

Systems Requirements First Implementation

By James Hicks and Jonathan Kimbro

Summary :

For our first implementation we will be focusing on our main menu when you load into the game. In the main menu we will need to implement a new game feature, a save game feature that will be added to the main, and a load game feature. We will also need to create a player data class that will contain the players data. If the player creates a new game it will create a file that will be named the name that the player entered and contain the base stats that the player will have (how many troops, how much money they have, and what level they are on). For the load class we will have to get all of the player data that the file contains and make it to where the player data equals whatever is in the file. For the save file function we will need to be able to save the current player data to whatever file name they currently have.

1. Menu Requirements

- a. The player will need to be able to select what they would like to do (create a new game or load a previous game).
- b. The player will need to be able to enter the name of the file that they are creating or loading.

2. New Game Requirements

- a. The player will need to be able to enter the name of the file they wish to create.
- b. After the player has chosen a name for that file the machine needs to enter the set amount of data that the player will have at the beginning of the game (what troops they have, how much money they will start with, and what level they will be).
- c. After the game has created the player file it will need to go to the main function so the player can start the game.

3. Load Game Requirements

- a. The player will need to enter the name of the file they previously played on, the machine will then search for the file.
- b. If the file is found the game should take the data from that file and transfer the info into the playerData class which will contain the player's data.
- c. If the file is not found it should tell the player that it could not find the file and send them back to the main menu.
- d. After the file has been found and loaded it should go to the main function.

4. Save Game Requirements

- a. The player should be able to save the file from the main method.
- b. Whenever the player saves from the main method it should find the name of the current game and find the corresponding file.

- c. Once it has found the corresponding file it should overwrite the player data in the file with the player data that is currently in the game.
- d. Once it is done with that it should return to the main method.

5. Player Data Class

- a. The player data class will need to be able to hold all of the values that will need to be carried throughout the game.
- b. It should be able to hold the file name for the game, the amount of money that the player has, the highest number of level that they have completed, and a list of troops.
- c. Whenever the game starts the playerData class should have all of the information from the file that was loaded or created.
- d. Whenever the player saves the game the contents of the playerData class should be saved to the file that the player currently has