Chapter 4

External specification

4.1 Hardware and software requirements

The game supports building under Windows, so Windows 7 or higher is recommended. The game was tested on hardware with a configuration of:

- **Processor:** Intel Core i5 13600k.
- Motherboard: Gigabyte Z690 Gaming X DDR4.
- **RAM**: 48 GB DDR4.
- Graphics Card: AMD Radeon RX 6700 XT.

On which no performance issues were noticed and the framerate was locked at 120 fps.

4.2 Installation procedure

4.2.1 Pre-built executable file

If the game has been supplied in an already pre-built form, all that is required is to run the MakeFarmApp.exe executable file.

4.2.2 Building the project using CLion 2022.2

To build a project, the user will need a CMake version of at least 3.16, an internet connection, and an MSVC compiler that supports the C++20 standard.

1. Start CLion and select the option to open a new project (see Figure 4.1).

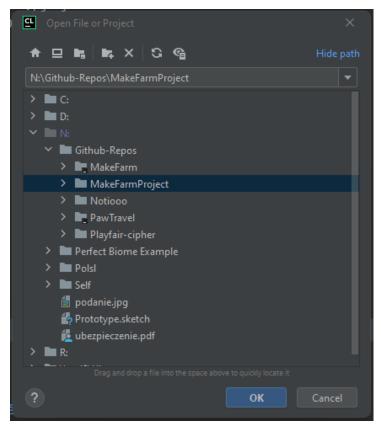


Figure 4.1: Selection of a new project in the "Open File or Project" tab.

2. There is a CMakeLists.txt in the main MakeFarm folder, which the user has to right-click on and select "Load CMake Project" (see Figure 4.2).

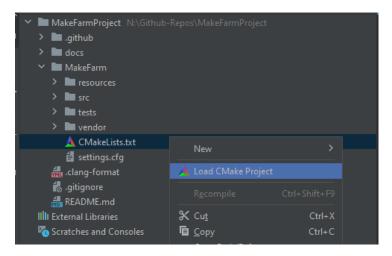


Figure 4.2: Loading the MakeFarm project from the main CMakeLists file.

[Finished]

3. Downloading external libraries and building them will begin (see Figure 4.3).

```
"C:\Program Files\JetBrains\CLion 2022.2.4\bin\cmake\win\bin\cmake.exe" ...
-- Fetching SFML...
-- SFML Fetched!
-- Fetching ImGui...
-- ImGui Fetched!
-- Fetching ImGui-SFML...
-- Found ImGui v1.89 WIP in N:/Github-Repos/...
-- ImGui-SFML Fetched!
-- Fetching GLM...
-- GLM: Version 0.9.9.9
-- GLM Fetched!
-- Fetching FastNoiseLite...
-- FastNoiseLite Fetched!
```

Figure 4.3: Sample output displayed in the "CMake" tab when the project is being loaded.

4. The build should end with the line: [Finished] (see Figure 4.4).

```
...
-- Configuring done
-- Generating done
-- Build files have been written to: N:/Github-Repos/...
```

Figure 4.4: The final output in the "CMake" tab showing the correct loading of the project.

5. Then select *MakeFarmApp* from the list of available configurations and click "Run Make-FarmApp" (see Figure 4.5).

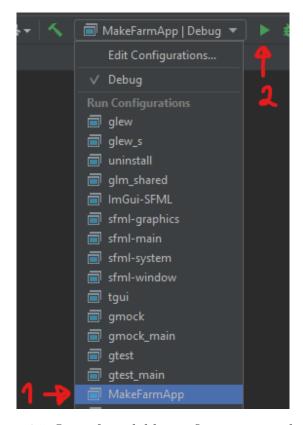


Figure 4.5: List of available configurations to build.

6. Once the compilation is complete, the game window should launch. If there any problems occurred please check in the CLion settings that under *Build*, *Execution*, *Deployment* > *CMake* the Toolchain is set to *Visual Studio*.

4.3 User manual

4.3.1 How to start

To start the game, simply run the .exe executable file. The game window with the main menu should appear on the screen (see Figure 4.6).



Figure 4.6: The main game menu.

From this level, it is possible to exit the game by clicking "Quit Game". Selection, or creation of a world is possible by clicking "Singleplayer" (see Figure 4.7).

4.3.2 Loading world

If the game already has worlds created, they will be visible in this view. In Figure 4.7 there can already be seen the two created worlds "Custom named world" and "New world". In this case, all that is needed is to click on an existing world and the game will start.

4.3.3 Creating new world

When creating a new world, all that have to be done is to click "Create New World". If the world is not named in any way, it will simply be called "New world" and any subsequent worlds created will be called "New world (1)", "New world (2)" and so on. In Figure 4.7, the newly created world would be called "My new world name" thanks to the name entered in the field to the left of the button.



Figure 4.7: View of world selection after clicking on the "Singleplayer" option.

4.3.4 Control basics

To move around the game world use the **W**, **A**, **S**, **D** keys, which allow to move in the four directions of the world. To jump press the **spacebar**. The spacebar also allows to swim upwards when the player is in the water. In the game, player can look around using the mouse.

4.3.5 Hotbar

Hotbar (see Figure 4.8) is a quick access bar around which there is a lot of additional information. The hotbar contains 9 slots for storing items. The currently selected slot is marked with a white and gold border. Each item has its own texture, and in the upper left corner of the slot appears in what quantity the item is in the slot. In the middle above the hotbar displays the name of the currently selected item.

Health

Above the left side of the hotbar the current health status can be seen. If all the hearts are light-red then the player has the maximum amount of life, if all the light-red hearts disappear and are replaced by dark-red hearts then the player dies. One example of possible loss of life is a fall from a great height onto a hard surface. Water in this case cushions the fall.

Oxygen

In addition, when the player is underwater, the current oxygen level is displayed above the hearts. If all the bubbles disappear and the player is still underwater then the player will start to lose life. Emerging from the water renews the oxygen level to the maximum possible level.



Figure 4.8: An in-game hotbar that contains items for quick access.

4.3.6 Destroying blocks

To destroy a block, all it takes is pointing the crosshair at the block to be destroyed and then pressing the **left mouse button**. Depending on block properties, the destroyed

block may or may not appear in the inventory. An example of this operation can be seen in Figures 4.9 and Figure 4.10. It is worth noting that in these Figures can be seen that after destroying the block, the number of blocks in the inventory increased by one.



Cobblestone

Figure 4.9: Screenshot of the game before the block was destroyed.

Figure 4.10: Screenshot of the game after the block is destroyed.

4.3.7 Placing blocks

To place a block, the player must first have a block in the inventory (see 4.3.5 and 4.3.8). Then use the mouse scroll to select the block in the hotbar that should be placed (see 4.3.5). To place a block, simply aim with the crosshair inside the game at the place where the block should be placed. Then simply click the **right mouse button**. The block will appear adjacent to the indicated block wall. See the Figure 4.11 before the block is placed and the Figure 4.12 after the block is placed. It is worth noting that on the figures (see 4.12 and 4.11), the amount of blocks in the inventory decreased by one after placing the block. Each time a block is placed, it always requires having a block and then losing one quantity of it from the inventory.



Figure 4.11: Screenshot of the game before the block was placed.



Figure 4.12: Screenshot of the game after the block is placed.

4.3.8 Inventory

The full inventory can be found under the $\bf E$ key. While the hotbar only shows 9 slots of space, the entire inventory allows to store 36 different items. The equipment is shown in Figure 4.13.

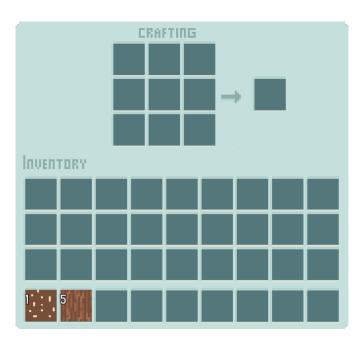


Figure 4.13: In-game equipment.

Moving around the inventory

To take and move the blocks with the mouse, press the **left mouse button**. To put away all the blocks being held by mouse, click with the left mouse button on a free slot. If player left-clicks on an item of the same type as the one being held by the mouse, their quantities will add up. Use the **right mouse button** to insert only one item at a time into a slot. Example operations can be seen in Figure 4.14.

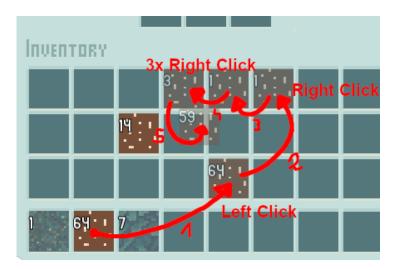


Figure 4.14: Examples of operations performed in the inventory.

Items and quantities

In most cases, blocks and items can be stored in a maximum of 64 units. This means that one slot can usually only hold 64 blocks, or other items. Some items nevertheless have their own limitations and sometimes a slot can hold fewer items than 64. An attempt to combine two items with a total quantity greater than 64 is shown in Figures 4.15 and 4.16



Figure 4.15: Player attempts to combine two blocks with a total of more than 64.



Figure 4.16: As a result of merging, 64 blocks are inside the slot and the mouse holds the remaining 14 that did not fit.

4.3.9 Crafting

Crafting can be found directly in the inventory under the **E** button. Crafting makes possible that using different types of items with the right arrangement in crafting-zone (see Figure 4.17) results in obtaining another item in different quantity.

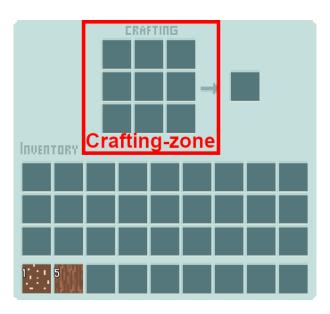


Figure 4.17: Crafting-zone as part of the inventory.

An example of the crafting process can be seen in figures 4.18 and 4.19.

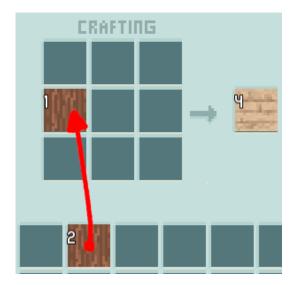


Figure 4.18: The player inserts one block of wood into the crafting-zone, which is the correct arrangement for creating the planks. The planks appear as the proposed crafting result.

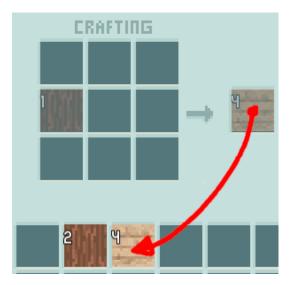


Figure 4.19: The player takes away the result of crafting. When the resulting crafting item is caught with the mouse, the items in crafting-zone responsible for creating the new item disappear.

4.3.10 Death and respawn

In the event of death, the game screen turns red and the player has two options: save the game and return to the main menu or respawn (see Figure 4.20). In the case of a respawn, the player appears at the starting point from where they started and the player's inventory is cleared. Of course, by selecting the options to return to the main menu and returning to the world later, the player will be in the same situation.

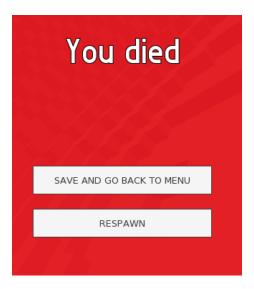


Figure 4.20: Death screen displayed in the event of a player's death.

4.3.11 Pause

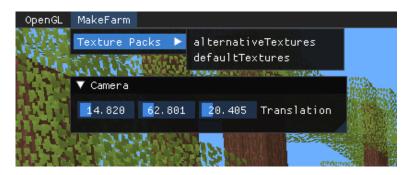
To pause the game, simply click the **ESC** (Escape) button. The game in the background stops and the screen becomes darker and two buttons appear in front of the player: save and exit the game, and save and return to the main menu. The pause window can be seen in Figure 4.21.



Figure 4.21: Pause screen.

4.3.12 Developer options

When the **F4** button is pressed, additional developer options are activated. These are the player's current position, additional drop-down menu (see Figure 4.22), and the frame rate in the top right corner (see Figure 4.23). In addition, the **F1** button allows the cursor to be detached from the game, so that the developer interface can be used. The **F2** button takes half heart of life away from the player. The **F3** button allows the player to fly, where in addition the **spacebar** is responsible for flying upwards and the left shift for descending.



118.51 FPS

Figure 4.22: Additional developer options.

Figure 4.23: Frame rate counter.

Texture Packs

By choosing from $MakeFarm > Texture\ Packs$, textures can be changed at runtime. An example of the effect of changing from default to alternative textures can be seen in Figure 4.24.



Figure 4.24: Switching from default textures to alternative textures.

Wireframe

By selecting $OpenGL > Switch \ Wireframe$, it is possible to switch the view to the default or wireframe at runtime. An example of such a switch can be seen in Figure 4.25.

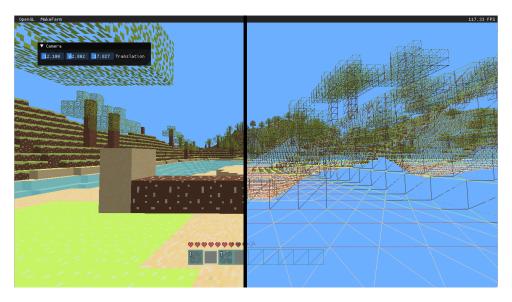


Figure 4.25: Switching on and off the wireframe mode.

4.4 Working scenarios

Most of the functionality has already been shown in the Section 4.3, so mostly the scenery of the game world will be presented here.

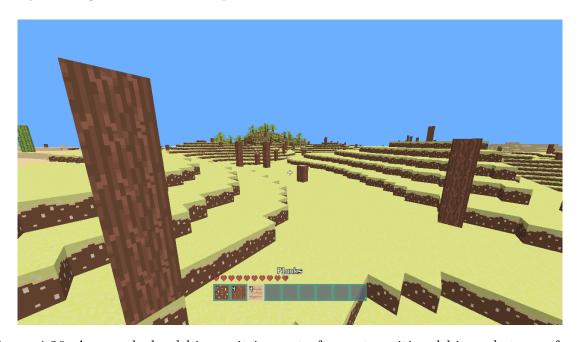


Figure 4.26: A very dryland biome, it is most often a transitional biome between forest and desert. Dead trees are present. Its terrain is very flat.

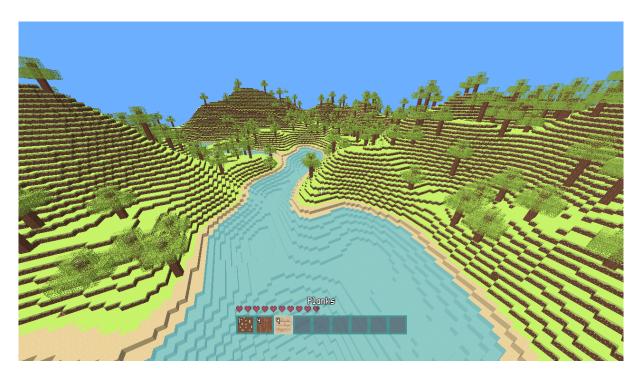


Figure 4.27: A forested biome, its terrain has quite a few hills and lakes. Trees are present.

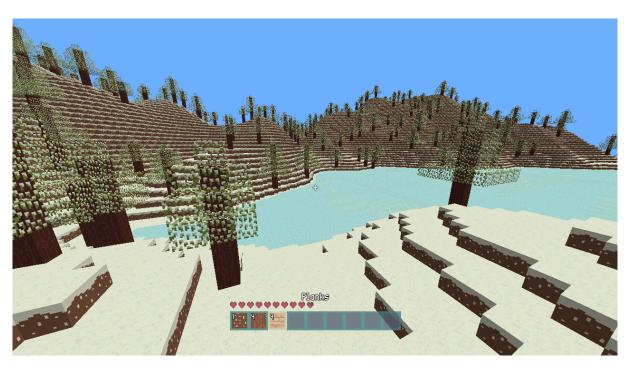


Figure 4.28: A winter biome, it features the darker type of trees whose leaves are covered in snow. The water is frozen and the ground is covered with snow.

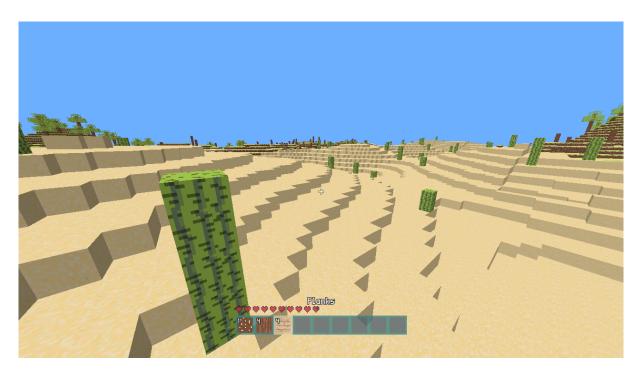


Figure 4.29: The desert biome has very flat terrain where cacti grow and there is plenty of sand almost no water.



Figure 4.30: Small house built on forest biome.