Cs319 Term Project

Section 3

Group 3D

Nomanleft

Iteration 1 Final Report

Group Members:
Ahmet Akif Uğurtan
Berkay Karlık
Can Kaplan
Teymur Bakhishli
Eren Aytüre

Supervisor: Eray Tüzün

1. Introduction

Through to implementation stage of the first Iteration, we made some progress in the development of the game and got accustomed to the development environment. We have used Intellij IDE to manage our project. For the version control, we used Git with SourceTree, a program that provides GUI for Git which made everything user-friendly and easy to manage.

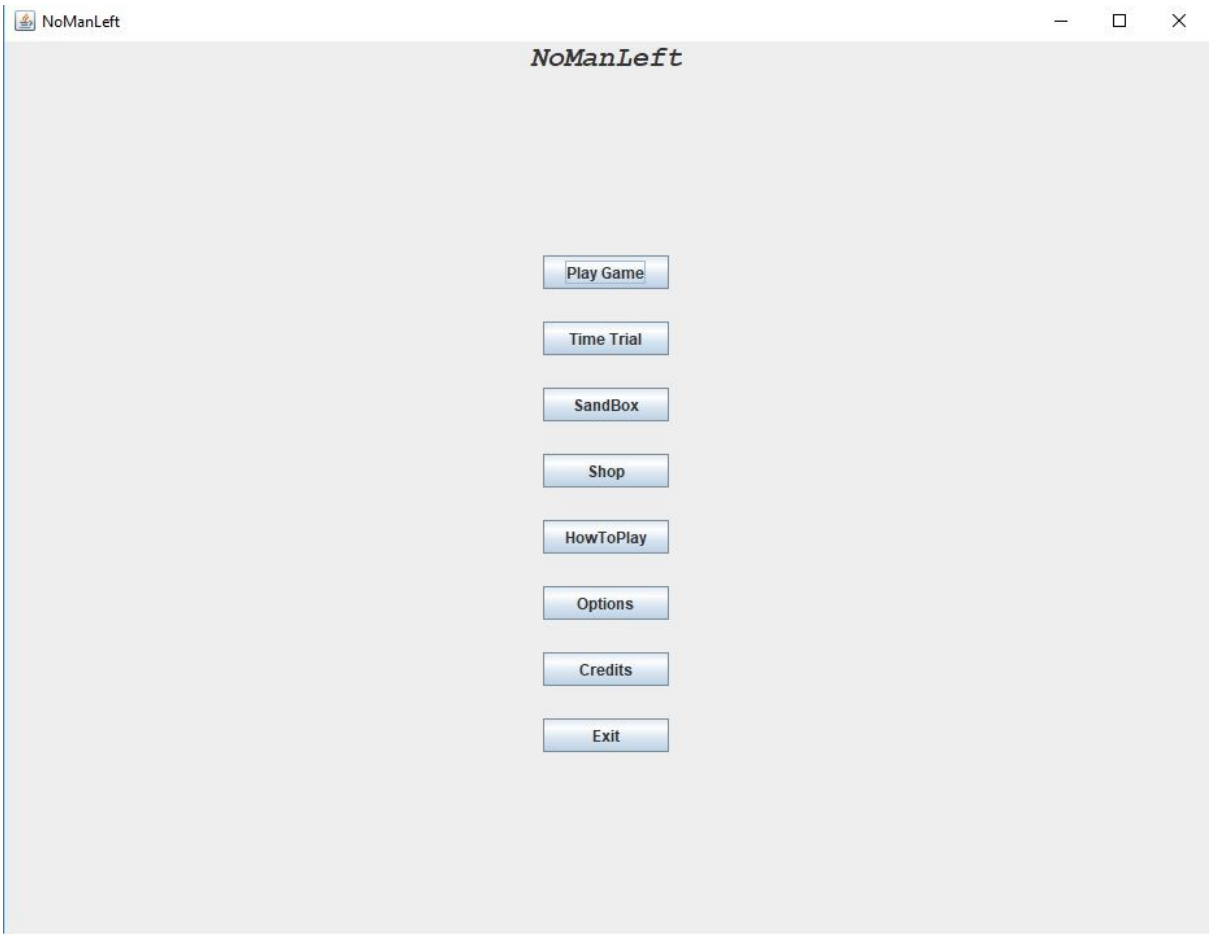
We divided the workload according to MVC pattern and started with the important parts of each subsystem. In our controller subsystem, we have GameManager and FileManager. We have implemented the core methods for a minimum value product (opening a game, playing one level).

In our model subsystem, we have completed the MapObject classes and subclasses. As we stated in Design Report, we implemented FlyWeight design pattern to reduce memory usage. The booster classes are not yet implemented. MapModel class is implemented but may require further changes.

Lastly, in view subsystem, we have implemented some parts of the LevelPanel, MainMenuPanel, PlayGamePanel, ShopPanel, and OptionsPanel. HowToPlayPanel and CreditsPanel are rather trivial so we didn't focus on them in this iteration.

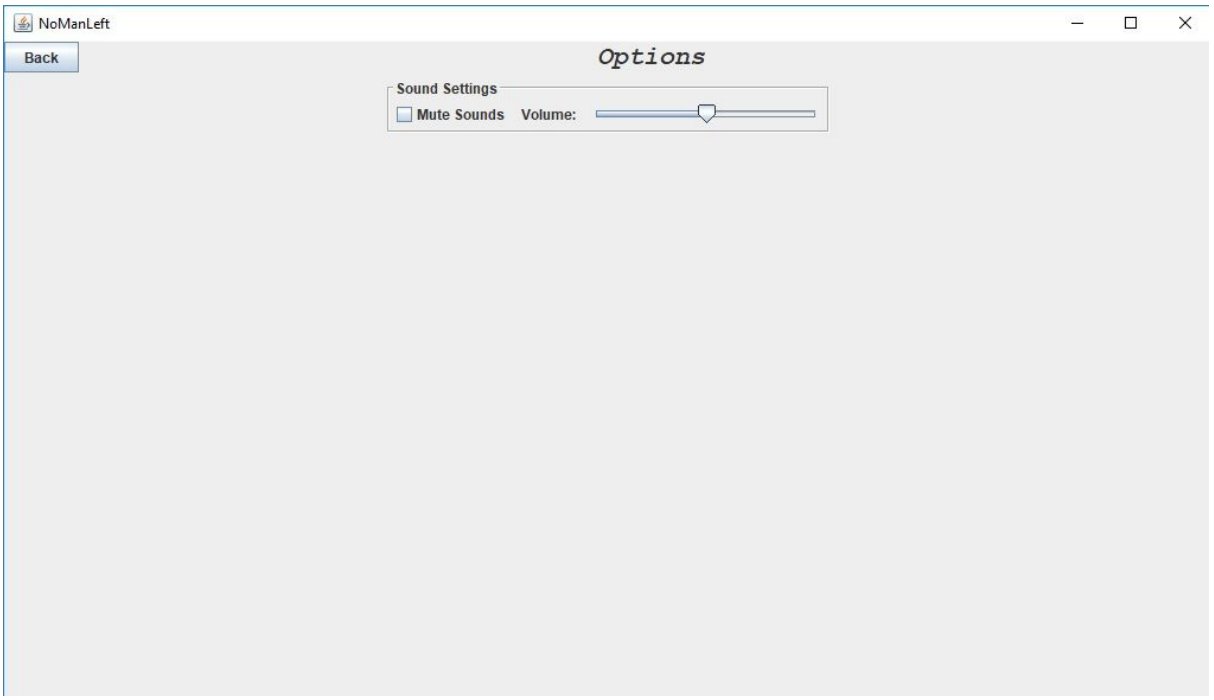
We have faced some problems due to the lack of knowledge about Intellij IDE and Version Control via Git which slowed us down. Also, we had to make some changes in our Class Diagram which we couldn't predict in the first iterations design process. Up to the point we managed to implement, the player can open the game, select one level and play it. Here are the in-game looks and their description:

1.2 Main Menu Panel



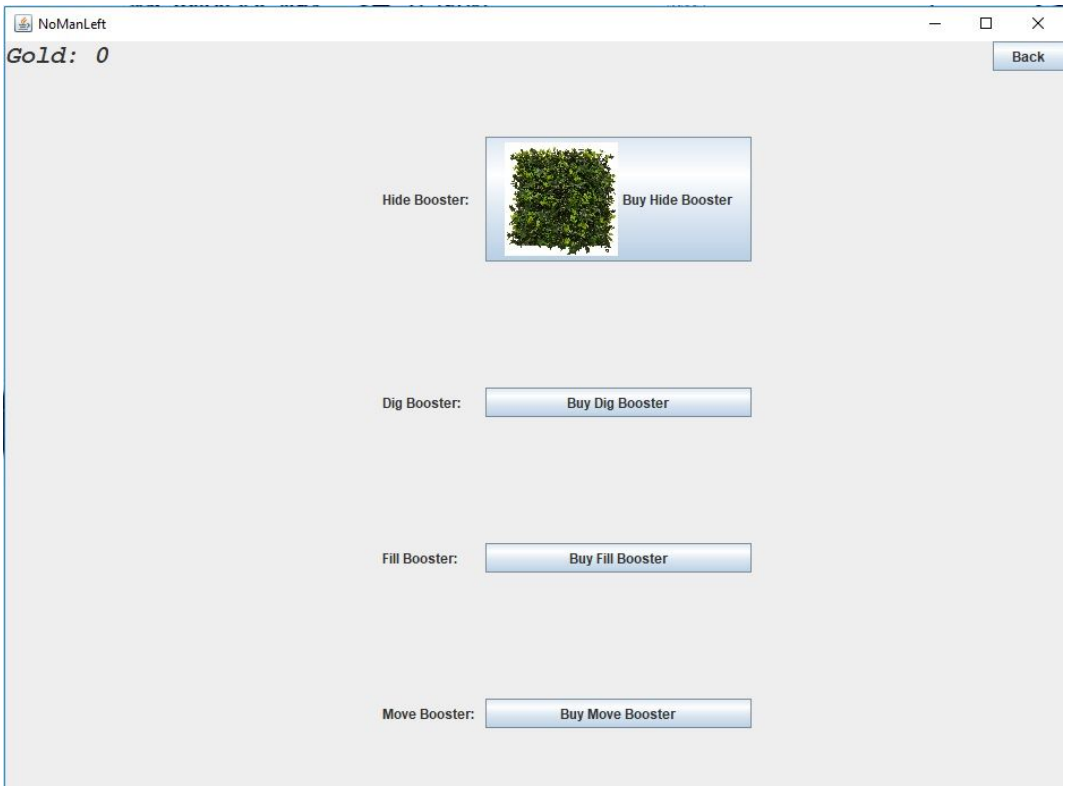
This is how our MainMenuPanel currently looks. Play Game, Shop, Options and Exit buttons are usable. When any of the PlayGame, Shop and Options buttons clicked their respective panels are loaded to ContentPane. A click to exit button closes the game.

1.2 Options Panel



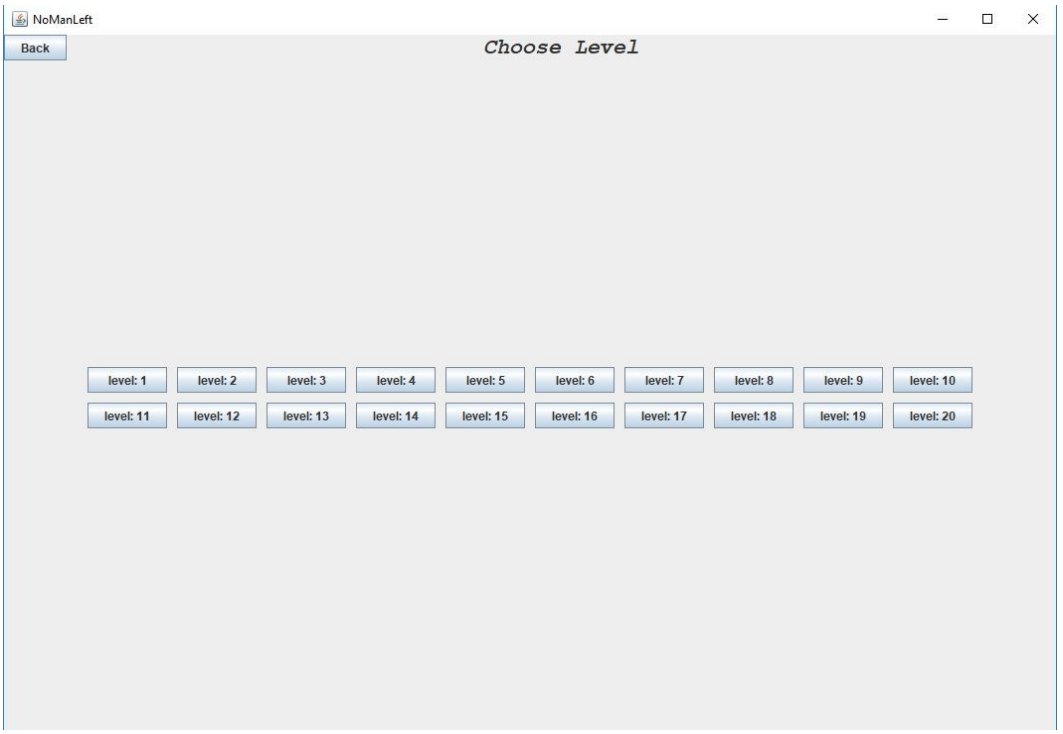
In options menu, only sound settings are implemented. The player can mute/unmute sounds or can change the volume via a slider bar. The panel may look empty for now but there will be a subpanel for customization under the SoundPanel. A click to the back button takes the player back to Menu.

1.3 Shop Panel



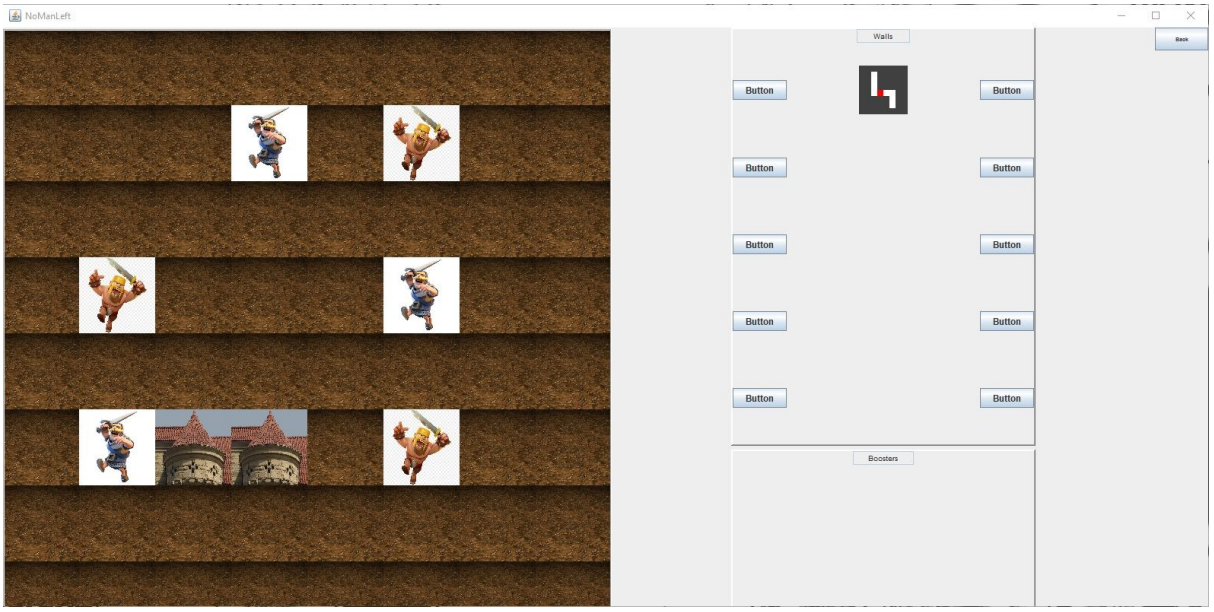
The above view is the shopPanel. There is a JPanel in the top-left part that shows the current amount of gold player has. In the center, there are buttons that player can use to buy boosters. The current shop view is rather a prototype. The button content, images, and prices will be updated as soon as we implement the boosters and multiple levels that we can gain gold from. Again Back button takes the player back to the menu.

1.4 PlayGame Panel



After clicking the PlayGame button in the Main Menu the PlayGamePanel is loaded to Frame. Here for each existing level in the file, a button is created. Since we currently have only one level, level:1 button starts a level while other buttons are just for testing the layouts and looks. In the next implementation progress, the same panel will be used for the Custom Made levels in the Sandbox mode.

1.5 Level Panel



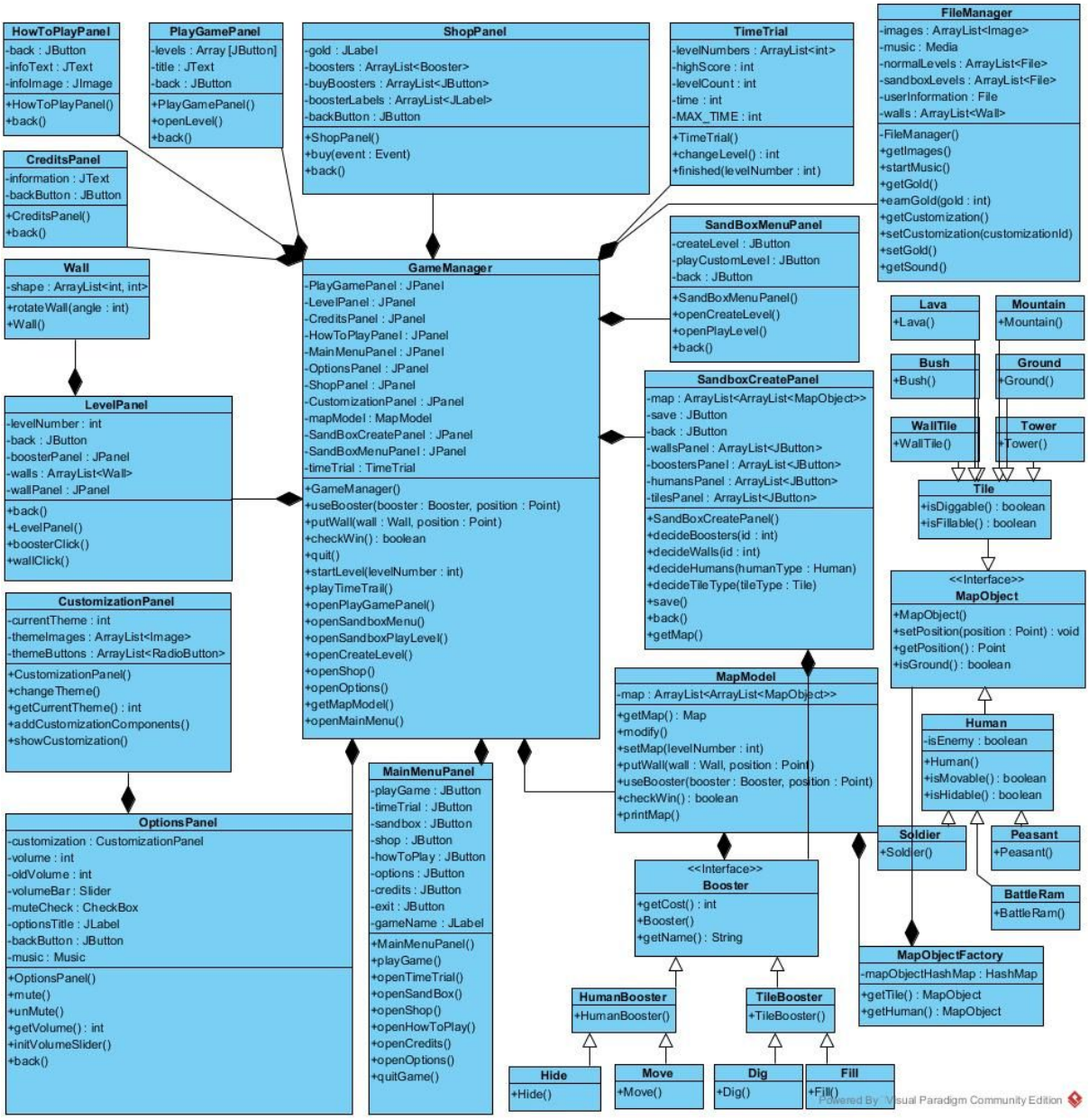
This is how our LevelPanel looks. The buttons allow the player to rotate the walls and player may drag and drop a wall to the map on the left. Back button, upon clicking, opens a pop up as follows:

If the player chooses Yes, they return to the level menu.

2. Design Changes

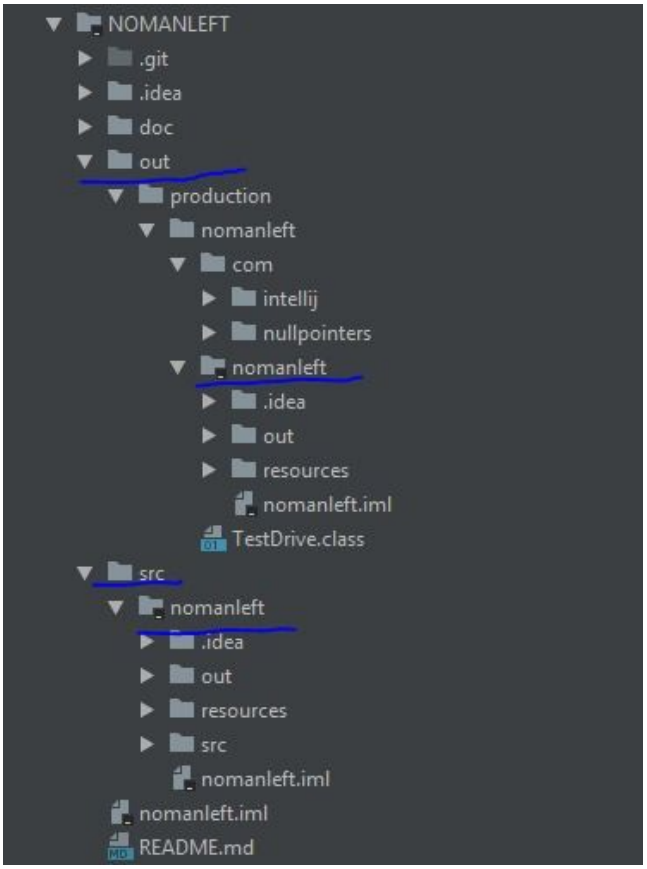
Generally, there were no substantial design change decision after the design report of the project. However, during the preliminary implementation of the project, we had few classes or method changes in order to make the whole project easy-to-code and understandable. İn FileManager class we added the setter and getter methods of all game objects (Wall, WallTile, Ground, Lava and etc.) to easily reach them from the FileManager class. We added WallPanel class to View classes which extends JPanel and uses the object of Wall class to repaint and update the map while the changes in wall structures in the game. Other than those, there have few changes in class methods. The updated form of project

Class diagram is given below:



3. Lessons Learnt

While we trying to use GitHub we made some mistakes. For example, In our project, we used IntelliJ as IDE but some of us were using the older version of IntelliJ because of that pushing and pull to the Github caused some errors. Addition to that we made a mistake at creating a project clone in our computers. One of us created the project file from the outer folder which was changed the tree of the project. When he pushed that project to the GitHub, and we did not recognize at first and pulled it and because of that our project clones are messed as well.



In order to reach images from the project, we need to create an external folder within the project folder which is the resources folder from this picture.

We learned the use of GUI Forms in IntelliJ and we learned that how to use custom from objects.

We made and trained some teamwork. For example, for implementing our levelPanel one of us need to communicate with the other who wrote the mapModel class.

4. User's Guide

4.1. System requirements & installation

The recommended system requirements for the game is having Operating System - Windows 7, 8 or 10. The game does not have special Memory or Graphics system requirements as it occupies very small storage and has basic music sounds.

Minimum Requirements:

OS	Windows 7
Processor	Core i3-6006U and above
Memory	2GB Memory
Graphics	Intel HD Graphics 520
Additional	1600*900 resolution

4.2. How to use

The user can push the play game button and choose level panel opens. After selecting enabled buttons, the user starts the game. Moreover, to understand the content of the game, the user can choose a test trial.ın the shop panel, the user can buy boosters which help the user to finish game easily. In how to play panel, there is a description of the game. In the sandbox, the user can create a map as a level of the game. In the credits panel, there is a version of the game and developers are demonstrated. In the option panel, the user can modify the volume of the sound. If the player wants to customize map objects like peasant, soldier, battle ram or other tiles, customize panel have a variety of options. When the player wants to exit, the player must push exit bıtton and there will be a pop-up screen that indicates “Are you sure to exit? ”. If the player selects exit, the application closes, otherwise, selecting cancel leads to the application not to stop.