Use case descripton: Save game

Primary actor: player

Precondition:

1. The user has created and is playing a game.

Postcondition:

1. The game has successfully saved the game.

Stakeholders and interests:

Player: To save their game and play later.

Game developer: To allow convenience for the user, to ensure the game is able to interact and maintain ongoing game data.

Main success scenario:

1. The user pauses the game.
2. The system pauses the game.
3. The user requests to view game options.
4. The system provides the user with options like save, exit, or change settings.
5. The user decides to save their game. (Alt 1:The user exits) (Alt. 2: The user changes settings)
6. The system saves the game and notifies the user about it.
7. The system provides the user the option to either exit or resume the game.
8. The user exits the game. (Alt 3: Player resumes game)

Alternative flows:

Alt 1.: User decides to exit game

- Resume at main scenario step 8.

Alt 2: User decides to change setting

- The system provides the user with the ability to change game settings.
- The user changes the settings.
- The system changes the game based on selected settings.

- Resume main scenario at step 3.

Alt 3: The user resumes the game.

- The player decides to resume the game.
- The system unpauses the game.

Exceptions:

1. Failure to save the game should result in the system to notify the user about it.

Special Instructions:

1. The game should save and notify the user instantly.

Open issues:

1. The UI to save the game or change settings may end up sharing a common space if the team focuses on other functionalities.