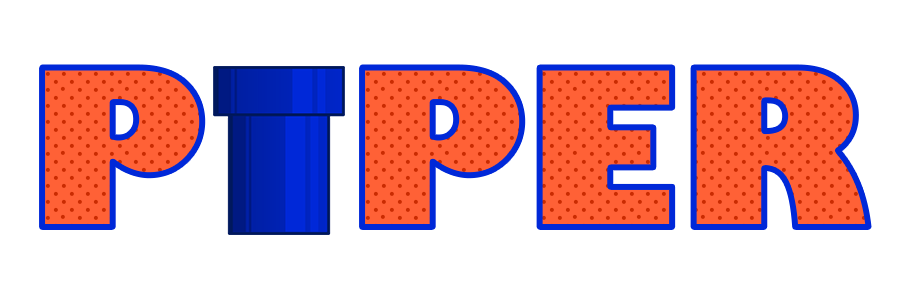
**JAVAFX PROJECT REPORT**

**BOARD GAME ‘PIPER’**

****

**ID : 150117015 ID : 150117021**

**Name : Yasin Enes Name : Cenk Kaan**

**Surname : POLAT Surname : Kanar**

**Problem Definition**

This game's goal is to create a proper path for a ball to travel from the beginning point to the end point by dragging boxes. However not all boxes are able to move. Empty, start, end and static boxes can not be moved.

Moreover user shall not transport a box diagonally, turn it upside down or any directions.

**Implementation Details**

***UML* *Diagrams***

|  |  |
| --- | --- |
| Pane16 | |
| +  +  +  +  +  +  +  +  + | media : Media  correct : MediaPlayer  level1 : Level  level2 : Level  level3 : Level  level4 : Level  level5 : Level  currentLevel : Level  entranceScene : Scene |
| +  +  +  +  + | start(Stage primaryStage) : void  reListen(VBox[] arr, ArrayList<ImageView> imgArr) : void  doAction(int i, double x, double y, VBox[] arr, ArrayList<ImageView> imgArr) : void  check(ArrayList<ImageView> imgArray, int k, int currentPlace) : boolean  compare(Image i1, Image i2) : boolean |

|  |  |
| --- | --- |
| Level | |
| -  - | level : String  completed : boolean |
| +  +  +  +  + | Level(String s)  showLevel(Level lvl, VBox[ ] vb, Arraylist<ImageView> arr) : void  selectImage(String s1, String s2) : ImageView  isSolved(ArrayList<ImageView> imgArr) : boolean  solve(Level currentLevel, Circle circle) : void  getters/setters |

* **reListen** method simply adds mouse drag listener to each VBoxes after clicking on next or previous level buttons. Because, VBoxes’ listeners being deleted when the user complete the path.
* **doAction** method simply does the action as the name indicates. Changes the ImageViews of source and destination VBoxes. The method determines the destination VBox by checking where the user drags the mouse.
* **check** method checks if the source tile can be moved and also checks the destination tile whether the source tile can move there or not.
* **compare** method takes two Images as parameter and compares them using PixelReader. If they are equal returns true, otherwise returns false.
* **showLevel** method simply changes the ImageViews of all VBoxes in the ArrayList. It takes the tile types from the .txt file in the Levels folder and gets the ImageView using **selectImage** method and assigns it to the corresponding VBox.
* **selectImage** method first gets the .png file from Proje folder, creates Image with it and assigns that Image to an ImageView, then changes its height and width values. Lastly returns the ImageView to the caller.
* **isSolved** method simply checks if the currentLevel is completed or not. It first finds the starter tile, base on that tile, checks if the path created or not dynamically. If correct path created, returns true, otherwise returns false.
* **solve** method shows the animation if the currentLevel is completed. Circle starts from the starter tile and follows the correct path and arrives to the end tile statically.

**Incomplete Part(s) of Our Program**

Our program meets the minimum reqirements except one thing. Circle’s animation path created statically. If we have had more time, I believe we can implement dynamic one.

To writing the method for control of solution was difficult for us. Because it checks the solution path dynamically.

When we create a method for changing tiles places, firstly the method didn’t work with wireless mouse, because wireless mouses sometimes jump a few pixels on screen and method checks direction with pixels.

We increased the quality of tiles’ pictures after a few days.

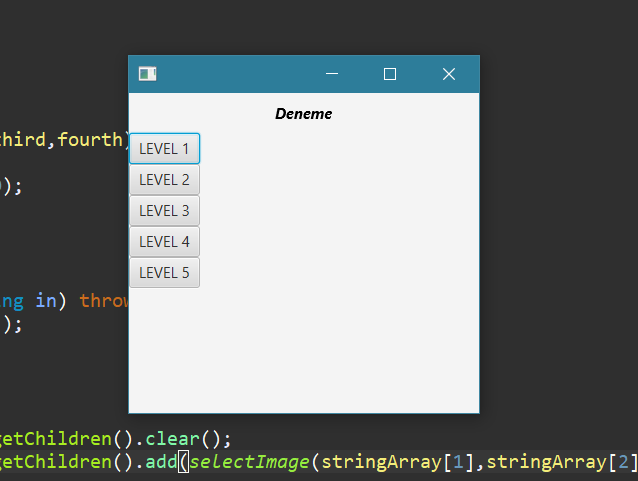
We added a main menu, info menu, background music to our program. Info menu has a text which is information about how Piper game works. Main menu has Play button, Info button and music sound button in its scene.

We added a few animations to our program. Some buttons starts an slide animation after clicking it. For example, if Play button clicked, Play button and text will start to move left. After animation stopped, scene changes to the game board scene.

If user clicks sound button, music’s volume changes depends on current volume value.

If the current level finished, a sound for correct solution will automatically plays.

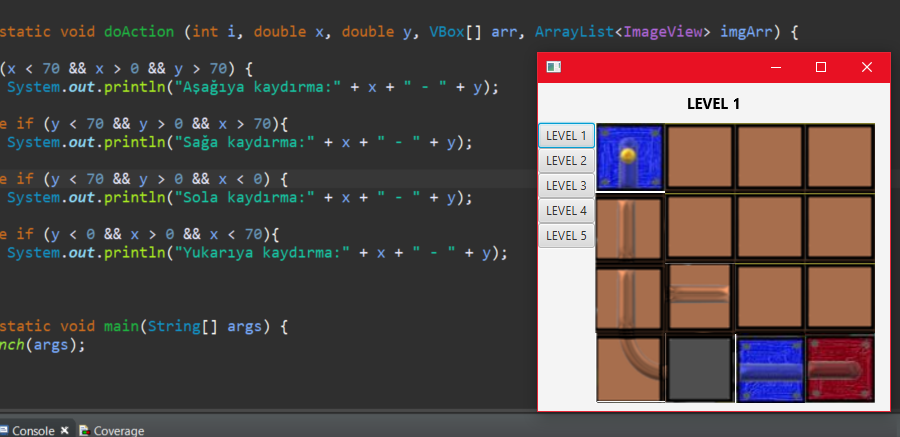
**Test Cases**



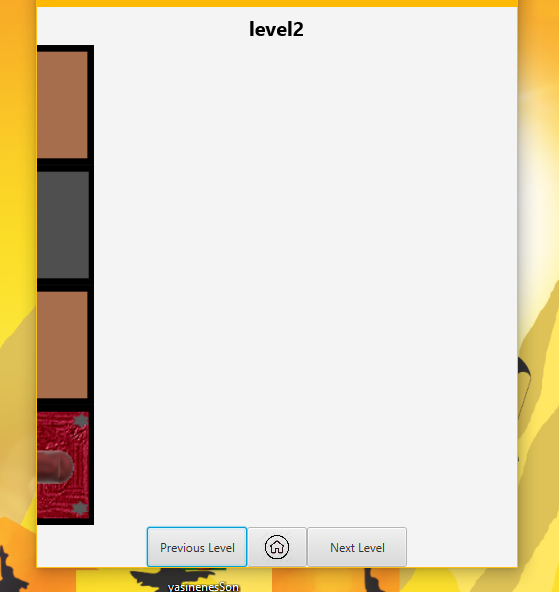
1) We created the scene with 5 level buttons and a text.



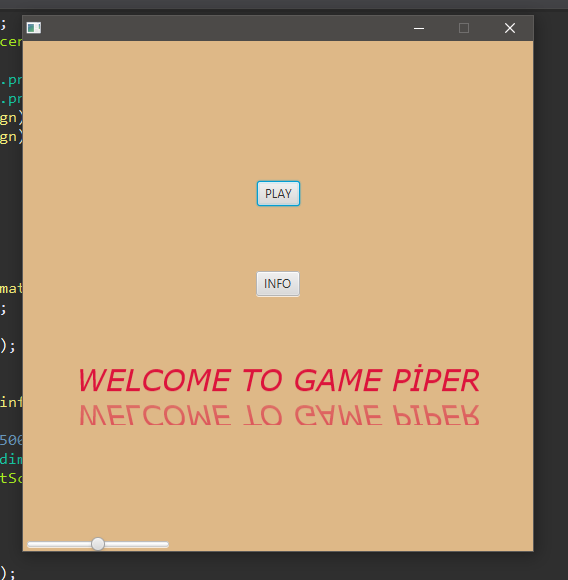
2) We added the pipe tiles to the scene with function.



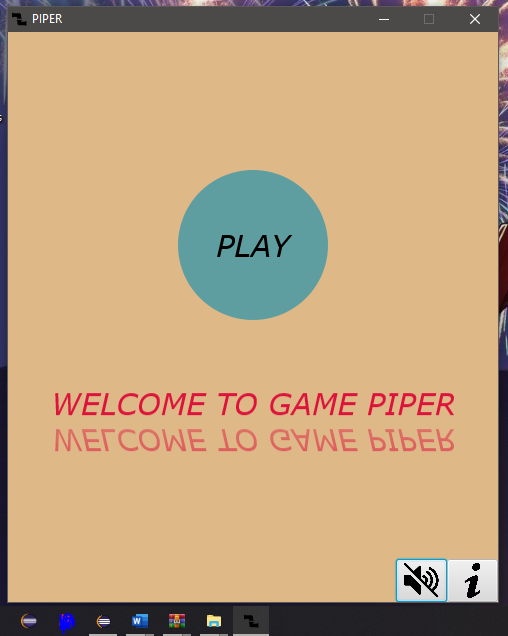
3) The drag handler method implemented.



4) Design changed and animation added to game board.



5) Main menu added to our program



6) Main menu design changed, few animation added.



7) Final design of main menu.