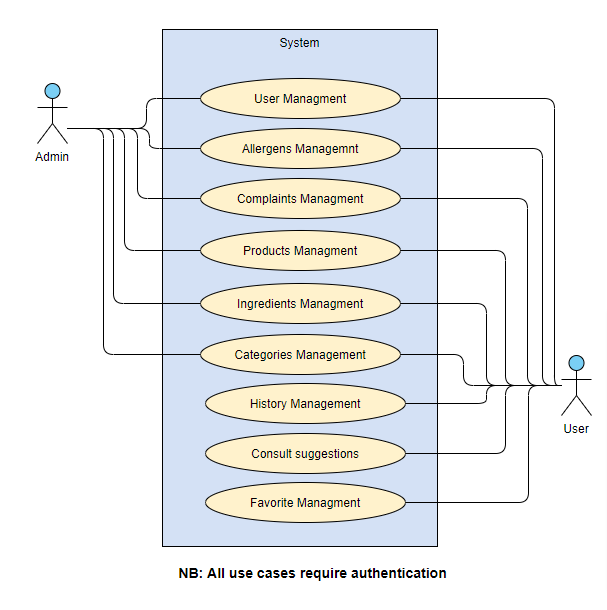


Figure 10: Global Use Case

## Software Development Methodology

**Introduction**

A software development methodology or system development methodology in software engineering is a framework that is used to structure, plan, and control the process of developing an information system.[7]

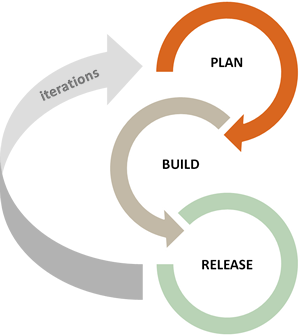
There are the following methodologies:

* Agile Software Development
* Crystal Methods
* Dynamic Systems Development Model (DSDM)
* Extreme Programming (XP)
* Feature Driven Development (FDD)
* Joint Application Development (JAD)
* Lean Development (LD)
* Rapid Application Development (RAD)
* Rational Unified Process (RUP)
* Scrum
* Spiral
* Systems Development Life Cycle (SDLC)
* Waterfall



### Agile Methodology

Agile software development is a conceptual framework for undertaking software engineering projects. There are several agile software development methodologies e.g., Crystal Methods, Dynamic Systems Development Model (DSDM), and Scrum.[8]

We can view agile methods such as Extreme Programming (XP) and Scrum as a reaction to plan-based or traditional methods, which emphasize a "rationalized, engineering-based approach, incorporating extensive planning, codified processes, and rigorous reuse.

In contrast, agile methods address the challenge of an unpredictable world, emphasizing the value competent people and their relationships bring to software development. To clarify the effectiveness of agile methods, we reviewed the agile development literature and conducted a systematic study of what we know empirically about its benefits and limitations.

Figure 11: Agile Methodology workflow



### Scrum Methodology

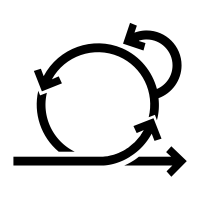
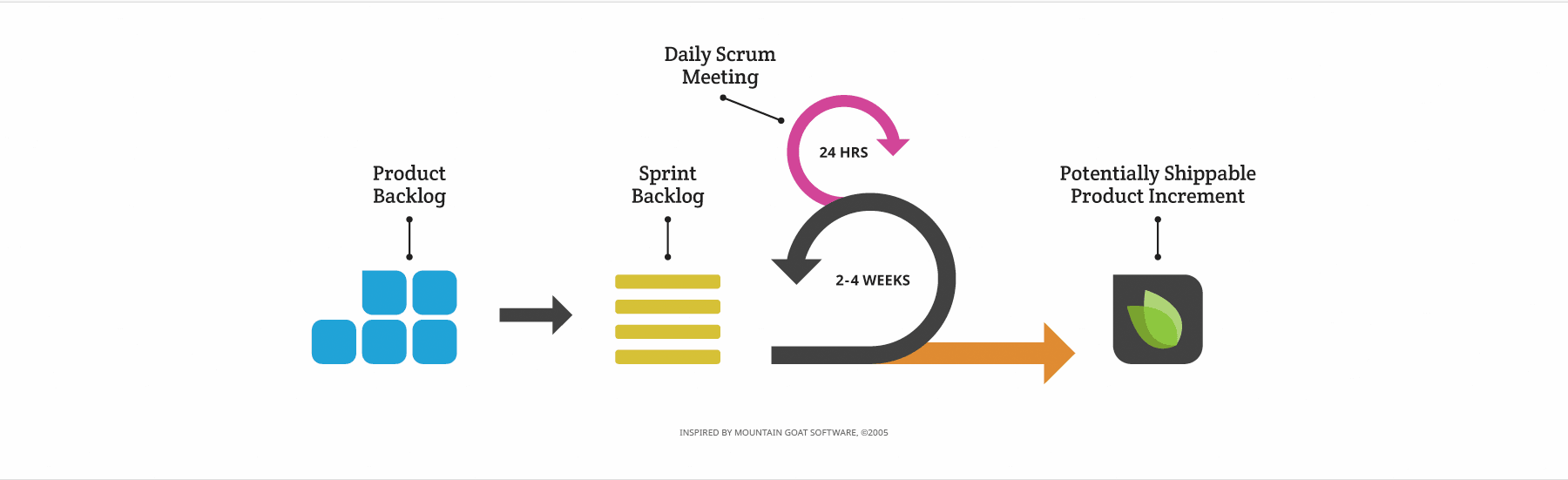
When Jeff Sutherland created the scrum process in 1993, he borrowed the term "scrum" from an analogy put forth in a 1986 study by Takeuchi and Nonaka, published in the Harvard Business Review. In that study, Takeuchi and Nonaka compare high-performing, cross-functional teams to the scrum formation used by Rugby teams. Scrum is the leading agile development methodology, used by Fortune 500 companies around the world.[9]

Figure 12: Scrum Methodology Overview

The Scrum Alliance exists to transform the way we tackle complex projects, bringing the Scrum framework and agile principles beyond software development to the broader world of work.[10]

Figure 13: Scrum Values

#### The Scrum Team

|  |
| --- |
| Product Owner: **Mrs. Ridha Leila** |
| Product owner is the champion for his product. He is focused on understanding business and market requirements, then prioritizing the work to be done by the engineering team accordingly.  Product owner is not a project manager. Product owner is not managing the status of the program. He focusses on ensuring the development team delivers the most value to the business. |
| Scrum Master: **Mr. Ben Ahmed Zied** |
| An effective scrum master deeply understands the work being done by the team and can help the team optimize their delivery flow. As the facilitator-in-chief, they schedule the needed resources (both human and logistical) for sprint planning, stand-up, sprint review, and the sprint retrospective.  Scrum masters also look to resolve impediments and distractions for the development team, insulating them from external disruptions whenever possible. |
| Development Team: **Sadok Laouissi** |
| Development team is the champion for sustainable development practices. The most effective development team is tight-knit and co-located. Team members have differing skill sets, and cross-train each other so no one person becomes a bottleneck in the delivery of work. All members of the team help one another to ensure a successful sprint completion. |

Table 1: The Scrum Team

#### Scrum Events

Prescribed events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum.[11]

* Sprint Planning:

Sprint Planning is time-boxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter.

Sprint Planning answers the following:

Topic One: What can be done this Sprint?

Topic Two: How will the chosen work get done?

* Daily Scrum Meeting

The Daily Scrum is a 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours.

During the meeting, the Development Team members explain:

What did I do yesterday that help the Development Team meet the Sprint Goal?

What will I do today to help the Development Team meet the Sprint Goal?

Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?

* Sprint Review

A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed. Based on that and any changes to the Product Backlog during the Sprint, attendees collaborate on the next things that could be done to optimize value.

* Sprint retrospective

The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning.

The purpose of the Sprint Retrospective is to:

Inspect how the last Sprint went with regards to people, relationships, process, and tools.

Identify and order the major items that went well and potential improvements.

Create a plan for implementing improvements to the way the Scrum Team does its work.

#### Scrum Artifacts

* Product Backlog

The Product Backlog is an ordered list of everything that might be needed in the product and is the single source of requirements for any changes to be made to the product. The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering.

* Sprint Backlog

The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality into a “Done” Increment. The Sprint Backlog makes visible all the work that the Development Team identifies as necessary to meet the Sprint Goal.

## Project Management

### Product Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | As a(n) | I want to | So that I can | Priority |
|  | Administrator | authenticate | access the platform | 17 |
|  | Administrator | Add categories | Add its products | 1 |
|  | Administrator | Edit categories | Maintain the app design & functionalities | 2 |
|  | Administrator | Delete Categories | Restrict categories shown | 3 |
|  | Administrator | View Categories | Chose which to update/delete | 4 |
|  | Administrator | View Complaints | Update complaint state | 39 |
|  | Administrator | Update complaint state | (user) Visualize complaint progress | 40 |
|  | Administrator | View Users | View users global data | 18 |
|  | Administrator | View Allergens | Visualize allergens data | 6 |
|  | Administrator | Add Allergens | Add allergens to products | 7 |
|  | Administrator | Associate allergens to ingredients | (user) Visualize allergens in ingredients | 8 |
|  | Administrator | Edit Allergens | Change allergens data | 9 |
|  | Administrator | Delete Allergens | Disable allergen | 10 |
|  | Administrator | View Products | Visualize Products data | 24 |
|  | Administrator | Delete Products | Delete unwanted Products | 25 |
|  | Administrator | Enable/disable Product | Un/Restrict for users | 26 |
|  | Administrator | Add Product | (user) Gets latest products | 27 |
|  | Administrator | Associate ingredients to products | (user) Gets product’s ingredients | 28 |
|  | Administrator | Add ingredients | Associate ingredients to products | 12 |
|  | Administrator | View ingredients | View ingredients’ data | 13 |
|  | Administrator | Edit ingredients | Edit ingredients’ data | 14 |
|  | Administrator | Delete ingredients | Delete unused ingredients | 15 |
|  | User | Authenticate | Access the mobile App | 19 |
|  | User | Sign up | Create an account | 20 |
|  | User | Validate account | Enable account | 21 |
|  | User | Take diagnosis | Associate to a skin type | 41 |
|  | User | Add personal allergens | Check product’s compatibility | 11 |
|  | User | Update profile information | Edit user’s profile | 22 |
|  | User | Update password | Secure account | 23 |
|  | User | Add search history | View account history | 36 |
|  | User | Delete search history | Remove unwanted search | 37 |
|  | User | View search history | Track account activity | 38 |
|  | User | Add complaint | Adapt user experience | 43 |
|  | User | View complaints | Visualize complaint progress | 44 |
|  | User | Add to favorite | Save preferred products | 34 |
|  | User | Delete from favorite | Remove unwanted products | 35 |
|  | User | View all categories | Get products based on selected category | 5 |
|  | User | View all products types | Get products based on selected types | 31 |
|  | User | View all products | View products data | 30 |
|  | User | Search all products | Find wanted products | 29 |
|  | User | Filter products | Get products based on selected filters | 32 |
|  | User | View product details | View product’s data | 33 |
|  | User | View product ingredients | Visualize product’s ingredients | 16 |
|  | User | View suggestions | Get similar, popular and newest products | 42 |

Table 2: Product Backlog

### Sprint Planning

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint Number | Duration | Start Date | Finish Date |
| Sprint 1 : Categories Management | 15 days | 18/03/2021 | 02/04/2021 |
| Sprint 2 : Allergens Management | 17 days | 05/04/2021 | 22/04/2021 |
| Sprint 3 : Ingredients Management | 21 days | 23/04/2021 | 14/05/2021 |
| Sprint 4 : Users Management | 25 days | 17/05/2021 | 11/06/2021 |
| Sprint 5 : Products Management | 25 days | 14/06/2021 | 09/07/2021 |
| Sprint 6 : Favorite & History Management | 26 days | 12/07/2021 | 09/08/2021 |
| Sprint 7: Complaints & Sugguestions Management | 27 days | 10/08/2021 | 08/09/2021 |

Table 3: Sprints Planning

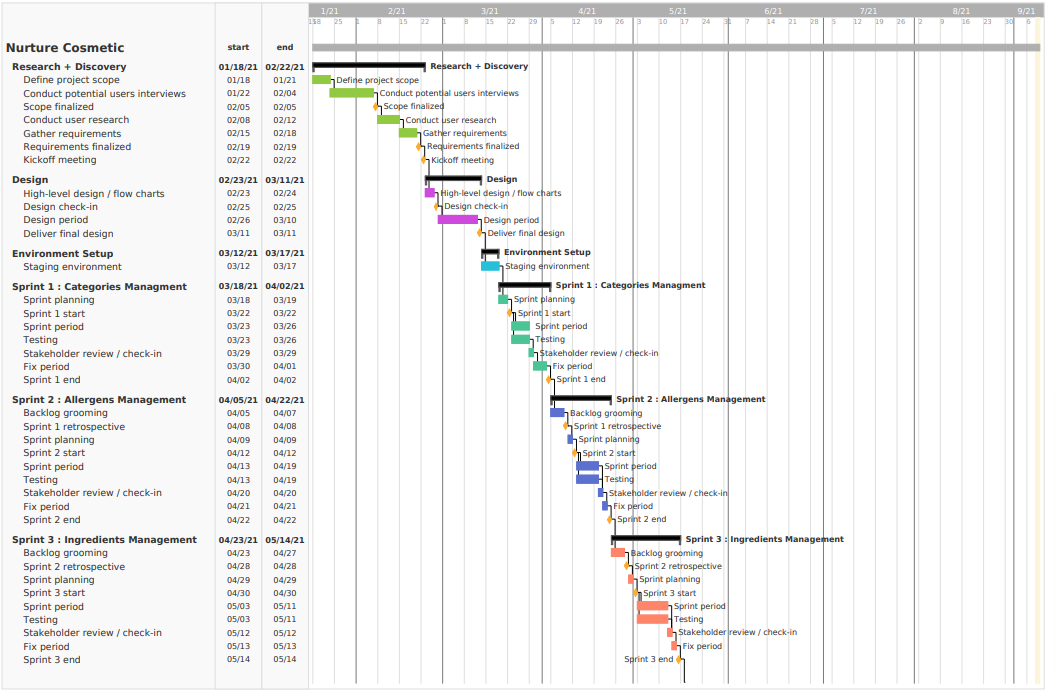


Figure 14 : Gantt Diagram Part 1

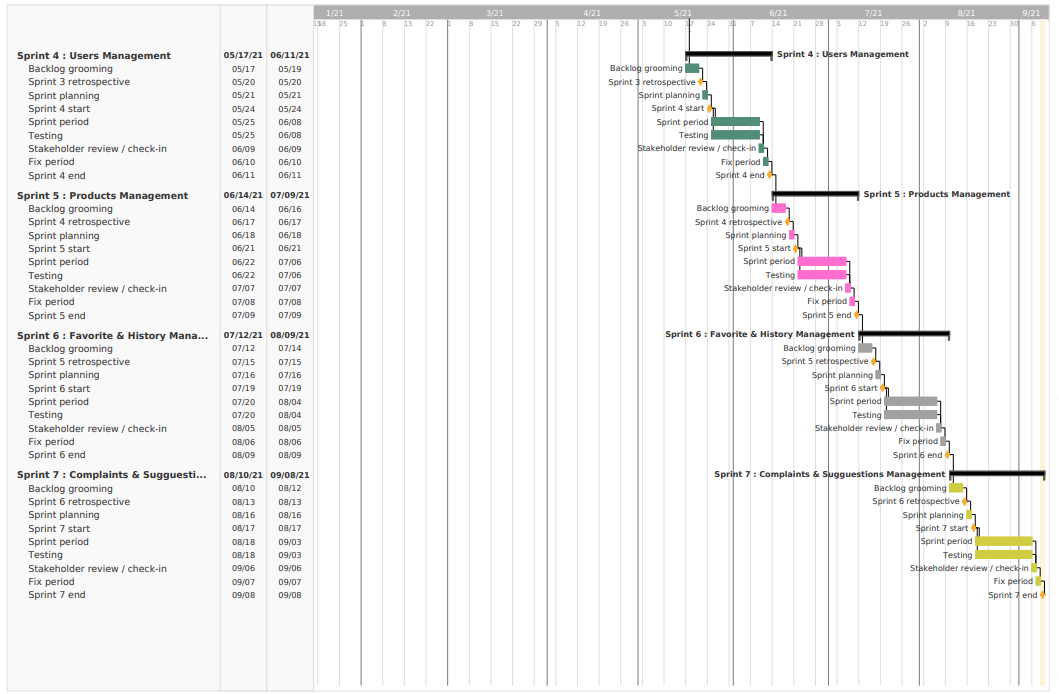


Figure 15 : Gantt Diagram Part 2



# Chapter III: State of the art

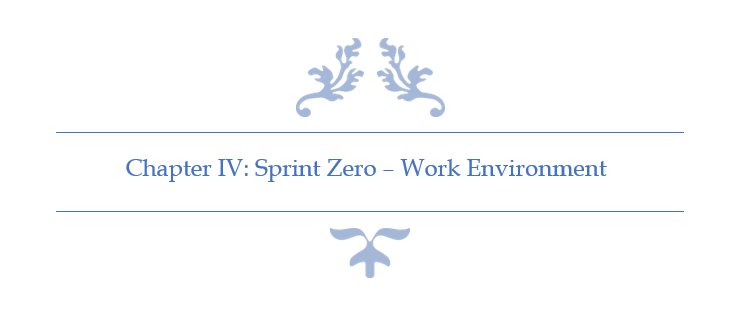
## Introduction

On this part, we are going to explore cloud computing, OpenStack, DevStack, OpenStack Rest API and some implementation features.

## Conclusion

We discovered Cloud computer on this chapter as well as OpenStack, DevStack and the REST API. The Chosen library used in our situation is Openstack4j. It’s well developed, easy to use and contains various services which will be very helpful to understand OpenStack Components.

On the next chapter, we are going to present the work environment, Hardware and software and the technologies used for the implementation.



# Chapter IV: Sprint Zero - Work Environment

## Introduction

In this chapter I am going to present the different tools and technology used. First, I am going to describe the development environment (Hardware and Software) as well as the implementation features. Then, I am going to present the application’s physical and logical architecture.

## Work Environment

### Hardware Environment

Table 4: Hardware Environment Characteristics

To develop the application, we used as hardware environment 2 laptops and 2 desk- computers which have the following characteristic:

|  |  |  |
| --- | --- | --- |
| Hardware | Laptop | Desktop |
| Mark | HP-Pavilion Gaming 15 | Dell |
| Processor | 2.2 GHz Intel Core i7 | 2.5 GHz Intel Core i7 |
| Storage | 250 To SSD & 1 To HDD | 1 To HDD |
| RAM | 20 Go | 16 Go |
| Operating System | Windows 10 | Windows 10 |

### Software Environment

#### Android Studio v4.2

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020.

Figure 16: Android Studio Logo

#### Visual Studio Code v1.55.2

**Visual Studio Code** is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

Figure 17: VS Code Logo

#### Postman v8.3.1

Postman is a collaboration platform for API development. Postman's features simplify each step of building an API and streamline collaboration so we can create better APIs—faster.

Figure 18: Postman Logo

#### WampServer v3.2.0

WampServer refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

Figure 19: WampServer Logo

#### Visual Paradigm Online

Visual Paradigm is a leading and globally recognized provider for Business and IT Transformation software solutions. It enables organizations to improve business and IT agility and foster innovation through popular open standards. Our award-winning products are trusted by over 320,000 users in companies ranging from small business, consultants, to blue chip organizations, universities and government units across the globe.

Figure 20: Visual Paradigm Online Logo

#### Adobe XD

Adobe XD (also known as Adobe Experience Design) is a vector-based user experience design tool for web apps and mobile apps, developed and published by Adobe Inc. It is available for macOS and Windows, although there are versions for iOS and Android to help preview the result of work directly on mobile devices. Adobe XD supports website wireframing and creating click-through prototypes.

Figure 21: Adobe XD Logo

#### Adobe Photoshop

Photoshop is a computer-assisted editing, processing and drawing software, launched in 1990 on MacOS and then in 1992 on Windows. Published by Adobe, it is mainly used for the processing of digital photographs, but also for the ex-nihilo creation of images.

Figure 22: Adobe Photoshop Logo

### Project Management Tools

#### GitHub Desktop

GitHub Desktop is a tool that allows us to interact with GitHub from the desktop. With this new application, one can work easier without having to depend on his browser. GitHub Desktop supports: Attributing commits with collaborators. Checkout branches with pull requests.

Figure 23: GitHub Desktop Logo

#### Teamgantt status reportTeamGantt

TeamGantt is a refreshing take on project planning software that brings Gantt charts online. It allows to plan, schedule, and manage projects with a free Gantt chart software. Best of all, it also allows to invite clients and teams to collaborate on a Gantt chart or project plan.

Figure 24: TeamGantt Online Logo

### Choice of implementation

#### Programming language

##### Dart

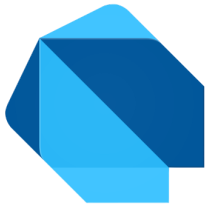
Dart is a programming language designed for client development, such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications.

Figure 25: Dart Logo

Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. Dart can compile to either native code or JavaScript. It supports interfaces, mixins, abstract classes, reified generics, and type inference.

##### TypeScript

TypeScript is a strongly typed programming language which builds on JavaScript giving us better tooling at any scale.

TypeScript may be used to develop JavaScript applications for both client-side and server-side execution (as with Node.js).

Figure 26: TypeScript Logo

#### Database

##### MySQL

MySQL is a freely available open-source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).

Figure 27: MySQL Logo

SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. MySQL is an essential part of almost every open-source PHP application. Good examples for PHP & MySQL-based scripts are WordPress, Joomla, Magento and Drupal.

#### Framework

##### A

J2EE is a set of specifications, which define the standard for developing multi-tier enterprise applications with Java. The J2EE platform provides a complete framework for design, development, assembly, and deployment of Java applications built on multi-tiered distributed application model. J2EE is a platform for building server-side application.



Figure 28: J2EE Logo

##### Apache Camel

Apache Camel is a Java-based, open source integration framework which supports most of the Enterprise Integration Patterns (EIP).

The configuration of Camel is easy. You just specify the “path” where you want to send your data. This path is called the route. You can use a bunch of languages such as XML, Scala DSL, Java DSL etc.

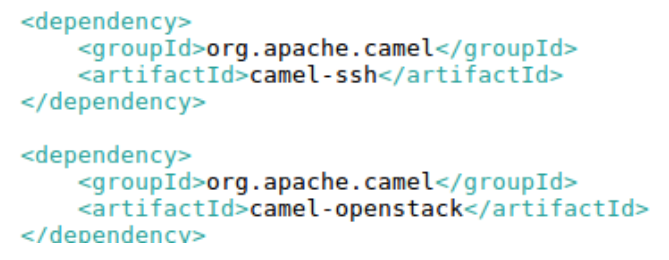


Figure 29: Apache Camel Dependency

##### JClouds

Apache jclouds® is an open source multi-cloud toolkit for the Java platform that gives you the freedom to create applications that are portable across clouds while giving you full control to use cloud-specific features.



Figure 30: JClouds Users

##### OpenStack4j

OpenStack4j is an open source OpenStack client which allows provisioning and control of an OpenStack system. Hosted and sponsored by Rackspace.



Figure 31: OpenStack4j Users

##### Maven

Apache Maven is a software project management and comprehension tool. Based on the concept of a project object model (POM), Maven can manage a project's build, reporting and documentation from a central piece of information.



Figure 32: Maven Logo

##### Spring

The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.

Figure 33: Spring Logo

A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

##### Hibernate

Hibernate is a high-performance Object/Relational persistence and query service, which is licensed under the open source GNU Lesser General Public License (LGPL) and is free to download. Hibernate not only takes care of the mapping from Java classes to database tables (and from Java data types to SQL data types), but also provides data query and retrieval facilities.

Figure 34: Hibernate Logo

#### REST API vs. SOAP

REST operates through a solitary, consistent interface to access named resources. It’s most commonly used when you’re exposing a public API over the Internet. SOAP, on the other hand, exposes components of application logic as services rather than data. Additionally, it operates through different interfaces. To put it simply, REST accesses data while SOAP performs operations through a more standardized set of messaging patterns. Still, in most cases, either REST or SOAP could be used to achieve the same outcome (and both are infinitely scalable), with some differences in how you’d configure it.

#### Data Format

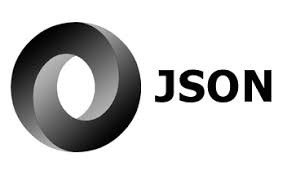
JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

Figure 35: JSON Logo

#### Front-End

##### Bootstrap

Bootstrap is a free front-end framework for faster and easier web development.

Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.

Bootstrap also gives you the ability to easily create responsive designs.

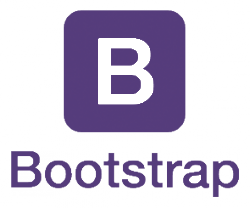


Figure 36 : Bootstrap Logo

## Application Architecture

### Physical Architecture

Our Application is based on 4-tiers Architecture as explained in the following figure. The first part is the Application Dashboard. It’s the application’s interface. The second part is the Server Web which contains the servers web and the application’s back office and contains all the controller. It is also the link between the application’s different parts. The third part is the local database. The last part is the Application Cloud which contains the different services as Keystone, Glance, Nova… The web server communicates with the application cloud through the Openstack4j API.

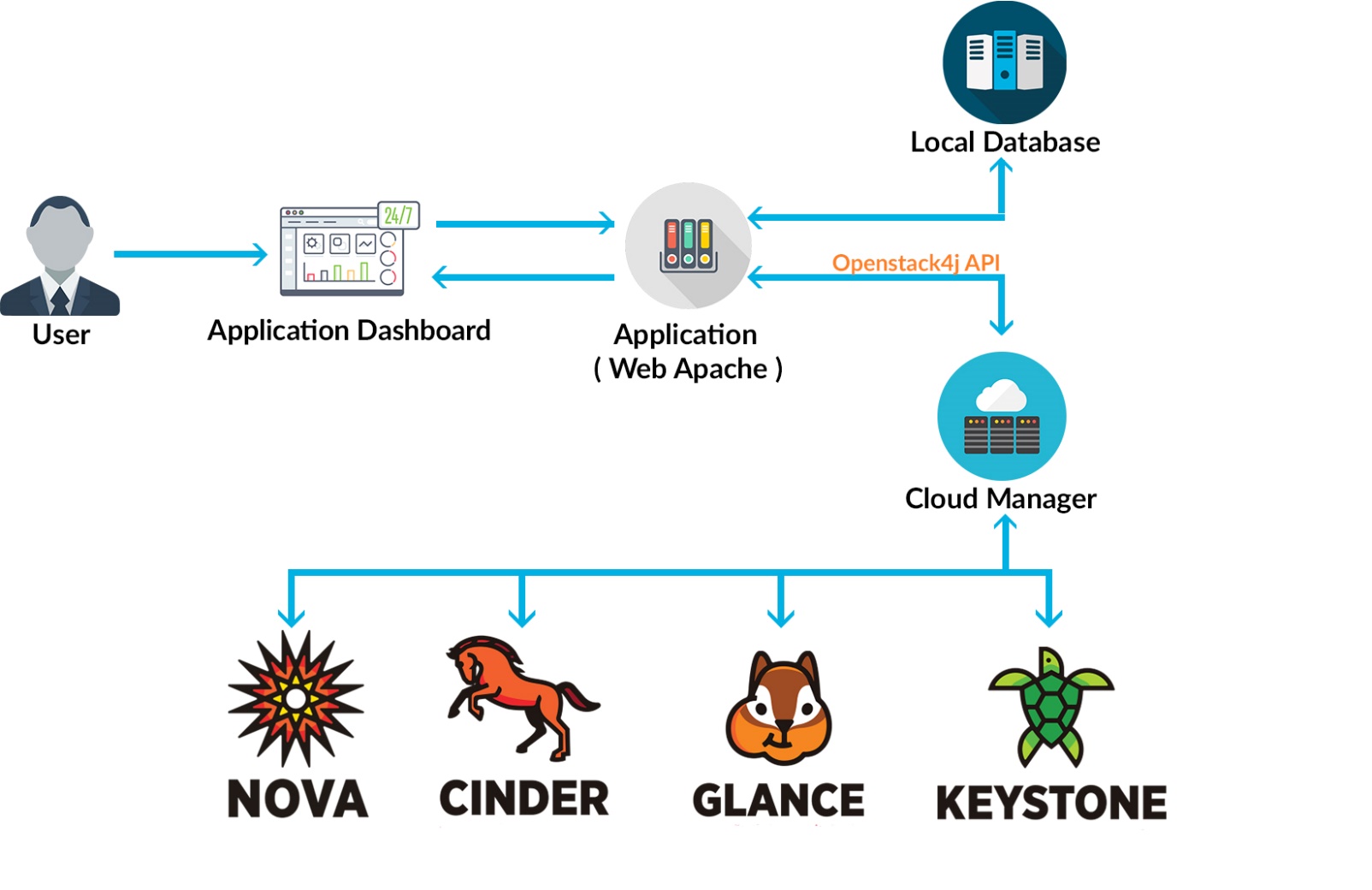


Figure 37 : Application's Physical Architecture

### Logical Architecture

#### Design Pattern MVC

Model–view–controller is commonly used for developing software that divides an application into three interconnected parts. This is done to separate internal representations of information from the ways information is presented to and accepted from the user. The MVC design pattern decouples these major components allowing for efficient code reuse and parallel development.

The following figure explains the MVC Pattern used in our situation.

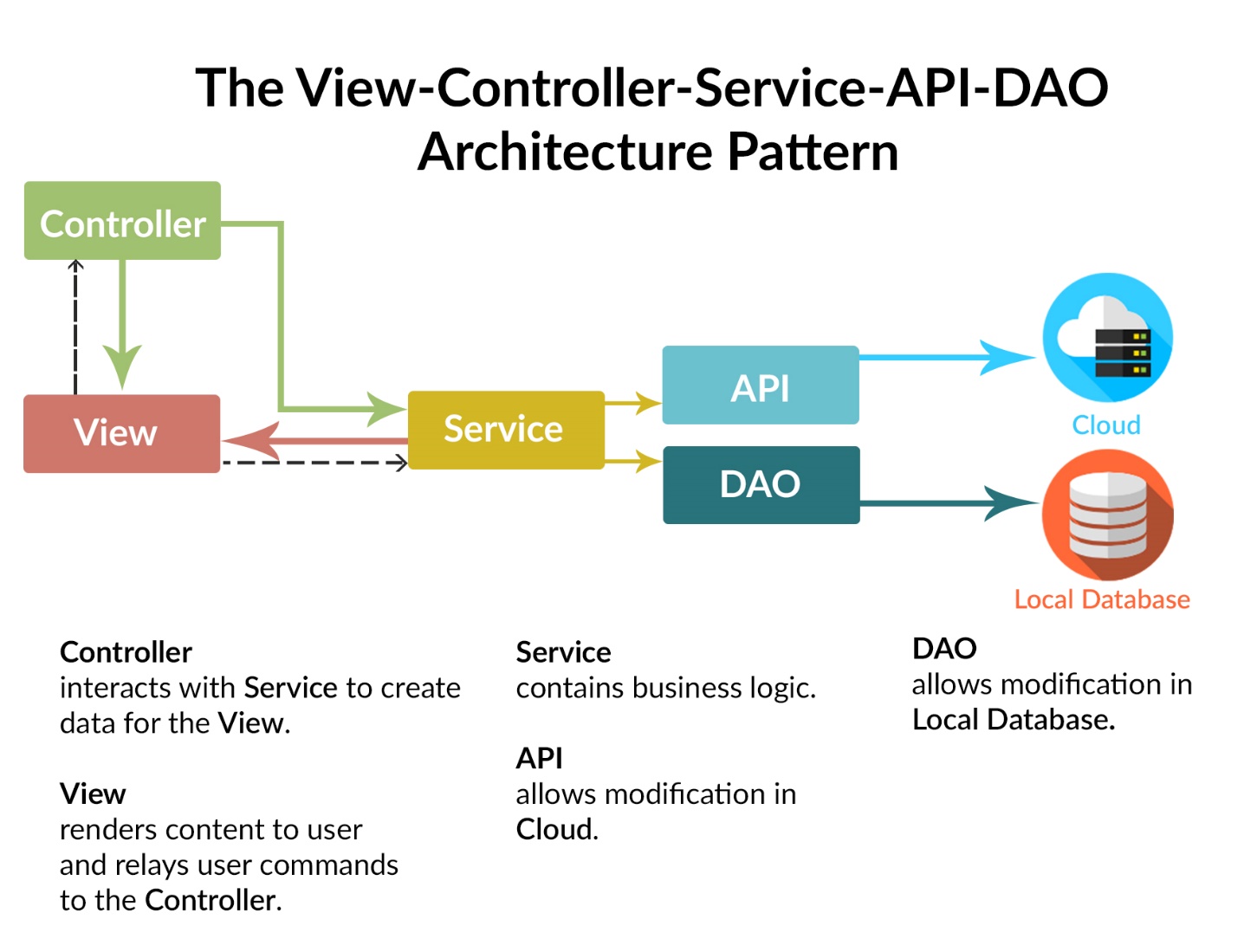
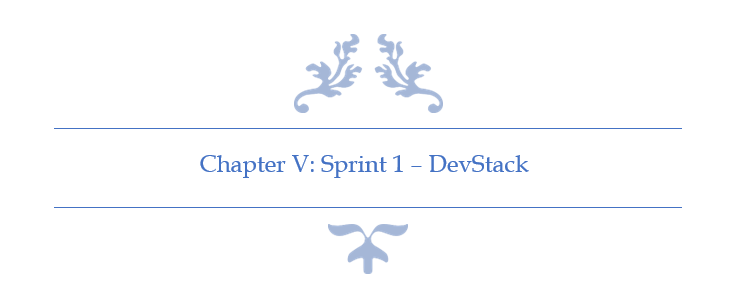


Figure 38 : Design Pattern MVC

## Conclusion

In this chapter, we described the hardware and software platforms on which we built our application. Then, we presented the different technologies used in the realization. Also, we presented the application’s physical and logical architecture.

On the next chapter, we are going to start the development of the first sprint.



# Conclusion & Perspectives

The misuse of IT resources, the lack of control over computer, and especially the time waste during the balancing between different operating systems could all be resolved using cloud computing tools for provisioning and monitoring. That’s the idea behind our project.

To realize this solution, we used the essential framework Openstack4j and Apache Camel to use their APIs to communicate with DevStack that we had already installed using command line. Then we implemented our MySQL database using the Hibernate framework since we’re using JEE. These is overall the technologies that we used to build our web application.

After succeeding to build the app, we’re able to communicate with Keystone to manage users, roles, courses, and tokens, we were able also to communicate with Glance to manage OS images, and finally Compute to communicate with flavors, virtual machines, and floating IPs.[2]

For better results and improvements, implementing the service Magnum would be a good feature. Magnum is responsible for orchestrating an OS image and run it, in either virtual machines or bare metal in a cluster configuration.[24]

In other words, if we have many virtual machines using the same operating system we could use containers to minimize the use of resources by using the same OS to build VMs on it .

# Reflection

When I started this project, I knew nothing about OpenStack or even the cloud computing technology, it was quite challenging to learn about the process and to find the right framework to work with. But here we are now, documenting our work and planning for new features to improve our work.

At the beginning, it was tough to understand the concept of cloud computing and how to implement it in our work. It was the first time to work with these technologies. We spend a lot of time doing researches, information wasn’t easy at all. The concept of provisioning and monitoring cloud is new in Tunisia, and even worldwide, developing it using java wasn’t an easy choice, since python was already implemented in these kind of application, after all OpenStack is written in python.

For sure, we didn’t regret our choice, we feel very mature and more confident after achieving this project, we know we did something very challenging. We believe after this experience, we’re going to dig more and more in the cloud computing and why not build something very big and innovative for Tunisia.

We can now look back and realize that this experience has helped us both as students and as young professionals.[1]

# Table of Acronyms and Abbreviations

|  |  |
| --- | --- |
| API | Application Programming Interface |
| JEE | **Java** Platform, **Enterprise Edition** |
| JSON | **JavaScript Object Notation** |
| IaaS | **Infrastructure as a service** |
| SaaS | **Software as a service** |
| PaaS | **Platform as a service** |
| URI | **Uniform Resource Identifier** |
| VM | **Virtual Machine** |
| DB | **Database** |
| XML | **Extensible Markup Language** |
| UI | **User Interface** |
| REST | **Representational State Transfer** |

# References

|  |  |  |
| --- | --- | --- |
| [1] | https://www.wikihow.com/Write-a-Reflection-Paper | [06/22/2018] |
| [2] | https://writingcenter.unc.edu/tips-and-tools/conclusions/ | [06/22/2018] |
| [3] | http://www.isetr.rnu.tn/ | [05/15/2018] |
| [4] | https://pdfs.semanticscholar.org/5ff8/0e440c6da17edf433c6fd665b85fa93e1f4e.pdf | [05/15/2018] |
| [5] | https://en.wikipedia.org/wiki/Unified\_Modeling\_Language | [05/15/2018] |
| [6] | https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/ | [05/15/2018] |
| [7] | http://www.itinfo.am/eng/software-development-methodologies/ | [05/15/2018] |
| [8] | https://pdfs.semanticscholar.org/5d2a/d96c2dce23e6875dfeeae70a4cf122372457.pdf | [05/15/2018] |
| [9] | <https://www.scrumalliance.org> | [05/15/2018] |
| [10] | https://ieeexplore.ieee.org/document/854065/ | [05/15/2018] |
| [11] | https://www.scrumguides.org/scrum-guide.html | [05/15/2018] |
| [12] | https://aws.amazon.com/what-is-cloud-computing/ | [05/17/2018] |
| [13] | https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/ | [05/20/2018] |
| [14] | https://www.techopedia.com/definition/719/virtualization | [05/20/2018] |
| [15] | https://www.vmware.com/solutions/virtualization.html | [05/20/2018] |
| [16] | https://www.openstack.org/software/ | [05/20/2018] |
| [17] | https://wiki.openstack.org/wiki/DevStack | [05/20/2018] |
| [18] | https://ilearnstack.com/2013/04/26/request-flow-for-provisioning-instance-in-openstack/ | [05/23/2018] |
| [19] | <https://trello.com> | [05/23/2018] |
| [20] | https://searchmicroservices.techtarget.com/definition/RESTful-API | [05/23/2018] |
| [21] | https://jclouds.apache.org/start/what-is-jclouds/ | [05/23/2018] |
| [22] | http://www.openstack4j.com/learn | [05/23/2018] |
| [23] | https://docs.openstack.org/devstack/latest/ | [05/25/2018] |
| [24] | https://wiki.openstack.org/wiki/Magnum | [06/22/2018] |