

Delta Design Document – Final Documentation Stage

Throughout the code construction process, several changes had to be made to the requirements and design established earlier in the software creation process. The major changes are detailed below.

The Spectator Actor

During the requirements stage, it was determined that a mode where a human user could watch AI players compete would be desirable, and a Spectator actor was detailed in the requirements document. The Spectator would have been able to view the game screen without the fog of war, and see every player's pieces at once, but not affect the state of the match. During construction, the mechanism for allowing AI players to compete against each other was put in place, but a spectator mode for watching AI players take their turn was not implemented. A human user will only see the black turn transition screen as each AI player takes its turn in sequence.

Inspect Space Action

Originally, human players would have the option to inspect a hex on the game board using an Inspect action. This would allow players to see all pieces occupying a space, in case several pieces were occupying the same space and visually obscuring each other. It would have also provided information about each piece's health and other statistics. This option was not correctly implemented in the final version of the software.

Ability to Select Robot AI

The system has been designed so that the Interpreter component can parse any robot script, even those which are malformed. However, an interface for the player to select exactly which robot script to use for each AI was not integrated in time, so AI players can only be created with a default robot script.

Display Component

One of the largest changes to any component involved making each screen its own subclass of the Screen class. The original design of the Display component called for the GameScreen inheriting from Screen because of the extra functionality it would require, and for other, more basic screens like the Title Screen to be an instance of Screen. However, during construction it was decided that it would be easier to create the component by making every screen a subclass of the Screen class.

Integration Issues

From the eight Display screens that were mentioned in the Design document, some could not be fully implemented and some could not be integrated with the rest of the system. The following display screens are implemented, but are not fully integrated with the system.

Robot Archive Screen

Because the features to revise, retire and register a robot are not yet implemented, the Robot Archive screen was decided to not be displayed and integrated to the system before integrating its features.

Settings

The options such as sound volume and keyboard keys were offered via the settings screen. Because these features were in "Nice-to-have" and have not been implemented, the Settings screen is not integrated to display.

Results

The Results screen is not yet fully integrated. The information about each team's performance needs to be fetched and displayed, and the ability to start a new match with the same parameters ("Rematch") was not implemented.

Screens that Need Improvement

There are some Screens that, while implemented, could be improved in some ways. These are noted below.

Game

The Game screen is functional. However, the exit button is not yet incorporated into the system. The panning option is also not yet implemented, and the side status bar for each team is not well-developed, although it shows correct information at all times.

Additionally, though the Board's logic does handle the rotation of pieces correctly, visual rotation of pieces was not implemented by the Display.

Results

The Results screen does not yet show proper results as described in the requirements document. The pictures of the scout, sniper and tank need to be put, as well as information about each team's pieces. Moreover, little tweaks such as color and position need to be made.

Logger

The Logger was described as a "might-have" in the requirements document and has been implemented nearly to its entirety. However, there still exist some bugs within the Logger component that have not been resolved and the Logger component has not been integrated with the Display component. Rather than

Flow

The Flow component had the greatest number of changes. During construction, it was decided that the AWTEventCatcher would not be implemented. Instead, the GameManager and the MenuManager classes would handle all the user events. The AWTEventCatcher could not be completely removed since it ties the component together.

There are some improvements that still need to be made for the Flow component. The Flow needs to integrate the Robot Archive features as well as the Settings features. It also needs to integrate the Logger into the display.