Requirements and Analysis Document

Legend of Chalmers

Kevin Hoogendijk
Alexander Håkansson
Alexander Karlsson
Maxim Goretskyy

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

Table of content

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

---

1. Introduction

1.1 Purpose of application

The application is for pure entertainment and is targeted at students at Chalmers University of Technology.

1.2 General characteristics of application

Legend of Chalmers is a game where you walk around campus and discover different minigames where you earn hec (higher education credits). The game is a standalone desktop application for Windows, Linux and Mac. It is a top-down styled 2D game where the player walks around in a world and encounters mini-games as s/he explores the world. The mini-games is an interpretation of the different problems students stumble upon during their time at Chalmers. The game is a single-player game but will be created in a way that enables for a future multi-player implementation. The game is real-time and grid based. The goal of the game is for the player to earn enough credits to get their exam (300 hec).

1.3 Scope of application

A 2D game where you walk around Chalmers campus Johanneberg and do mini-games to earn hec. You will not be able to walk inside building or outside of campus. There will only be about 2 minigames implemented but it will be easy to add new ones.

1.4 Objectives and success criteria of the project

The objective is to create a world map that will simulate Chalmer's campus and to be able to walk freely and participate in mini-games. The project will be considered done when it is possible for the player to walk around at least some part of the campus and collect hec. There should be at least 2 minigames.

1.5 Definitions, acronyms and abbreviations

- LoC = Legend of Chalmers
- hec = Higher education credits
- FPS = Frames per second
- NPC = Non-playable character

2. Requirements

2.1 Functional requirements

The game will consist of a map with a player on it. The player should be able to move around on the map and interact with it in various ways. One of these interactions should be to pick up items that are spread out on the map. Through picking up items the player will gather points that progresses the game. When the player has collected 300 points the game will be won. Another way to interact with the map is for the player to talk with NPCs that are spread out on it.

2.2 Non-functional requirements

2.2.1 Usability

The game should be very easy to play with simple controls that remind of other games in the genre.

2.2.2 Reliability

N/A

2.2.3 Performance

The game should run without any noticeable lag - should run with over 30 FPS at all times.

2.2.4 Supportability

The game should be able to run on Mac, Windows and Linux.

2.2.5 Implementation

To achieve platform independency, even over multiple types of devices, the application will use the libGDX framework with Java. Java will have to be installed on the host. The libGDX dependencies will be embedded in the application and won't need to be installed separately by the user.

2.2.6 Packaging and installation

The application and all its dependencies will be packaged into a java package (.jar) and everything that needs to be installed is Java.

2.2.7 Legal

There might be legal issues concerning the music in the game. This will not be covered here.

2.3 Application models

2.3.1 Use case model

See appendix for use case document.

2.3.2 Use cases priority

1. Move player
2. Pick up beverage
3. Dialog with NPC
4. Save game

2.3.3 Domain model

See appendix for the domain model.

2.3.4 User interface

The screen resolution is going to be 1024x576, that is 32 tiles in width, and 18 tiles in height giving a 16:9 aspect ratio. There will be a menu where the player can navigate and perform several actions. Menu is hidden by default. Dialogs with NPCs are shown in the

---

bottom of the screen and they are only shown when there is a conversation going on between the player and a NPC. The points are always shown in the top left corner. See appendix for mockup of the user interface.

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

Appendix

1. Use cases

1.1 Dialog with NPC

1.1.1 Summary

The game allows for interaction with non-playable characters (NPCs) which typically means a conversation. A dialog with an NPC gives information for the player and sometimes allows for making simple choices.

1.1.2 Priority

Medium

1.1.3 Participators

The human player

1.1.4 UC Tables

1.1.4.1 Normal flow of events

#	Player	System
1	Player stands next to NPC and faces in the direction of the NPC.	
2	The player presses SPACE	
3		The NPC turns to face the player.
4		A message from the NPC is shown to the player in the bottom of the screen.
5	Player presses SPACE to continue the dialog.	
6		The message disappears

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

1.2 Move player

1.2.1 Summary

This is how the player navigates in the world.

1.2.2 Priority

Highest

1.2.3 Participators

The human player

1.2.4 UC Tables

1.2.4.1 Normal flow of events

#	Player	System
1	Presses UP, DOWN, LEFT or RIGHT arrow key.	
2		Player moves one tile in the pressed direction

1.2.4.2 Next tile in direction is collidable

#	Player	System
1	Presses UP, DOWN, LEFT or RIGHT arrow key.	
2		Player stays in the same position but faces the direction pressed.

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

1.3 Save game

1.3.1 Summary

The player should be able to save the game's progress so it can be resumed at a later time.

1.3.2 Priority

Low

1.3.3 Participators

Human player

1.3.4 UC Tables

1.3.4.1 Normal flow of events

#	Player	System
1	User presses ESC, P or ENTER key to open menu.	
2		Player interaction is disabled. (i.e. the player can't move or talk to NPCs).
3		A menu is shown in the top right corner and shows menu options in a list format
4	The player selects the save option using UP or DOWN key to navigate the menu.	
5	Player confirms the option using the SPACE key.	
6		Game is saved. Confirmation message is shown and menu stays open.

1.3.4.2 Player is in a non-savable area

#	Player	System
1	User presses ESC, P or ENTER key to open menu.	
2		Game is just paused instead of showing the menu.

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

1.4 Pick up beverage

1.4.1 Summary

Points (higher education credits) can be earned by picking up beverages placed on various locations of the map. These beverages are collected by the player by walking on them.

1.4.2 Priority

High

1.4.3 Extends

Move Player

1.4.4 Participator

The human player

1.4.5 UC Tables

1.4.5.1 Normal flow of events

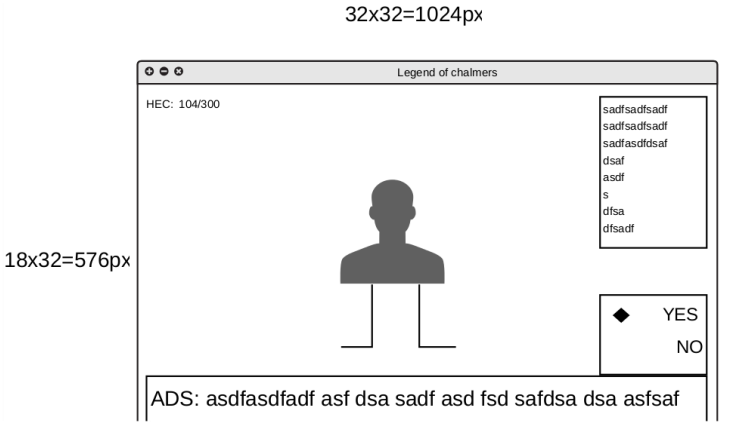
#	Player	System
1	Player moves to a tile which has a beverage on it.	
2		A sound is played.
3		The beverage disappears from the tile.
4		Points are gained.

Version: 2.0
Date: 2/4-15
Author: Kevin Hoogendijk, Alexander Håkansson, Alexander Karlsson, Maxim Goretskyy
This version overrides all previous versions.

---

2. GUI

Mockup of user interface. Also found in “mockup.pdf”.



3. Domain model

Also found in “domain-model.pdf”.