|  |
| --- |
| Fontys University of applied science |
| User Requirements Specification |
| Parcel Handling Simulation |
|  |
| **GDS - Group 3** |
| **9/1/2010** |

|  |  |  |
| --- | --- | --- |
| Document name |  | User Requirements Specification |
| Project name |  | Parcel Handling Simulation |
| Version |  | 1.0 |
| Department |  | ICT |
| Client |  | Casper Schellekens |
| School tutor |  | Peter Boots |

|  |  |  |
| --- | --- | --- |
| Group leader |  | Qian Li |
| Group members |  | Antoine Girard |
|  |  | John Ibeagha |
|  |  | Kritian Kolev |
|  |  | Sebastien Lepage |

**Approval Signatures**

|  |
| --- |
| Approved by: School tutor  Date: |
| Approved by: Client  Date: |

**Document Change Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Related pages** | **Brief Description of Change** |
| 1.0 | 2010-09-01 | All |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Review and Approval**

|  |  |  |  |
| --- | --- | --- | --- |
| **Review** | | | |
| **Version** | **Date** | **Reviewed by** | **Position/Role** |
| 1.0 |  | Peter Boots | School tutor |
| 1.0 |  | Casper Schellekens | Client |
| 2.0 |  | Peter Boots | School tutor |
| 2.0 |  | Casper Schellekens | Client |

|  |  |  |  |
| --- | --- | --- | --- |
| **Approval** | | | |
| **Version** | **Date** | **Approved by** | **Position/Role** |
|  |  | Peter Boots | School tutor |
|  |  | Casper Schellekens | Client |

**Table of Contents**

[1. Introduction 1](#_Toc271134305)

[1.1 Purpose 1](#_Toc271134306)

[1.2 Cyclic Development of Application 1](#_Toc271134307)

[2. Use Cases 2](#_Toc271134308)

[3. MoSCoW Matrix 10](#_Toc271134309)

# Introduction

## Purpose

This document represents the analysis of functional and user requirements of the Parcel Handling Simulation application that we intend to develop for the client. We will here in state some of the basic requirements that would be made available in the first edition of the application.

## Cyclic Development of Application

The way our team will build this application is through the Iterative Application Development approach. This means that in developing this application, we will have to repeat certain stages (requirements, design, and development) in a cyclic manner.

For the first part of the development (**first cycle**) we plan to get a working application, with some basic functionality that will prove our application is working fine and the architectural backbone is viable. This will mean building the application by following the User Requirements Document.

Next cycle (**cycle no. 2**)

As discussed with the client, we shall take into account the possibility for adding extra functionality e.g. using attributes such as parcel dimension diversity, storage facility, belt speed etc. in our next cycle, which we will note in this document.

As the development of the application continues, we will keep in contact with the client and note any preferences he might have for changes within the application. Depending on the resolutions made we will integrate what has been decided upon.

## Non Functional Requirements

* User Manual, Technical Manual
* English should be used as the language.
* The application should be able to work on the windows OS platform.
* The application would be able to work as a distributed application.

# Use Cases

|  |  |
| --- | --- |
| ID | **1** |
| Name | **New simulation** |
| Goal | To create a new simulation file |
| Pre-condition | The client program is open  The simulation is not running |
| Main  Success  Scenario | 1. The user selects File->New or the “New simulation…” button on the toolbar 2. The system creates a new empty workspace |
| Extensions | 1a. A simulation file is already open and there are unsaved changes  1. The system presents the user with the choices: Save, Don’t Save, Cancel  2. The user selects one of the three and the system responds accordingly:   |  |  | | --- | --- | | Save | * 1. The system goes to "Save" use case   2. The system returns to MSS, step 2 | | Don't save | * 1. The system closes the current simulation, discarding unsaved changes   2. The system returns to MSS, step 2 | | Cancel (or close button) | The system exits the use case | |
| Post-condition | A new empty workspace is created |
| Author | Kristian Kolev |

|  |  |
| --- | --- |
| ID | **2** |
| Name | **Save As…** |
| Goal | To save the currently open simulation under a user-chosen name and location |
| Pre-condition | The client program is open  A simulation is open |
| Main  Success  Scenario | 1. The user selects File->Save As… or the “Save As…” button on the toolbar 2. The system prompts the user to enter the directory and file name under which to save the simulation 3. The user enters the information 4. The system saves the simulation file |
| Extensions | 1a. The file already exists   1. The system prompts the user if they wish to overwrite the existing file 2. The user makes a choice and the system responds accordingly:  |  |  | | --- | --- | | Yes | The system overwrites the existing file | | No | The system goes back to MSS, step 2 | |
| Post-condition | All changes to the simulation file are saved |
| Author | Kristian Kolev |

|  |  |
| --- | --- |
| ID | **3** |
| Name | **Save** |
| Goal | To save the currently open simulation |
| Pre-condition | The client program is open  A simulation is open |
| Main  Success  Scenario | 1. The user selects File->Save or the “Save simulation…” button on the toolbar 2. The system saves all changes to the simulation file |
| Extensions | 1a. The user has not previously saved the current simulation  1. The system goes to the “Save as…” use case, MSS, step 2 |
| Post-condition | All changes to the simulation file are saved |
| Author | Kristian Kolev |
| ID | **4** |
| Name | **Exit** |
| Goal | To exit the client program |
| Pre-condition | The client program is open |
| Main  Success  Scenario | 1. The user selects File->Exit or the Close button 2. The system exits |
| Extensions | 1a. A simulation with unsaved changes is open  1. The system goes to the “Save” use case, MSS, step 2 |
| Post-condition | The client program exits successfully |
| Author | Kristian Kolev |

|  |  |
| --- | --- |
| ID | **5** |
| Name | **Start Simulation** |
| Goal | To start the simulation |
| Pre-condition | The program is open  Simulation mode is launched |
| Main  Success  Scenario | 1. The user click on start button 2. Simulation starts |
| Extensions |  |
| Post-condition | Simulation is running |
| Author | Antoine Girard |

|  |  |
| --- | --- |
| ID | **6** |
| Name | **Pause Simulation** |
| Goal | To pause the simulation |
| Pre-condition | The program is open  Simulation mode is launched  Simulation is currently running |
| Main  Success  Scenario | 1. The user click on pause button 2. Simulation pauses |
| Extensions |  |
| Post-condition | Simulation paused |
| Author | Antoine Girard |

|  |  |
| --- | --- |
| ID | **7** |
| Name | **Stop Simulation** |
| Goal | To stop the simulation |
| Pre-condition | The program is open  Simulation mode is launched  Simulation is currently running or paused |
| Main  Success  Scenario | 1. The user click on stop button 2. Simulation stops |
| Extensions |  |
| Post-condition | Simulation stopped |
| Author | Antoine Girard |

|  |  |
| --- | --- |
| ID | **8** |
| Name | **Add check-in gate** |
| Goal | To add a check-in gate to the editor |
| Pre-condition | The application is in editor state |
| Main  Success  Scenario | 1. The user selects the check-in gate icon 2. User puts the check-in gate icon on the workspace 3. Application adds check-in gate icon to the workspace |
| Extensions | 1a. User doesn’t select the check-in gate icon correctly  Nothing happens in the workspace.  Go to MSS step 1  2a. User puts check-in gate icon outside the workspace  Nothing happens in the workspace.  Go to MSS step 1 |
| Post-condition | The check-in gate has been added |
| Author | Ibeagha Ginika |

|  |  |
| --- | --- |
| ID | **9** |
| Name | **Add check-out gate** |
| Goal | To add a check-out gate to the editor |
| Pre-condition | The application is in editor state |
| Main  Success  Scenario | 1. The user selects the check-out gate icon 2. User puts the check-out gate icon on the workspace 3. Application adds check-out gate icon to the workspace |
| Extensions | 1a. User doesn’t select the check-out gate icon correctly  Nothing happens in the workspace.  Go to MSS step 1  2a. User puts check-out gate icon outside the workspace  Nothing happens in the workspace.  Go to MSS step 1 |
| Post-condition | The check-out gate has been added |
| Author | Ibeagha Ginika |

|  |  |
| --- | --- |
| ID | **10** |
| Name | **Add parcel** |
| Goal | To add a Parcel to the simulation |
| Pre-condition | The application is running in simulation state |
| Main  Success  Scenario | 1. The user selects the Add Parcel icon 2. System gives a form to fill in Parcel destination, ID number, check-in gate and/or priority 3. When form is filled user accepts choices filled 4. Parcel is added |
| Extensions | 2a. User doesn’t fill the parcel properties form correctly  System gives message that parcel properties form should be correctly filled  Go to MSS step 2 |
| Post-condition | A parcel has been added |
| Author | Ibeagha Ginika |

|  |  |
| --- | --- |
| ID | **11** |
| Name | **Remove an item** |
| Goal | In case where the user want to remove an added item in the simulation. |
| Pre-condition | The simulation is not started |
| Main  Success  Scenario | The selected item doesn’t displayed anymore |
| Extensions |  |
| Post-condition |  |
| Author | Sébastien Lepage |

|  |  |
| --- | --- |
| ID | **12** |
| Name | **Clear all items** |
| Goal | In case where the user wants to reinitialize the simulation and make a new simulation with a different number and type of items. |
| Pre-condition | The simulation is not started |
| Main  Success  Scenario | 1. Click on the Edit tab 2. Select “Clear all items”   The number of items is 0. |
| Extensions |  |
| Post-condition | The user has to create a new scenario with new items. |
| Author | Sébastien Lepage |

|  |  |
| --- | --- |
| ID | **13** |
| Name | **Set Speed** |
| Goal | Define the conveyer’s average of speed |
| Pre-condition | Selected item is a conveyer  The conveyer is not running |
| Main  Success  Scenario | 1. Click on the item 2. Display the properties 3. Define the speed |
| Extensions |  |
| Post-condition | The speed is updated for the next simulation. |
| Author | Sébastien Lepage |

# MoSCoW Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functions | M | S | C | W |
| New Simulation |  |  |  |  |
| Save |  |  |  |  |
| Save as |  |  |  |  |
| Exit |  |  |  |  |
| Start Simulation |  |  |  |  |
| Stop Simulation |  |  |  |  |
| Pause Simulation |  |  |  |  |
| Add check-in gate |  |  |  |  |
| Add check-out gate |  |  |  |  |
| Add parcel |  |  |  |  |
| Remove an item |  |  |  |  |
| Clear all items |  |  |  |  |
| Set speed |  |  |  |  |

\* **M** - MUST have this.

\* **S** - SHOULD have this if at all possible.

\* **C** - COULD have this if it does not affect anything else.

\* **W** - WON'T have this time but WOULD like in the future