

Class: AmazeActivity

Responsibility:

This class is responsible for presenting user a title interface. User can select parameter settings: size, generation algorithm and driver algorithm. I use seek bar to present options for size and spinner to present options for two algorithm selections. Once the "explore" button is clicked, move to the generation stage with an intent

Collaborators: AppCompatActivity, GeneratingActivity

Class: GeneratingActivity

Responsibility:

This class is responsible for presenting user an "intermediate" interface of generating a maze. Since the actual maze generation is not integrated yet, I use Async class to intentionally delay execution, which waits the background thread to do all computations but does not block the UI thread. Once the progress bar reached 100%, move to manual or animation stage based on user's input. If the user presses the back button, cancel all computations and get back to the title interface

Collaborators: AppCompatActivity, PlayingManuallyActivity, PlayingAnimationActivity. AMazeActivity

Class: PlayManuallyActivity

Responsibility: Displays the maze and lets the user manually navigate the robot through the maze. Provides a feature to toggle visibility of the map plus functionality to toggle visibility of the solution on the map. show the whole maze from top or not; show the solution in the maze or not; show the currently visible walls or not; scale the size of the map; Pressing the back button returns to State Title to allow the user to choose different parameter settings and restart.

Collaborators: AppCompatActivity, WinningActivity, AMazeActivity

Class: PlayAnimationActivity

Responsibility: Displays the maze and lets the user watch a robot exploring the maze. Displays the remaining energy, consider using a ProgressBar for this. Show the whole maze from top or not. show the solution in the maze or not. show the currently visible walls or not. scale the size of the map. Screen provides a start/pause button to start the exploration and to pause the animation. Introduce a button "Go2Winning" to directly move the UI to State Winning, and a corresponding button "Go2Losing".

Collaborator: AppCompatActivity, AMazeActivity, WinningActivity, LosingActivity

Class: WinningActivity

Responsibilities: Displays the finish page for the case of winning the game and informs the user what happened and how to restart the game. Shows the overall

consumption. Shows the Shows the length of the path taken and the length of the shortest possible path.If the user played the game manually, only shows the length of the path taken and the length of the shortest possible path. If pressing the back button, then returns to State Title.

Collaborators: AppCompatActivity, AMazeActivity

Class: LosingActivity

Responsibility: Displays the finish page for the case of winning the game and informs the user what happened and how to restart the game. Shows the overall energy consumption. Shows the length of the path taken and the length of the shortest possible path. Visualizes if robot stopped for lack of energy, or if it is broken. Pressing the back button returns to State Title.

Collaborators: AppCompatActivity, AMazeActivity