|  |  |  |
| --- | --- | --- |
| Project Team |  |  |
| Rozalina Miladinova 3285588  Aleksandar Georgiev 3229742  Nidhi Sharma 3233006  Nguyen Bao Quoc 3477347  Sander Van Bemmel 3261107  Fares Alsalama 3396029 |  |  |



**Airport baggage simulator**

**User Requirements Specification**

|  |
| --- |
|  |

Contents

[Version History 3](#_Toc19465440)

[Introduction 3](#_Toc19465441)

[Functional Requirements 3](#_Toc19465442)

[Non-functional Requirements 4](#_Toc19465443)

[Use Cases 4](#_Toc19465444)

## Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Changes | Author |
| 0.1 | 15/09/2019 | Initial version | Whole team |
| 0.2 | 29/09/2019 | Use cases order changed | Aleksandar Georgiev |
| 0.3 | 06/10/2019 | Removed and Edited some use cases | Aleksandar Georgiev |
| 0.4 | 10/10/2019 | Update extensions and remove 3 use cases and put it in the first use case | Rozalina Miladinova, Nidhi Sharma, Nguyen Bao Quoc,  Aleksandar Georgiev |
| 0.5 | 07/12/2019 | Removed and added some use cases | Nidhi Sharma |
| 0.6 | 19/01/2020 | Final update of document. Added Export statistics use case. | Nidhi Sharma |

## Introduction

The purpose of this document is to give a description of the requirements for the “Airport baggage simulation” software. The document contains a description of all functional and non-functional requirements and explains the user interactions with the system. This document is mainly intended to be provided to the client for approval and a reference for developing the software solution for the simulation software.

## Functional Requirements

|  |  |  |
| --- | --- | --- |
|  | **Functions** | **Which use cases** |
| 1 | Draw the MDA | Use case #1 |
| 2 | Create new check-in | Use case #1 |
| 3 | Create conveyor belt | Use case #1 |
| 4 | Create security check | Use case #1 |
| 5 | Create drop-off cart | Use case #1 |
| 6 | Set conveyor belt speed | Use case #1 |
| 7 | Set conveyor belt length | Use case #1 |
| 8 | Assign number of employees working per drop-off point | Use case #1 |
| 9 | Assign capacity to drop-off point | Use case #1 |
| 10 | Create Flight | Use case #2 |
| 11 | Delete Flight | Use case #3 |
| 12 | Run simulation | Use case #4 |
| 13 | Stop simulation | Use case #5 |
| 14 | Display Statistics | Use case #6 |
| 15 | Save airport map with settings | Use case #7 |
| 16 | Import simulation map with settings from file | Use case #8 |
| 17 | Export statistics | Use case #9 |

## Non-functional Requirements

|  |  |  |
| --- | --- | --- |
| 1 | Robustness | The solution prevents mistakes to happen. |
| 2 | Reliability | The solution should be able to perform even if the load intensifies. |
| 3 | Usability | The solution should be easy to use for the users. |
| 4 | Readability | The solution should contain clearly description of the code. |
| 5 | Extensibility | The solution should be able to easily extend with features. |
| 6. | Testability | The solution supports testing. |
| 7. | Stability | The solution is optimized by the Object-Oriented Software Design methodology. |

## Use Cases

**#1 - Initialize Information**

**Input the initial information needed to start the simulation.**

**Actor:** User

**Pre-condition:**

1. The application has started successfully
2. User chooses to draw the MDA.

**MSS:**

1. User creates the desired number of check-ins
2. User creates the desired number of drop-offs locations
3. User inputs the number of employees working at the drop-off locations
4. User chooses drop-off capacity/ capacities
5. User creates conveyors
6. User creates the desired number of security blocks
7. User proceeds to Use Case 2 to create a flight/flights
8. User sets the conveyor speed.
9. User sets the conveyor length.
10. System validates in the inputted information
11. System enables the button to run the simulation
12. Proceed to Use Case 4

**Extensions:**

1. The simulation is missing any input
   1. Error message is displayed
   2. Go back and re-enter input or cancel

##### **#2 – Create Flight**

**Actor:** User

**Pre-condition:**

1. At least one check-in and one drop-off created

**MSS:**

1. User inputs the flight number, baggage amount, and selects from the drop downs, available check-in and drop-off
2. User clicks Add Flight
3. System checks the input
4. System registers the flight

**Extensions:**

3.The input is not correct

* 1. System displays error message
  2. Go to MSS step 1 or cancel

##### **#3 – Delete Flight**

**Actor:** User

**Pre-condition:**

1. Flight is created

**MSS:**

1. User Select a flight from the list box
2. User presses the Delete button
3. System checks
4. System performs deletion

**Extensions:**

1. If there is no selected flight
   1. System displays message
   2. Go to MSS step 1 or cancel

**#4 - Run Simulation**

**Preforming the simulation from its start to end point.**

**Actor:** User

**Pre-condition:**

1. Initial data is set correctly, and airport map is built

**MSS:**

1. The user chooses the option to start the simulation
2. System begins the simulation
3. System displays the transportation of the baggage

**Extensions:**

2. User decides to stop simulation before all luggage has reached the drop-off desks

* 1. . Go to use case #5

##### **#5 – Stop Simulation**

**Actor:** User

**Pre-condition:**

1. Simulation is running

**MSS:**

1. User presses the Stop button
2. System resets the data
3. Simulation ends

##### **#6 – Visualize Simulation Results**

**Visualization of the output of the simulation depicted into a chart**

**Actor:** User

**Pre-condition:**

1. Simulation has finished successfully

**MSS:**

1. System processes and stores the data from the simulation

2. User clicks the “Show Statistics” tab

3. System displays the result from the simulation in a chart

**Extensions:**

1. User does not want to view results
   1. Use case ends

##### **#7 – Save airport map with settings**

**Save simulation map and settings**

**Actor:** User

**Pre-condition:**

1. Simulation has finished successfully

**MSS:**

1. User clicks on the Save button
2. System opens the save file window
3. User chooses directory and file name
4. System saves simulation map and settings

**Extensions:**

1. User does not want to save simulation settings and map
   1. User presses cancel button
   2. Go to MSS step 1 or cancel

##### **#8 – Import simulation map with settings from file**

**Input the simulation map and settings from a file**

**Actor:** User

**Pre-condition:**

1. Simulation application is started

**MSS:**

1. User clicks on the Load button
2. System opens the file loader window
3. User chooses the desired file
4. System check the content
5. System draws the map and fills the settings from the file

**Extensions:**

1. User provides an invalid file

2.1. System displays error message

2.2 Go to MSS step 1 or cancel

4. The chosen file is corrupted

4.1 System displays an error message.

4.2 Go back to MSS step 3 or cancel

**#9 – Export Statistics**

**After you receive the statistics from the application you can export it.**

**Actor:** User

**Pre-condition:**

1. The application should have been completed

**MSS:**

1. User clicks on the stop button.
2. System loads a dialog box with two options saying, ‘Yes or No’.
3. User clicks on the yes button.
4. System opens a save file window.
5. User chooses directory and gives a name for the file.
6. System saves the statistics in a text file.

**Extensions:**

2. User click on the no button.

2.1. System resets all data and returns to the application.