Programming Technology 1

Adnan Adnan (NBDK4L)

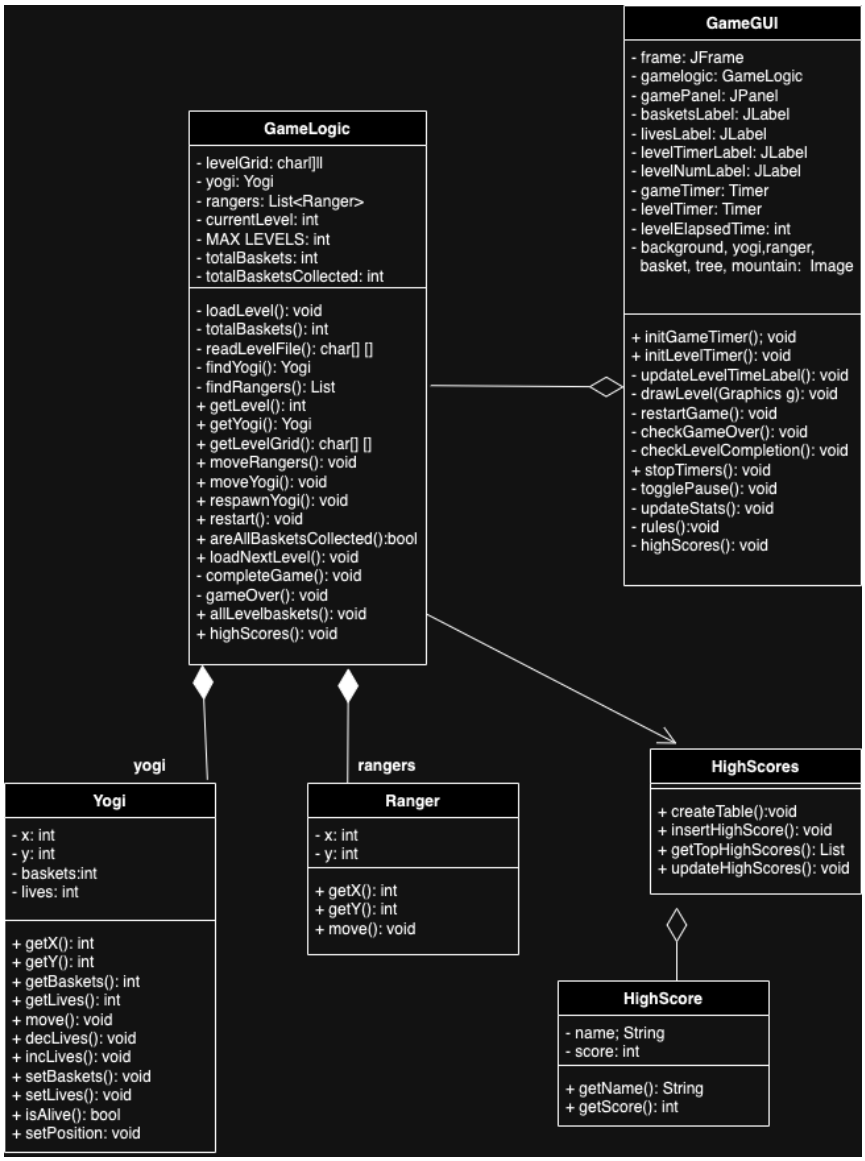
YOGI BEAR – Task (1)

Description of the Task:

Yogi Bear embarks on an adventure in Yellowstone National Park to collect picnic baskets. The park is filled with mountains, trees, and rangers who move horizontally or vertically. Yogi can collect picnic baskets but has to avoid getting too close to the rangers, as getting within one unit of them will cause Yogi to lose one life. Yogi starts with three lives, and if he loses all of them, a game over screen is shown. During the game, a counter tracks the number of baskets Yogi collects, and once all baskets are collected, a new level is generated or loaded.

Additionally, if Yogi loses all his lives, a message box pops up asking the player for their name to save their score. There is also a highscore table that displays the top 10 scores and a restart option for starting a new game.

Class Diagram:



Method Descriptions:

Yogi Class

The Yogi class represents the main character, Yogi Bear, in the game. This class is responsible for tracking Yogi's position on the grid, his lives, and the number of picnic baskets he collects.

- **move()**:
  Moves Yogi by changing his position according to the given change in x (dx) and y (dy).
- **decLives()**:
  Decreases Yogi's life count by 1.
- **getX()**:
  Returns the current x-coordinate of Yogi.
- **getY()**:
  Returns the current y-coordinate of Yogi.
- **getBaskets()**:
  Returns the number of picnic baskets Yogi has collected.
- **getLives()**:
  Returns the number of lives Yogi has left.
- **setBaskets()**:
  Sets the number of baskets Yogi has collected to the value n.
- **setLives()**:
  Sets the number of lives Yogi has to the value n.
- **isAlive()**:
  Returns true if Yogi has more than 0 lives, indicating that Yogi is still alive.
- **incBaskets()**:
  Increases the number of baskets Yogi has collected by 1. This method is typically called when Yogi collects a basket.
- **setPosition()**:
  Sets Yogi's position to the specified coordinates (x, y).

Ranger Class

The Ranger class represents a ranger character in the game, whose primary role is to track the position and movement of rangers on the park grid. Rangers move in the park, and if they get too close to Yogi Bear, they will cause Yogi to lose a life.

- **getX()**:
  Returns the current x-coordinate of the ranger.

- **getY()**:

  Returns the current y-coordinate of the ranger.

- **move()**:

  Moves the ranger by the given amount along the x and y axes.

Game Logic Class

The GameLogic class is managing the grid, Yogi's actions, ranger movements, and the collection of baskets. It handles the game's levels, player interactions, and game-over conditions. The class ensures the proper flow of the game, including checking win conditions, restarting the game, and advancing through levels.

- **loadLevel()**:
  Loads the level from the given file and initializes the game state (Yogi's position, rangers, total baskets).
- **totalBaskets()**:
  Calculates and returns the total number of baskets ('B') in the current level grid.
- **readLevelFile()**:
  Reads a level file and returns a 2D character array representing the grid.
- **findYogi()**:
  Searches the grid for Yogi's starting position ('Y') and returns a Yogi object with the coordinates.
- **findRangers()**:
  Searches the grid for rangers ('R') and returns a list of Ranger objects with their positions.
- **moveRangers()**:
  Randomly moves each ranger in the grid (either horizontally or vertically). If a ranger moves to Yogi's position, Yogi loses a life. If Yogi's lives reach 0, the game is over.
- **moveYogi()**:
  Moves Yogi by the specified deltas (dx and dy), which correspond to horizontal and vertical movement.
  Yogi can collect baskets ('B') while moving, and if he encounters a ranger ('R'), he loses a life.
- **respawnYogi()**:
  Respawns Yogi at the starting position (0, 0) when he loses a life.
- **restart()**:
  Restarts the game from level 1, resetting Yogi's lives, baskets, and the game grid.
- **areAllBasketsCollected()**:
  Returns a boolean indicating whether Yogi has collected all baskets in the current level.
- **loadNextLevel()**:
  Advances to the next level by loading a new grid and resetting relevant game data. If the maximum level is reached, the game ends and prompts the player for their name to save the score.

- **completeGame()**:
  Called when the player finishes all levels, prompting the player for their name to save the score, and then exiting the game.
- **allLevelbaskets()**:
  Adds the baskets collected in the current level to the total baskets collected across all levels.
- **gameOver()**:
  Displays the game over screen and prompts the player for their name to save the score when Yogi loses all lives.
- **highScores()**:
  Displays the leaderboard with the top 10 scores stored in a database.

GameGUI Class

The GameGUI class is responsible for managing and rendering the graphical user interface (GUI) of the Yogi Bear game.

- **initGameTimer()**
  Initializes and starts the game timer to control the movement of rangers at regular intervals.
- **initLevelTimer()**
  Initializes and starts the level timer to track the elapsed time for the current level.
- **updateLevelTimeLabel()**
  Updates the label displaying the time elapsed in the current level.
- **drawLevel(Graphics g)**
  Renders the current level grid, drawing various game elements like trees, baskets, and characters.
- **restartGame()**
  Restarts the game from the beginning, resetting game states and reinitializing timers.
- **checkGameOver()**
  Checks if the game is over by verifying if Yogi has no remaining lives.
- **checkLevelCompletion()**
  Checks if all baskets are collected to mark the level as complete and load the next level.
- **stopTimers()**
  Stops both the game and level timers to pause the game.
- **togglePause()**
  Toggles between pausing and resuming the game, stopping or starting the timers accordingly.
- **updateStats()**

Updates the displayed game statistics, including baskets collected, lives remaining, and current level.

- **rules()**
  Displays a dialog showing the game rules.
- **loadImages()**
  Loads all the necessary image assets for rendering game objects in the GUI.
- **highScores()**
  Displays the high scores table by calling the corresponding method from the GameLogic class.

Event and Event handlers:

GUI Buttons and Menu Items:

- The "Pause" button toggles the game pause state by calling togglePause().
- The "Rules" button displays the game rules using rules().
- The "Restart" menu item restarts the game with restartGame(), resetting the game state.
- The "High Scores" menu item shows the high scores by calling highScores().

Key Events:

- Arrow key inputs or WASD keys call gameLogic.moveYogi() to move the Yogi character in the game grid.
- After each key press, the game state is updated with updateStats(), and the level completion and game over conditions are checked via checkLevelCompletion() and checkGameOver().

Testing:

- **White Box Testing**: Tests were conducted to validate all logical conditions, including edge cases.

1. Yogi Moving

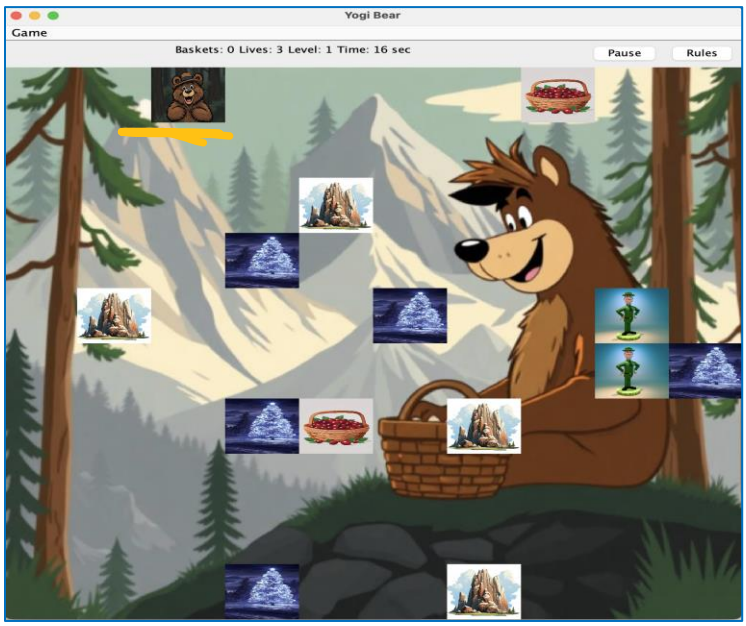
AS a user

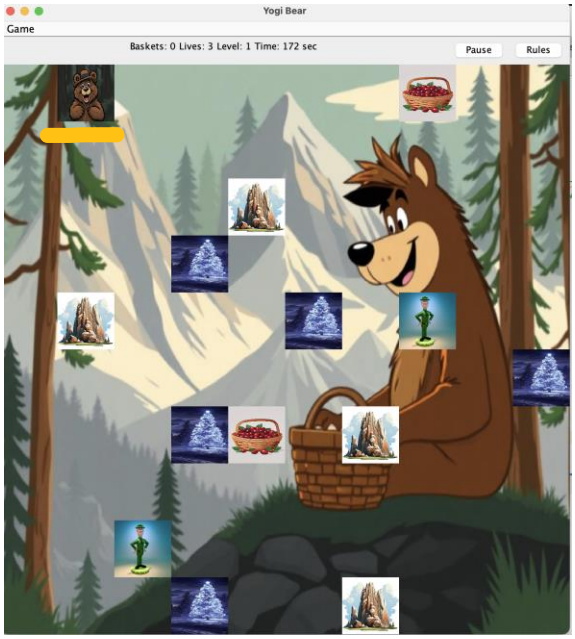
I WANT TO move Yogi left

GIVEN the game is running and Yogi is in the initial position

WHEN I press the "A" key

THEN Yogi should move one cell to the left on the grid and the game should update accordingly.





2. Yogi Collecting a Basket

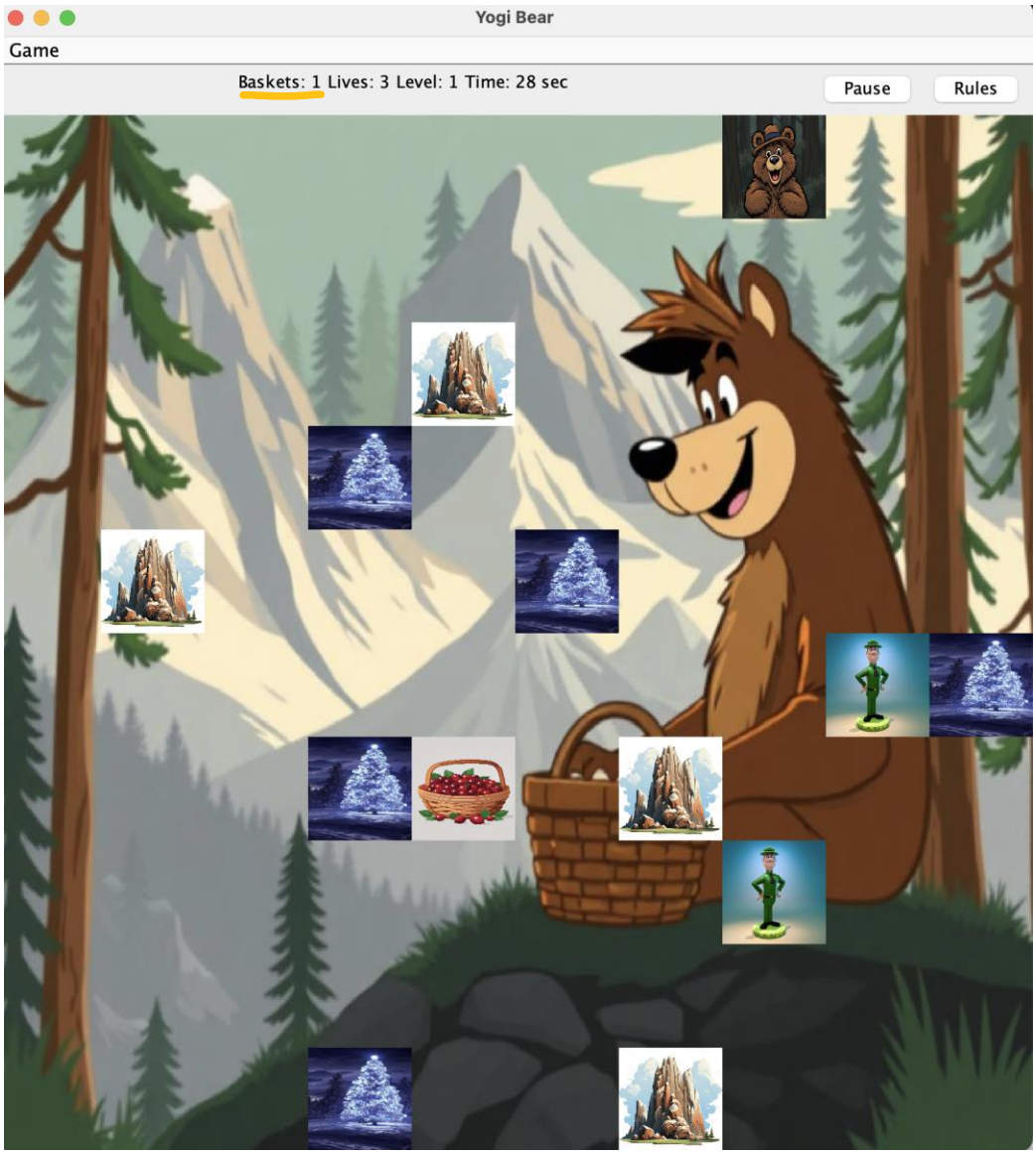
AS a user

I WANT TO collect a basket

GIVEN Yogi is on a grid cell with a basket (B)

WHEN Yogi moves to that cell

THEN the basket count should increase by 1, and the game should update the display to reflect the new basket count.



3. Yogi Colliding with a Ranger

AS a user

I WANT TO lose a life if Yogi collides with a ranger

GIVEN Yogi is on a grid cell with a ranger (R)

WHEN Yogi moves into that cell

THEN Yogi should lose a life, and the lives counter should update accordingly.

4. Level Completion

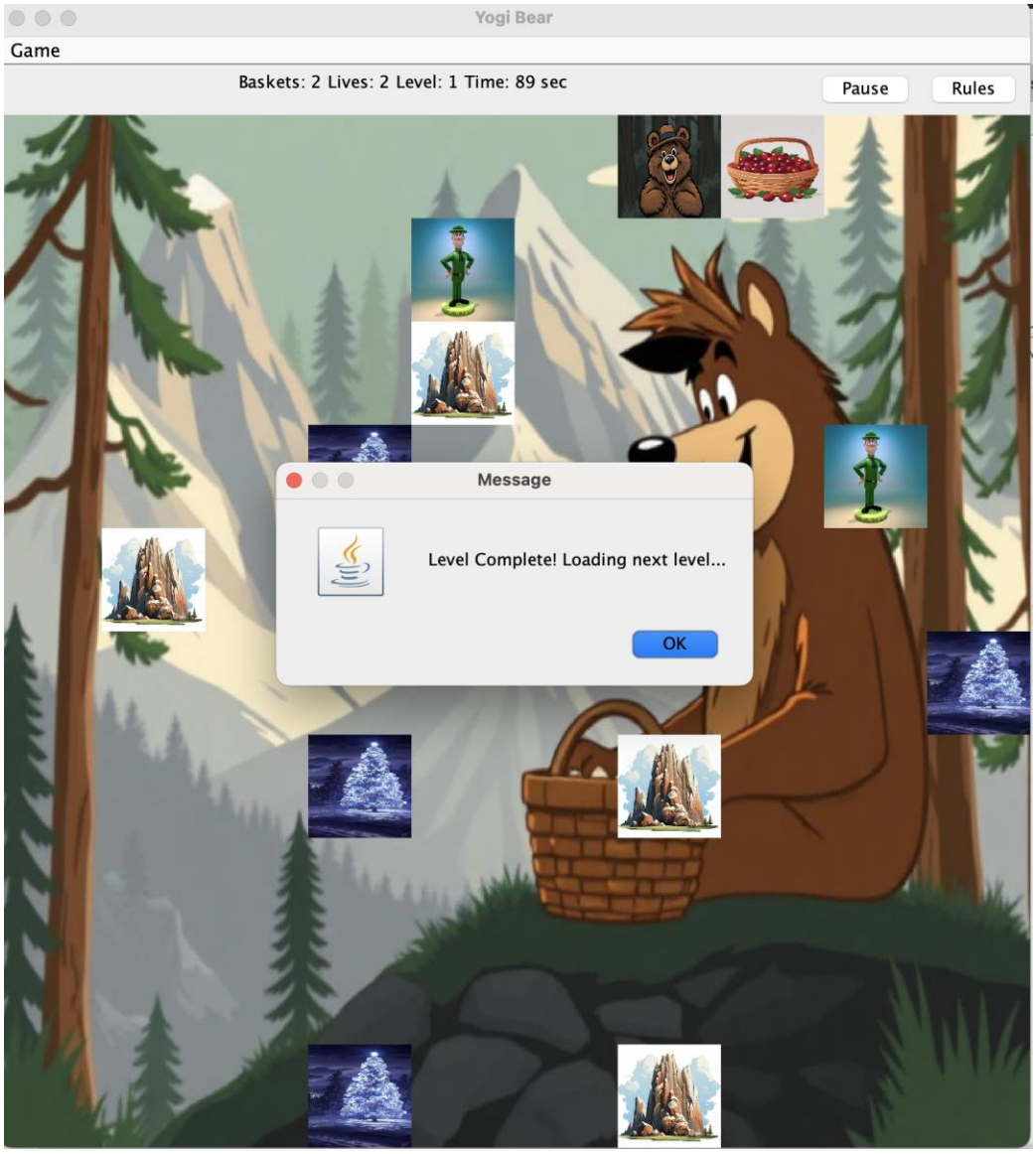
AS a user

I WANT TO complete the level

GIVEN Yogi has collected all the baskets in the current level

WHEN Yogi collects the last basket

THEN the game should stop, display a "Level Complete" message, and load the next level.



5. Game Over Condition

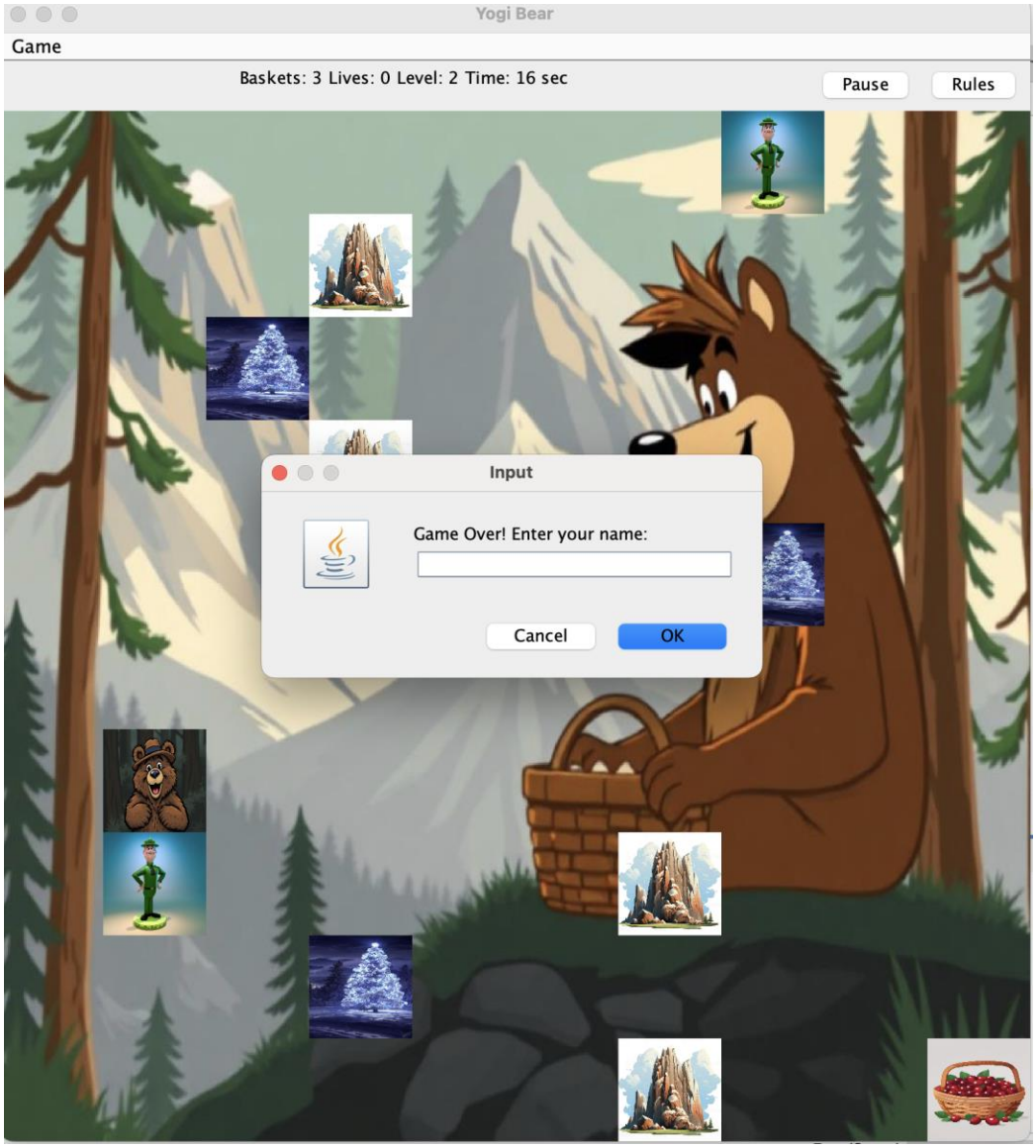
AS a user

I WANT TO end the game when Yogi loses all lives

GIVEN Yogi has no lives left

WHEN the number of lives reaches 0

THEN the game should stop, display a "Game Over" message, and allow the user to enter the name to save score



6. High Scores Table Display

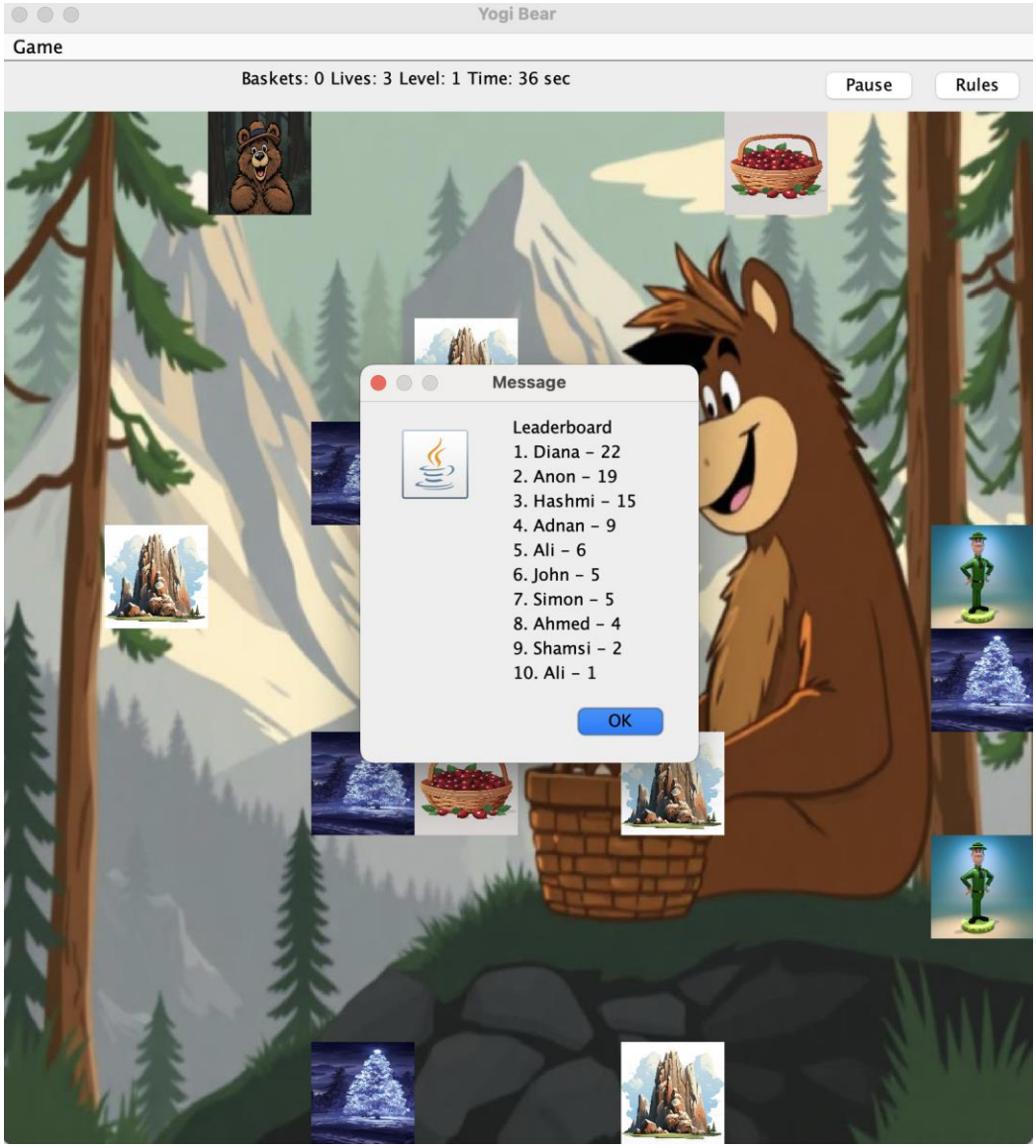
AS a user

I WANT TO view the high scores

GIVEN the game is running and the main menu is visible

WHEN I click on the "High Scores" menu item

THEN the high scores table should appear, displaying the top scores with player names and their corresponding scores.



7. Rules Display

AS a user

I WANT TO view the game rules

GIVEN the game is running and the main menu is visible

WHEN I click on the "Rules" button

THEN a message dialog should appear displaying the game rules with instructions for playing the game.