Date: 09.09.2022 Made By: Georgi Zhizgov

Project Plan

BookingNL

Table of Contents

[1. Introduction 2](#_Toc124266003)

[2. Problem Description 2](#_Toc124266004)

[3. Project Goal 2](#_Toc124266005)

[4. Project backlog and Acceptance criteria 3](#_Toc124266006)

[5. RESTful API design 7](#_Toc124266007)

# Introduction

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

# Problem Description

In the Netherlands, there are plenty of applications of any kind, but what is really missing is something that could arrange your accommodation (in hotels or guest houses) while in the Netherlands. BookingNL will let you reserve a place in hotels everywhere you would like to visit.

# Project Goal

The project goal is to deliver an efficient environment (web application) to give you the possibility to reserve a place to sleep(hotels), pay for it or even just take a look at all the possible places. What is more, the managers will be able to visit the web application in order to be in track with the information about the sales, upload new places to accommodate and remove them when they are not available.

# Project backlog and Acceptance criteria

###### Estimation will be done by using the 100% scale

###### Prioritization will be conducted by the MoSCoW table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | As a: | I want to be able to: | So that: | Estimation | Prioritization |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-1  US-2 | Customer,  Admin | Log in | I can see my private information and all the available destinations | 20% | Must |
| Acceptance criteria:   * If log in is successful, the main page is displayed * If log in is not successful, it remains on the log in page * If log in is successful profile button is displayed on the navigation bar | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-3 | Customer  (with no account) | Register  (create an account) | I will be able to look for future holiday destinations and make reservations | 20% | Must |
| Acceptance criteria:   * There will be multiple inputs in order to fill all the information * Product should not be existing in database | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-4 | Customer | See all the available destinations | I will be able to look at all the destinations available and select one to reserve | 20% | Must |
| Acceptance criteria:   * All available holidays will be listed * A holiday will contain a main photo, price, name and location | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-5 | Admin | See the most checked destination | I can see the statistics of what is the most reserved holiday and what is todays income | 40% | Must |
| Acceptance criteria:   * Only visible for managers | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-6 | Customer | Filter information (by price, place and etc.) | I can find faster whatever I am looking for | 20% | Must |
| Acceptance criteria:   * Possibilities for filtering options will be displayed (sort by: location, from cheapest to the most expensive and etc.) * All the holidays will be displayed * If nothing has been found the page print “Nothing found” | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-7 | Customer | Modify my personal information | Keep my profile up to date | 20% | Must |
| Acceptance criteria:   * All the private information will be displayed, such as: first name, last name and etc. * If the data is not in the correct format the modification of the current data will not be allowed(e.g. Deleting your first name, or entering invalid email address) | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-8 | Admin | Delete a destination | If it is not available | 20% | Must |
| Acceptance criteria:   * The product should already exist | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-9 | Admin | Modify a destination | If there is a mistake or an update I can modify it. | 20% | Must |
| Acceptance criteria:   * Only visible for managers | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-10 | Admin | Create a destination | I can upload new destination available for greater variety | 20% | Must |
| Acceptance criteria:   * There will be multiple inputs in order to fill all the information * Product should not be existing in database | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-11 | Customer | Create a reservation | I can reserve a destination whenever I can and whenever is possible | 20% | Must |
| Acceptance criteria:   * There will be inputs for count of guests, start date and end date | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| US-12 | Customer | See personal reservations | I will be able to remind myself when I was on a vacation | 20% | Must |
| Acceptance criteria:   * All the reservations the client has made will appear * If there are no reservations page will be empty saying “No reservations so far” | | | | | |

# RESTful API design

|  |  |  |  |
| --- | --- | --- | --- |
| URL | Resource | Operation | Result |
| /login | Customers | POST | Retrieves a student via email and password |
| /api/register | Customers | POST | Creates a new user |
| /api | Customers | PUT | Updates information |
| /api | Customers | POST | Retrieves the user by getting email parameter |
| /api/token/refresh | Customers | POST | Refreshes access and refresh token |
| /destinations | Destinations | GET | Retrieves all destinations |
| /destinations | Destinations | POST | Creates a new destination |
| /destinations/{id} | Destinations | DELETE | Deletes a destination |
| /destination | Destinations | PUT | Updates destination details |
| /reservations | Reservations | POST | Creates a reservation |
| /reservations | Reservations | GET | Retrieves all reservations for the user by passing an email parameter |
| /chat | Chat | MESSAGE | Sends a message in an open chatroom for everyone who is registered |