



Game Manager

Purpose: A centre point for the game. Maintains the state of the game

Should store the state of the game

File Manager

Purpose: Loads and saves game files

Should save the state of each game button

Should save the state of the boost game button

Should store the total time played

Should store the achievement model

Should store the total score

On restart state, should send load save file to all game objects

Game Button Model

Purpose: Contains all data required for the game buttons and the functionality that they need

Data Members: Name, Cost, Initial Score, Score, Amount, Cooldown, Cost To Automate, Elapsed Time, Automated

Responsibility:

Create a timer for when clicked – when the timer is running, disable the mouse area

Update values when more of the game button has been bought

If the automate has been clicked, keep a record of this and repeatedly send out signals every time the timer has finished

When the button has been bought, the score, amount and cost of the button is updated

Boost Button Model

Purpose: Contains all data required for the boost button activity

Data Members: Elapsed Time, Boost Cooldown, Active Cooldown, Used

Responsibility:

Should keep track of the last time it was clicked

Should keep track of how much time it has been active

Should keep track of if it has been used before (achievement)

Achievement Model

Purpose: Provides a collection of achievements and their states

Data Members: Achievement, State

Responsibility:

Should maintain a collection of achievements and their connected states

Achievement

Purpose: Creates the different achievements within the game. Stores all data needed for this

Data Members: Name, Image, Description

Responsibility:

Should show the title of the achievement and the accompanying image. A more detailed explanation is then needed which will be used when the achievement is clicked on

Settings Model

Purpose: Contains all data for the settings – stores the properties

Data Members: Sound Volume, Resolution, FullScreenState

Responsibility:

Should keep note of the state of the settings

When the sound is changed, the application's volume should increase

When the resolution is changed, the application's screen should update

When the full screen state is changed, the application's window should update

Game Button

Purpose: View for the game button model

Responsibility:

Must look at the Game Button Model and maintain the properties

Display: Show the UI needed for the game button

Buy Button Pressed: When the user clicks to buy the game button emit `onBuyButtonPressed`

Game Button Pressed: When the user clicks on the game button, emit the `onGameButtonPressed`

Automate Button Pressed: When the user clicks to automate the game button, emit the `onAutomatePressed`

Boost Button

Purpose: View for the boost button model

Responsibility:

Must look at the Boost Button Model and maintain the properties

Display: Show the UI needed for the boost button

Boost Clicked: When the user clicks on the boost, emit onBoostClicked

Game Screen

Purpose: A container for the various game items - contains all items needed to play the game

Responsibility:

Display: Show the six game buttons and the one boost button, the total score and a back button

Game Button Pressed: emit onGameButtonPressed

Boost Button Pressed : emit onBoostButtonPressed

Back Pressed: emit onBackPressed

Main Menu

Purpose: Provides the user with the options to go between the various screens

Responsibility:

Maintain the state of the game – when changing states, the screen visible needs to be updated.

Display: When shown, display the various UI required

Start Clicked: Game screen needs to be shown

Restart Clicked: Game screen needs to be shown

Achievements Clicked: Achievement screen needs to be shown

Settings Clicked: Settings screen needs to be shown

Quit Clicked: The application will close

Settings Screen

Purpose: View for the settings model – displays sliders/checkboxes etc. for changing various settings

Responsibility:

Display: Using properties provided, display the various UI needed

Sound Changed: When the user moves the progress bar for the sound, onSoundChanged is emitted

Resolution Changed: When the user selects a new resolution, onResolutionChanged is emitted

Full Screen Changed: When the user checks/unchecks full screen mode, the onFullScreenChanged is emitted

Back Pressed: When the user selects the back button, the onBackPressed is emitted

Achievement Screen

Purpose: View for the achievement model – displays the achievements in a grid like view

Responsibility:

Display: Display the Achievement model with various UI needed

Back Pressed: When the back button is pressed, the onBackPressed is emitted

Achievement Selected: Create an overlay displaying properties of the achievement selected