Case 1: Pairing Dice:

Primary Actors: Players

Stakeholders and Key Interests:

- Players: Wants to roll the dice and pair them up.
- Developers: Wants to take feedback from the player to find what was difficult/can be improved and solve the bugs
- Teachers and TAs: Wants to monitor the development.

Preconditions:

- The Board is visible on the screen.
- User can roll the dice.

Postconditions:

- System will have pairs of dice available for the game to proceed further.

Main Success Scenario:

1. The Player rolls the dice.
2. System rolls the dice for the player randomly.
3. Player gets 4 rolled dice.
4. The player selects two dice to pair them up.
5. The system pairs up the remaining two dice.
6. The dice finally are paired, and those two pairs are ready for further use cases.

Alternative Scenarios:

Exceptions:

- If the connection is lost or the game crasher, the use case ends.
- If the data is full the use case ends.

Special requirement:

- The user can pair dice only once. They cannot go back to re-select their new pairs.

Open issues:

- If player selects the pair of dice they cannot reselect their pairs.

Case 2: Load Game

Primary Actor: Players

Stakeholders and Key Interests:

- Player: Wants to load an already saved game.
- Developer: Wants to take feedback form the player to find what was difficult/can be improved and solve the bugs.
- Teachers and TAs: Wants to monitor the development of the game and grade it.

Preconditions:

- The game must have saved data to load.

Postconditions:

- A previous game with saved data will be resumed from where it was last saved.

Main Success Scenario:

1. The system provides with different option to choose from.
2. The player chooses to load a game. [Player chooses a different option]
3. The system displays the player with previously saved games.
4. Player chooses to load a specific game from the saved data.[Player has no saved game to resume]
5. The system loads the saved game.
6. The game starts off from where it was last saved.

Alternative Scenarios:

- Alt 1: Player chooses a different option:
    1. System provides user other options to choose from.
    2. Player chooses any other options.
    3. Game proceeds according to player option.

- Alt 2.: Player has no saved game to load:
    1. Player exits the load game option.
    2. System provides player other option to choose from.

3. Player chooses any other option.
4. Game proceeds accordingly.

Exceptions:

- If the connection is lost or the game crashes, the use case ends.
- If the game was not saved previously then the use case ends.

Special Requirements:

- Player can load any previous saved game from any time.
- Load game does not affect the saved data in the game.

Open Issues:

- If the save game data is tampered the game will not load properly.