**Project Plan**

***BookingNL***

*<<Customer>>*

<<

*This template can be used for all projects, especially software engineering projects. Chapters or parts that are not applicable can be removed.*

*Text in italic is background information and must be removed in the final version of your project plan.*

*Note that this is a template and can be changed for own purposes, e.g. you can adapt the layout to the layouts as used at the company of your internship.*

*For your project name, think of a name that highlights the most relevant aspect of your project, and specify whether it is about graduation internship or third year internship.*

>>

|  |
| --- |
| **Date : 09.09.2022** |
| **Version : 1** |
| **State :** |
| **Author : Georgi Zhizgov** |

#### Version history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Changes** | **State** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Distribution**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Receivers** |
|  |  |  |
|  |  |  |

Contents

[1. Project assignment 4](#_Toc42673512)

[1.1 Context 4](#_Toc42673513)

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

[1.3 Scope and preconditions 4](#_Toc42673515)

[1.4 Strategy 4](#_Toc42673516)

[1.5 Research questions 4](#_Toc42673517)

[1.6 End products 4](#_Toc42673518)

[2. Project Organisation 6](#_Toc42673519)

[2.1 Stakeholders and team members 6](#_Toc42673520)

[2.2 Communication 6](#_Toc42673521)

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

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

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

[4. 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)

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

[5.1 Project budget 9](#_Toc42673530)

[5.2 Risk and mitigation 9](#_Toc42673531)

# Project assignment

## Context

*What is BookingNL about? Going on a vacation and looking for a hotel to stay for a good price? BookingNL will present you hotels all around the Netherlands from the cheapest to the most exclusive ones!*

## Goal of the project

I consider the best outcome of this project in the form of a working web application, which can provide the customers whatever they need wherever they need and that they can use the web app securely.

*<<Describe the goal of the project. Take into account:*

*The why, what is the reason for doing this project?*

*What would the new preferred situation look like?*

*What are the advantages of this project?*

*How does this project add value to the company/context?*

*Which possibilities does the ICT product offer that the project will realize?*

*>>*

## Scope and preconditions

*<<What activities and which end products (to what extent or quality) belong to the project, and which don’t.>>*

|  |  |
| --- | --- |
| **Inside scope:** | **Outside scope:** |
|  |  |
|  |  |

*<< Indicate any preconditions. E.g., think of technology choices that have already been made by the company. Note that you are also expected to retain a critical, but constructive, mindset for choices already made >>*

## Strategy

The best aproach in this situation is the Agile methodology, which means constantly working on onn the parts- code part, documentation part, test part and etc. In this way everything will be up to date.

## Research questions and methodology

*<<*

*Describe the research questions that are most relevant to your project. For each research question, describe the approach and/or methodology. Use the Dot Framework to specify strategies and methods - see* [*http://www.ictresearchmethods.nl*](http://www.ictresearchmethods.nl) *for details.*

*Note that research is not only part of the initial phases (like analysis) of the project, but runs throughout the whole project. E.g., in the realization phases, you will probably do research in the Workshop and Lab context.*

*Realize that during the project your research questions may change, and that new ones will come up. That normal for any project, and is not a problem as long as you involve the right stakeholders, and keep your deliverables updated.*

*>>*

## End products

*<< A Product Breakdown Structure (PBS) lists the end products that you realize, including a description of each product. In software engineering, the products are more than just the project plan and the application itself. E.g., requirements documents, architecture documents, research reports and test reports are all end products. These are all important products that are required for effective handover. They are also necessary for further maintenance and follow up-projects. The PBS can change during the course of the project.>>*



# Project organisation

## Stakeholders and team members

*<<Indicate all stakeholders and team members for your project. For each stakeholder indicate the role for your project. Note that the role that a person has for your project is different from the function the person has. E.g., someone with the function “department manager of department X” can have the role of product owner for your project.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Abbreviation** | **Role and functions** | **Availability** |
| *Contact name (and specify further detail as needed, e.g., email or tel nr).* | *Abbreviation can help, e.g., when using the name in tools like Jira or MS project.* | *See above.* | *When is the person available for your project (define this in the way most relevant for your project, e.g., which days are available, the amount of time, or in which phase of the project).* |

## Communication

*<< Indicate the meetings and other channels of communication that you have established, or that you use for your project. Think of communication with all stakeholders including company supervisor, teachers, etc.*

*In which manner does each communication take place? Think of the goals, the location (or whether it should be online), the timing and frequency, and the attendee list>>*

# Activities and time plan

## Phases of the project

*<< Describe the main phases of your project. Even in a scrum project, you should specify at least the components at the beginning and end phases like problem analysis in the beginning, as well as handover, evaluation, reflection, and wrap up at the end.*

*For internship projects, reserve sufficient time for developing/maintaining the portfolio/thesis.*

*>>.*

## Time plan and milestones

*<< For a waterfall project you can indicate the phases and milestones below (can be adapted as required).*

*For an agile project, describe how the artefacts are planned. E.g., length of sprint (with justification), organization of stand up, demo, retrospective.*

*>>*

|  |  |  |  |
| --- | --- | --- | --- |
| **Phasing** | **Effort** | **Start date** | **Finish date** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Testing strategy and configuration management

## 

## Testing strategy

*<<Which testing strategy do you envision? E.g., on which levels will testing take place? Consider that you could choose unit, component, integration, system, or acceptance testing.*

*Justify your strategy, and also set goals where relevant. E.g., percentage code coverage for the relevant unit tests. For each of the planned tests, indicate what will be automated and what not.*

*Also think of quality testing setups like, e.g., Sonarqube.*

*>>*

## Test environment and required resources

*<< Describe the test environment. E.g., do you envision a DTAP (Development, Testing, Acceptance, Production) environment. Can you make use of a CI/CD environment or will you develop your own?*

*It often helps to use a picture to visualize the test environment.*

*If you already know, describe which resources are required for realization and testing. Think of hardware, cloud environments and specific tooling required for development and testing.*

*>>*

## Configuration management

*<< Describe the project approach with respect to version management (e.g. your GIT repository). This might include things like tooling, branching strategy, promotion-, release- and baseline strategy.*

*Also, when relevant, think of a mechanism to deal with change requests and problem reports.>>*