

## Sprint 3 Design Rationales

*(explaining the 'Whys')*

- Explain why you have revised the architecture, if you have revised it. What has changed should be covered in the previous point. This one is about why it changed.

Actions were added for selecting and deselecting tokens. Before these actions were added, each of the other actions needed to handle these operations independently. For example, the place action needed to handle selecting the token to be placed and the remove action needed to select the token to be removed. Overall, there was duplicated functionality throughout the action classes for selecting and deselecting tokens. Therefore, this functionality was extracted into two separate action classes to reduce code duplication.

The reason for using classes instead of methods is a better assignment of responsibilities. There is an event handler to handle user input, which then generates an action based on that input. When a token is clicked, it is not known whether a place action, remove action, or other action is to take place. Assigning the token selection and deselection to methods in these classes is not the right responsibility since many classes would share a responsibility, which breaks the separation of concerns principle. Therefore, the token selection and deselection must be independent of these other actions, which is why it has its own class.

The place, slide, and jump actions were combined into a single place action class. The main reason for this is to reduce code duplication. All of these actions were responsible for moving a token from one point to another, the only difference is the game rules only allowed certain movements. This logic for checking the game rules is small, so it wasn't enough to give each class its own major responsibility. Therefore, the core functionality of moving tokens was combined into one class which performed checks for valid moves against the game rules.

- Explain 2-3 quality attributes (as non-functional requirements, e.g. usability, flexibility) that you consider relevant to the 9MM game and have explicitly considered in your design. Why are they relevant and important to your game? Show (provide evidence) how your design manifests these non-functional requirements.

Usability is very important to the game. The game is to be played by anyone with varying knowledge and experience with software interaction. If it isn't easy to understand how to interact with the game then few people will play it. Additionally, if the game is difficult to use then people who can understand it won't enjoy using it, and will be turned away. There are several design features which were implemented which ensure good usability of the game:

- The highlighting of the selected token. This informs the user that the game has processed their input.
- The ability to easily deselect a selected token and select another one on the same turn. This allows the user to easily change their mind.
- The highlighting of valid destination locations. This makes it easy for the user to choose where to move their token and to see what moves are invalid.
- The highlighting of the opponent's tokens which can be removed. This makes it easy to see what options the user has.

Portability is also an important aspect of the game. Users will want to access the game from any device, including different desktop devices. This means the game should be able to run on any operating system. The Java language allows Java programs to be run on any machine, as long as a JRE exists. This is the only requirement and usually computers can run Java files. This means writing the game in Java allows it to be played on many devices, which is more attractive to users.

- Explain at least one human value (from Schwartz's theory, e.g. achievement, tradition, freedom) that you consider relevant to the 9MM game and have explicitly considered in your design. Why is it relevant and important to your game? Show (provide evidence) how your design manifests this value.

The human value of achievement is relevant to this game. Players can use their intelligence and gain ambition by becoming successful at the game.

Our 9MM application can apply the LeaderBoard and Rankings that showcase the number of tokens for each size. As players progress further in the game this leader not only helps to identify the progress of the game but also acts as a ranking between each player. This competitor will contribute to the user feeling experiences as a motivation for users to engage more deeply in the game to defeat their opponent. And at the end when the player can have more fun if they have the disadvantage from the start but because of their motivation, they get the win.