**Software Engineering Project**

**[Interim Project Report]**

|  |  |
| --- | --- |
| **PROJECT TEAM** | |
| **Student No.** | **Student Names** |
| x16138457@student.ncirl.ie | **Lorenzo Cipriani** |
| x16134613@student.ncirl.ie | **Nassima Kara** |
| x16135083@student.ncirl.ie | **Benjamin Adeline** |

**Project : Minecraft**

**7th April 2017**

Table of Contents

[I. Description of the game 3](#_Toc479367687)

[Gameplay 3](#_Toc479367688)

[Main aspects 3](#_Toc479367689)

[Player 3](#_Toc479367690)

[Block 3](#_Toc479367691)

[Mining 4](#_Toc479367692)

[Crafting 4](#_Toc479367693)

[Smelting 4](#_Toc479367694)

[Brewing 4](#_Toc479367695)

[Enchanting 4](#_Toc479367696)

[Mobs 4](#_Toc479367697)

[The Nether 4](#_Toc479367698)

[The End 5](#_Toc479367699)

[Multiplayer 5](#_Toc479367700)

[Realms 5](#_Toc479367701)

[Editions 5](#_Toc479367702)

[II. Project plan 6](#_Toc479367703)

[III. Requirements and Use Cases 7](#_Toc479367704)

[Overview 7](#_Toc479367705)

[Use Case – Manage Health (Lorenzo Cipriani) 7](#_Toc479367706)

[Use Case – Manage Inventory (Lorenzo Cipriani) 8](#_Toc479367707)

[Use Case – Build Block (Nassima Kara) 8](#_Toc479367708)

[Use Case – Destroy Block (Nassima Kara) 9](#_Toc479367709)

[Use Case – Select Mode (Benjamin Adeline) 10](#_Toc479367710)

[Use Case – Select Level (Benjamin Adeline) 10](#_Toc479367711)

[IV. Business Analysis 12](#_Toc479367712)

[V. Design 13](#_Toc479367713)

# Description of the game

Minecraft is a game where a ***player*** can ***move freely*** through a virtual three-dimensional environment. As a sandbox game, the system provides a simulated reality and allow players to develop their character and its behavior in the direction of their choosing. Minecraft focuses on allowing the player to ***explore***, ***interact*** with, and ***modify*** a dynamically-generated ***map*** made of one-cubic-meter-sized ***blocks***. In addition to blocks, the environment features ***plants***, ***mobs***, and ***items***. Some activities in the game include ***mining*** for ore, ***fighting*** hostile mobs, and ***crafting*** new blocks and tools by ***gathering*** various ***resources*** found in the game. The game's ***open-ended*** model allows players to create ***structures***, ***creations*** and ***artwork*** on various ***multi-player*** servers or their own ***single-player*** maps. Other features include ***redstone circuits*** for logic computations and remote actions, ***minecarts*** and ***tracks***, and a mysterious underworld called ***the Nether***. Ultimately, the completely optional goal of the game is to travel to a dimension called ***the End***, and defeat the Ender Dragon.

## Gameplay

The players can choose between the two following game ***modes***:

* ***Safe*** where the players have an unlimited number of blocks without any trials
* ***Survival*** where the players are challenged and should build/destroy blocks to avoid death or attack from creatures in the forest. If the game is played by multiple players, they will need to compete and confront each other to gain ownership the resources.

The ***degree*** of the challenges would vary in relation to the game mode that are split in three levels in the survival mode:

* ***Easy***
* ***Normal***
* ***Hard***

In the case the game mode is ***safe***, there is ***no*** defined specific ***level***.

## Main aspects

Player

The player is the person that the user controls in the world. When the user starts a game, the player is put in a world, ***generated*** by a ***random*** or specified seed, with an ***empty inventory***. If the ***bonus chest*** option is enabled, a chest filled with basic items generates near the player. The player has a ***health bar*** with 10 hearts, and can be ***damaged*** by ***falls***, ***suffocation***, ***drowning***, ***fire***, ***lava***, ***lightning***, ***cacti***, ***falling*** into the Void, and ***being*** ***hit*** by ***hostile mobs***. Damage to health can be ***mitigated by armor***, and health can be ***restored by eating food***, or if difficulty is set to Peaceful, health will regenerate on its own. ***Hunger*** is also a factor if the difficulty is not set to Peaceful, ***depleting over time*** and even faster while ***sprinting***. ***Food*** will replenish the hunger level; however, eating rotten flesh and raw chicken has a chance of giving the player a hunger effect. Depending on the difficulty level, a low hunger level will deplete a players health.  
A player can ***change their skin*** on the ***profile page*** of Minecraft.net.

Block

Blocks are the objects that make up the Minecraft world. There are ***different types*** of blocks; ***natural blocks*** such as ***grass***, ***stone***, and ***ores*** are ***randomly generated*** within the world. There are also blocks that ***players can craft***, such as a crafting ***table*** and a ***furnace***. ***Resources*** can be ***extracted from blocks*** by hand or by ***using tools***. Some of these resources are simply blocks in the player's inventory that ***can be placed*** elsewhere, while others are ***used as material to create*** other blocks or tools. Others yield no practical use whatsoever. ***Some blocks cannot be broken*** through normal survival means; these being Bedrock, End Portal Frames, and Command Blocks.

Mining

Mining is one of the main aspects of Minecraft. Mining is done to ***extract ore and other materials*** from below the surface of the map. These ores include ***coal***, ***iron***, ***gold***, ***redstone***, ***diamond***, ***lapis lazuli***, and ***emerald***. These are crucial in ***making*** several useful ***items***. Mining can involve ***digging*** a hole from the surface or going down through a ***cave***. Abandoned ***mineshafts*** create extra areas to look for resources.

Crafting

Crafting allows players to ***create*** new ***tools*** and ***blocks*** using items from their inventory. To craft, a player can use the ***2×2 grid*** in the ***inventory*** or the ***3×3 grid*** provided by a ***crafting table***.

Smelting

Smelting requires a ***furnace*** in addition to ***fuel***, and ***processes*** blocks such as iron ore into a more useful form (e.g. iron ingot).

Brewing

Brewing ***creates potions*** from various ingredients and water using a Brewing Stand. They are stored in a glass bottle and then ***consumed by the player or thrown*** at other mobs to generate a certain effect based on the ingredients used to create the potion.

Enchanting

Enchanting is also used to ***upgrade armor, tools, or weapons*** with an enchanting table. More powerful enchantments can be accessed by ***gaining experience*** and ***placing bookshelves*** around the enchanting table.

Mobs

Mobs (short for Mobiles) are the ***animals*** and other ***creatures*** that ***inhabit the map***. These include ***zombies*** that ***attack*** by ***melee*** and ***summon*** other zombies; ***skeletons*** that have a ***bow*** and ***arrow***, ***spiders*** that ***jump*** large distances and can ***climb*** walls, and ***creepers*** that ***explode*** when near the player. ***Rare mobs*** include spider jockeys, which is a skeleton riding a spider, chicken jockeys, which is a baby zombie riding a chicken, endermen, which are tall, black creatures with purple eyes and turn aggressive when the player looks at them, and slimes, which spawn deep within the map and in swamplands. ***Withers*** can be ***built*** in all dimensions. To ***aid*** the player there are several ***passive mobs***: ***pigs***, ***cows***, ***chickens***, ***sheep***, ***rabbits*** and ***squid***, and three ***tamable mob*** types, ***wolves***, ***ocelots***, and ***horses***. Wolves will ***attack*** enemy mobs if the ***player engages*** or is attacked by them. Cats, or tamed ocelots will keep creepers at bay and can't take fall damage. Passive mobs other than wolves ***yield resources when killed***, such as ***beef***, ***porkchops***, ***chicken***, ***wool***, ***leather***, and ***ink sacs***. If killed when ***on fire***, the meat drops are changed to ***steaks***, ***cooked porkchops*** and ***cooked chicken***.  
There is currently only one ***NPC mob***: ***villagers***, which spawn and move about within their village.

The Nether

The Nether is a ***dimension*** in Minecraft, ***accessible*** from the Overworld by a ***Nether*** ***Portal***. It consists mainly of ***Netherrack*** and generates ***lakes of lava***. It is populated by ***Zombie*** ***Pigmen*** (modified pigmen with a golden sword that are neutral), ***Blazes*** (which shoot fireballs and fly), ***Ghasts*** (flying mobs that spit exploding fire balls and attack without provocation), ***Wither Skeletons***, and ***Magma Cubes*** (similar to slimes but jump a bit higher).

The End

The End is another ***dimension*** of the game where the player battles the ***Ender Dragon***. The End is ***accessible*** by entering an ***End Portal*** in a Stronghold. The End is composed of End Stone and is ***inhabited*** by ***Endermen***. It also contains ***Obsidian Pillars*** and ***Ender Crystals*** that ***heal*** the Ender Dragon.

Multiplayer

Minecraft multiplayer servers have developed to include ***their own rules and customs***, guided by their ***administrators*** and ***moderators***. New ***user-created features*** have shown up in Minecraft. This includes features like ***money***, ***vehicles***, ***protection***, ***RPG elements*** and more. With the default controls, the ***chat screen*** is brought up by pressing T.

Realms

Minecraft Realms is an official ***subscription***-based server ***hosting*** service that allows players to ***create*** and ***manage*** their own ***private*** Minecraft ***servers***. Realms are ***not*** intended for ***large*** ***public*** servers, but for ***groups of friends*** or as a family server.

## Editions

The game is distributed in different editions:

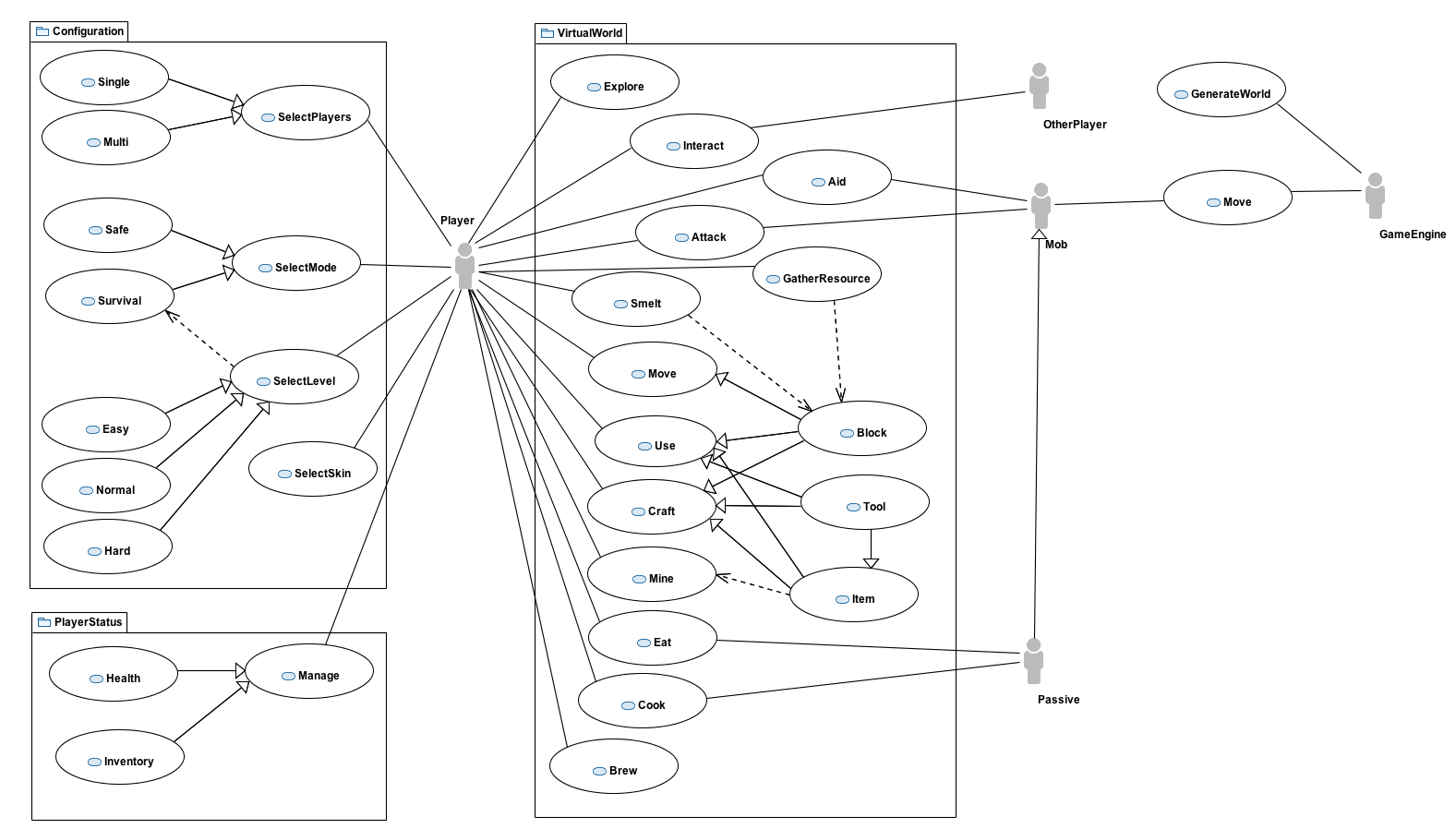
* Computer (PC, Mac, Linux)
* Pocket (Android, iOS, FireOS, Windows 10, Gear VR, Apple TV, Fire TV)
* Console (Xbox, PlayStation, Wii)
* Education (Mac, Windows 10)

# Project plan

The tool we used to create the GANTT is raising errors while trying to export the plan.  
We will try to add it later as an external file.

# Requirements and Use Cases

## Overview



## Use Case – Manage Inventory (Lorenzo Cipriani)

**Scope**

A player needs to check its inventory of items (tools, resources, materials) in order to decide what following actions it can do in the game.

**Description**

Exploring the virtual world allows the player to collect items. Some blocks contain resources. Materials can be collected with the mining actions. Items can be crafted, etc. All these items that are available to the player are listed into the inventory, where they can be selected in order to be used.

**Flow Description**

***Precondition***

For selecting and item, this must be present in the inventory.

***Activation***

The player clicks on the Inventory icon.

***Main flow***

1. The inventory list is displayed on the screen.
2. The player select an item

***Subflow***

1. The player scroll the list of items if it’s longer than the height of the view

***Alternate flow***

a)

1. The inventory list is displayed on the screen.
2. The player doesn’t select any item and close the inventory

b)

1. The inventory list is displayed on the screen.
2. The player puts into the inventory the collected item.

***Exceptional flow***

None

***Termination***

The player select an item from the inventory or clicks on the Close icon.

***Post condition***

The inventory list is removed from the screen.  
If the player selected an item then this can be used in the virtual world.

## Use Case – Build Block (Nassima Kara)

**Scope**

The scope of this use case is to craft blocks individually during the game in safe or survival mode

**Description**

This use case allows the player to build/craft individual blocks

**Flow Description**

***Precondition***

1. The player has authenticated himself under a specific name or as a guest

2. The player has started the game

***Activation***

This use case starts when it is the player turn either at the beginning of the game or during the game in a single or multiplayer mode.

***Main flow***

1. The player moves to the area where the block should be built

2. The player can browse the type of available materials from the tool menu (Ore block, Cloth block, Clay, Brick, Glass, Sand etc…)

3. The player selects the material according to the purpose of the action (build house, gates, stairs etc...) [S1]

4. The player targets the area where the block must be built

5. The player clicks on the target area and builds the block

***Subflow***

[S1] When the material is selected, the tool to build the block shows up automatically and replaces the previous tool that was used the previous use case

***Alternate flow***

1. No blocks can be built if there are no materials

2. No block can be built if there are no tools

***Exceptional flow***

- None

***Termination***

The system stores the actions of the player

***Post condition***

The system goes to a wait state for the next interaction with the player

## Use Case – Destroy Block (Nassima Kara)

**Scope**

The scope of this use case is to destroy blocks individually during the game in safe or survival mode

**Description**

This use case allows the player to destroy a block with the help of available tools

**Flow Description**

***Precondition***

1. The player has authenticated himself under a specific name or as a guest

2. The player has started the game

***Activation***

This use case starts when it is the player turn either at the beginning of the game or during the game in a single or multiplayer mode.

***Main flow***

1. The player moves next the block to be destroyed

2. The player clicks on the Tools button

3. The player can browse the list available options used to destroy a block.

4. The player selects the tool that he wishes to use

5. The player selects the block to be destroyed

6. The player clicks on the block and destroys it

***Alternate flow***

1. No block can be built if there are no tools

***Exceptional flow***

- None

***Termination***

The system stores the actions of the player

***Post condition***

The system goes to a wait state for the next interaction with the player

## Use Case – Select Mode (Benjamin Adeline)

**Scope**

The scope of this use case is for a player to choose his mode game (Safe or Survival).

**Description**

This use case describes the selection of the mode from the player.

**Flow Description**

***Precondition***

The system is idle.

***Activation***

This use case starts when a player has clicked on Play.

***Main flow***

Player select mode

***Alternate flow***

1. The Player can choose Safe mode.

2. The Player can choose Survival mode

***Exceptional flow***

No exceptional flow

***Termination***

The game mode is selected

***Post condition***

The system registered the choice and start to load the mode selected.

## Use Case – Select Level (Benjamin Adeline)

**Scope**

The scope of this use case is for a player to select survival level

**Description**

This use case describes selection of a level in survival mode from a player.

**Flow Description**

***Precondition***

The system is idle.

***Activation***

This use case starts when a customer has selected a survival mode.

***Main flow***

Player select Survival Level

***Alternate flow***

1. The player can select “Easy”

2. The player can select “Medium”

3. The player can select “Hard”

***Exceptional flow***

***Termination***

The use case finishes when the player has selected the level

***Post condition***

Now that the level is selected, the player can choose his map

# Business Analysis

# Design