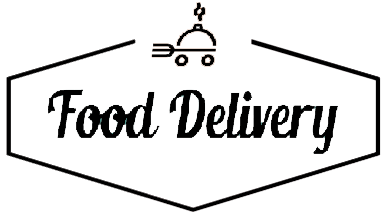
**Project Plan**

***Food delivery app***



#### Version history

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Changes** |
| 1.0 | 15.09.2020 | Nikola Yankov | Basic version |

Contents

[1. Project assignment 4](#_Toc42673512)

[1.1 Description 4](#_Toc42673513)

[1.2 Goal of the project 4](#_Toc42673514)

[1.3 Strategy 4](#_Toc42673516)

[1.4 End products 4](#_Toc42673518)

[1.5 User story and product backlog 4](#_Toc42673518)

[2. Activities and time plan 7](#_Toc42673522)

[3.1 Phases of the project 7](#_Toc42673523)

[3.2 Time plan and milestones 7](#_Toc42673524)

[3. Testing strategy and configuration management 8](#_Toc42673525)

[4.1 Testing strategy 8](#_Toc42673526)

[4.2 Test environment and required resources 8](#_Toc42673527)

[4.3 Configuration management 8](#_Toc42673528)

[4. Finances and Risk 9](#_Toc42673529)

[5.1 Project budget 9](#_Toc42673530)

[5.2 Risk and mitigation 9](#_Toc42673531)

# Project assignment

## Description

As we have a lot of freedom about the topic of the project, and also the requirements are not that specific, I decided to develop a web app for food delivery, which is similar to Deliveroo and Thuisbezorgd.

There will be 3 types of users and therefore 3 interfaces- for the customers, restaurants and admin. The sign-in form will be the same for the customers and restaurants, and the login will be the same for all of them.

In general, the customer interface will display all the restaurants and their menus. There will be an option for the customers to filter the products and to add them to their order.

When it comes to the restaurants, they will be able to add, modify and remove their products depending on their wishes. Some statistics about the sales of their products can be also displayed.

The third interface will be used by the admin of the app to see statistics about all restaurants, customers and maybe orders.

## Goal of the project

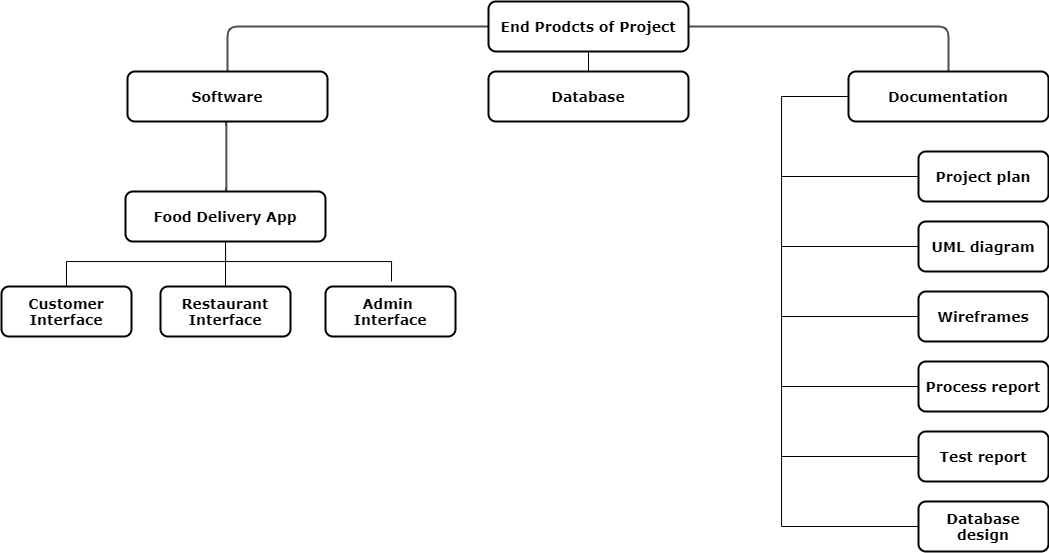
The project is part of my 3rd-year bachelor study program and its main goal is with it to acquire new knowledge, learn new techniques and principles and also to improve my programming skills. One of the biggest advantages of the project is the freedom of choosing a topic. This allows me to choose one which will be useful for my development and also interesting.

## Strategy

In this project, I am going to use scrum as agile management. I chose this methodology because of a few reasons. First of all the sprints which are set in the semester curriculum are with the same duration as the time- boxes used in scrum. Secondly, they allow keeping track of the deadlines and the deliverables in a really easy way, knowing that each period has its own requirements. In addition, reviewing each sprint before moving to the next means that testing is conducted throughout the process, which allows changing the scope or direction of the project at any point. Although the deadline and the main are fixed, the possibility to plan each sprint gives the opportunity to add and change things that were not possible or it was not sure that they can be delivered on time.

## End products

Product Breakdown Structure



## User story and product backlog

# Activities and time plan

## Phases of the project

Although I have decided to use scrum as an agile methodology and have the freedom to change and update my plan for each of the sprints I have a few things that are fixed. I will have 6 sprints including 3 weeks each. Starting with sprint 1 I have to create the first version of the project plan by analyzing the problems and the main goals of the project, setting my environment and get to know the java syntax. Later on, in the next sprint, I have to improve it using new techniques. Just before the end of each sprint, I have to plan the next one with clear goals and deliverables for it and also to review the last sprint. In the final sprint 6, I have to deliver the ready software solution and present it.

## Time plan and milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Phasing** | **Effort** | **Start date** | **Finish date** |
| 1. Sprint 1 | Writting project plan, basic API and CI/CD environment initialization | 31.08.2020 | 18.09.2020 |
| 1. Sprint 2 | Improved project plan, design document version 1 and 1st prototype iteration | 18.09.2020 | 09.10.2020 |
| 1. Sprint 3 | Design document version 2 and 2nd prototype iteration | 09.10.2020 | 06.11.2020 |
| 1. Sprint 4 | Design document version 3, 1st release version, report about OWASP, up- to- date product, sprint backlogs, burn-down charts | 06.11.2020 | 27.11.2020 |
| 1. Sprint 5 | Final design document, 2nd release version, up- to- date product, sprint backlogs, burn-down charts | 27.11.2020 | 16.12.2020 |
| 1. Sprint 6 | Final report, final release, up- to- date product, sprint backlogs, burn-down charts | 16.12.2020 | 22.01.2021 |

# Testing strategy and configuration management

## Testing strategy

For the moment I am planning to test some of the main functionalities in the logic layer without paying much attention to methods that aren’t that complex. Also, I would like to test the main function in the data layer.

This is because I don’t think that is needed to write unit tests for each function when the app is too large and some of them are not that complicated. On the other hand, it is a really good idea and even sometimes mandatory to write some tests when we talk for example for algorithms, security, and ex.

When it comes to quality I will try to test it by using the recommended Sonarqube, although I haven’t work with it until now.

## Test environment and required resources

(Should be extended, because misunderstanding DTAP and picture for visualizing).

I will use GIT in order to better track my progress and the whole development of the project. I don’t think that using CI/CD is needed because I will work alone and conflicts should not appear.

## Configuration management

The version management will be mainly presented by a description of each version and comparison between them and they can also be divided into different branches. When it comes up to change requests, I think that this won’t be a problem because the agile method which I will use (scrum) allows flexible planning and changes.

# Finances and risk

## Project budget

For these project to money will not be needed, but if we think for the finace like receiving something without giving anything I would say that the license for InteliJ IDEA is something like finance help.

## Risk and mitigation

|  |  |  |
| --- | --- | --- |
| **Risk** | **Prevention activities** | **Mitigation activities** |
| 1. Inability to deal with all requirements | Well thought out circumstances and a well structured plan | Restructuring the complexity of the functionalities |
| 1. Non-compliance with deadlines | Well thought out circumstances | Well structured plan |