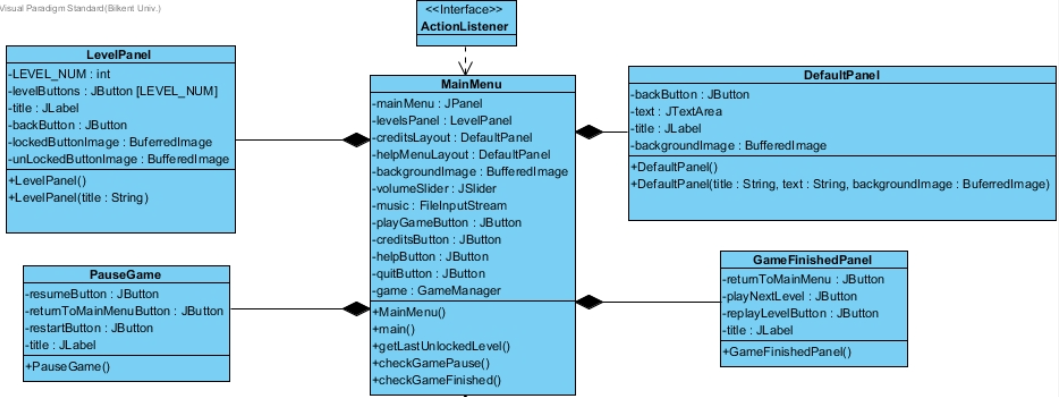
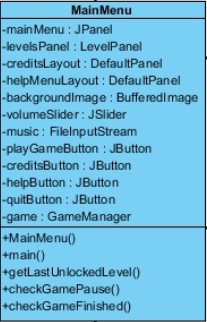
**User Interface Subsystem:**

This subsystem includes the classes which have the features of the interface of the game.



**MainMenu Class:**



**Attributes:**

* **mainMenu:** mainMenu is a JPanel that will display the Main Menu contents.
* **levelsPanel:** levelsPanel is a LevelPanel which is a separate panel from mainMenu. It includes the features provided by LevelPalen class.
* **creditsLayout:** creditsLayout is a DefaultPanel, having the features of DefaultPanel class. In this panel, brief information about the developers of “Run Dot Run” will be shown as a text**.**
* **helpMenuLayout:** helpMenuLayout is also a DefaultPanel. It includes a text of description of the game.
* **backgroundImage:** backgroundImage is aBufferedImage which will be seen on the background of the game.
* **music:** music is a FileInputStream, which will be playing on the background of the game.
* **volumeSlider:** volumeSlider is a JSlider that enables the user control the volume of the background music
* **playGameButton**: playGameButtonis a JButton that will direct the user to the levelsPanel when clicked.
* **creditsButton:** creditsButton is a JButton that will open creditsLayout.
* **helpButton:** helpButton is a JButton that will open helpMenuLayout.
* **quitButton:** quitButton is a JButton to quit the game.
* **game:** is a GameManager as the reference to the GameManager class.

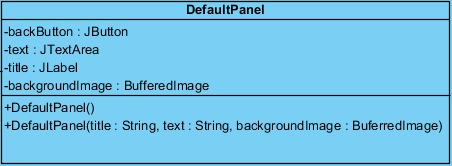
**Constructor:**

* **MainMenu():** is the default constructor.

**Methods:**

* **main():** is the method to initiate the game.
* **setLastLockedLevel():** this method sets the last level to keep in which level the game is left and up to which level the others are unlocked.
* **getLastLockedLevel():** gives the last locked level.
* **checkGamePause():** returns true if the game is paused, and opens PauseGame panel.
* **checkGameFinished():** checks if the game is over and opens gameFinished panel.

**DefaultPanel Class:**



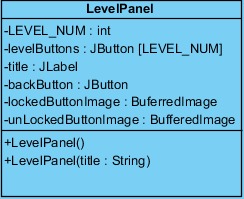
**Attributes:**

* **backButton:** backButton is a JButton that will help the user go to the previous page when it is clicked.
* **text:** text is a JTextArea that will include some text depending on where it is called.
* **title:** title is a JLabel which will include the name of the current page.

**Constructor:**

* **DefaultPanel():** is the default constructor.
* **DefaultPanel(title: String, text: String, backgroundImage: BufferedImage):** this constructor takes the title, text and background image of the current page.

**LevelPanel Class:**



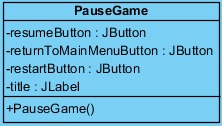
**Attributes:**

* **LEVEL\_NUM:** is an integer for the number of levels.
* **levelButtons:** levelButtons is an array of JButton with size of LEVEL\_NUM. Each button corresponds to different levels and starts the game when clicked.
* **title:** title is a JLabel including the name of the panel.
* **backButton:** is the button to help the user go to the previous page.
* **lockedButtonImage:** is a BufferedImage which will be placed on the levelButtons which are locked.
* **unLockedButtonImage:** is a BufferedImage which will be placed on the levelButtons which are unlocked.

**Constructor:**

* **LevelPanel():** default constructor.
* **LevelPanel(title: String):** this constructor takes the title as parameter.

**PauseGame Class:**

****

**Attributes:**

* **resumeButton:** resumeButton is a JButton that returns the game from the pause panel.
* **returnToMainMenuButton:** is a JButton helping the user return to main menu page.
* **restartButton:** is a JButton which restarts the current level.
* title: is a JLabel for the title of the Pause page.

**Constructor:**

* **PauseGame():** is the default constructor.

**GameFinishedPanel Class:**

****

**Attributes:**

* **returnToMainMenu:** is a JButton for the user to go back to the beginning page, which is the MainMenu.
* **playNextLevel:** is a JButton to pass the next level when the level is completed successfully.
* **replayLevelButton:** is a JButton which enables the user to play the same level again.
* **title:** is aJLabel for the title of GameFinishedPanel to be shown when the game is over.

**Constructor:**

* **GameFinishedPanel():** is the default constructor.