Programming Technology

Name: Murad Huseynov

Neptun ID: RX15MW

Third Assignment: TASK 1.

Yogi Bear

Yogi Bear wants to collect all the picnic baskets in the forest of the Yellowstone National Park. This park contains mountains and trees, that are obstacles for Yogi. Besided the obstacles, there are rangers, who make it harder for Yogi to collect the baskets. Rangers can move only horizontally or vertically in the park. If a ranger gets too close (one unit distance) to Yogi, then Yogi loses one life. (It is up to you to define the unit, but it should be at least that wide, as the sprite of Yogi.) If Yogi still has at least one life from the original three, then he spawns at the entrance of the park.

During the adventures of Yogi, the game counts the number of picnic baskets, that Yogi collected. If all the baskets are collected, then load a new game level, or generate one. If Yogi loses all his lives, then show a popup messagebox, where the player can type his name and save it to the database. Create a menu item, which displays a highscore table of the players for the 10 best scores. Also, create a menu item which restarts the game

Class Diagram
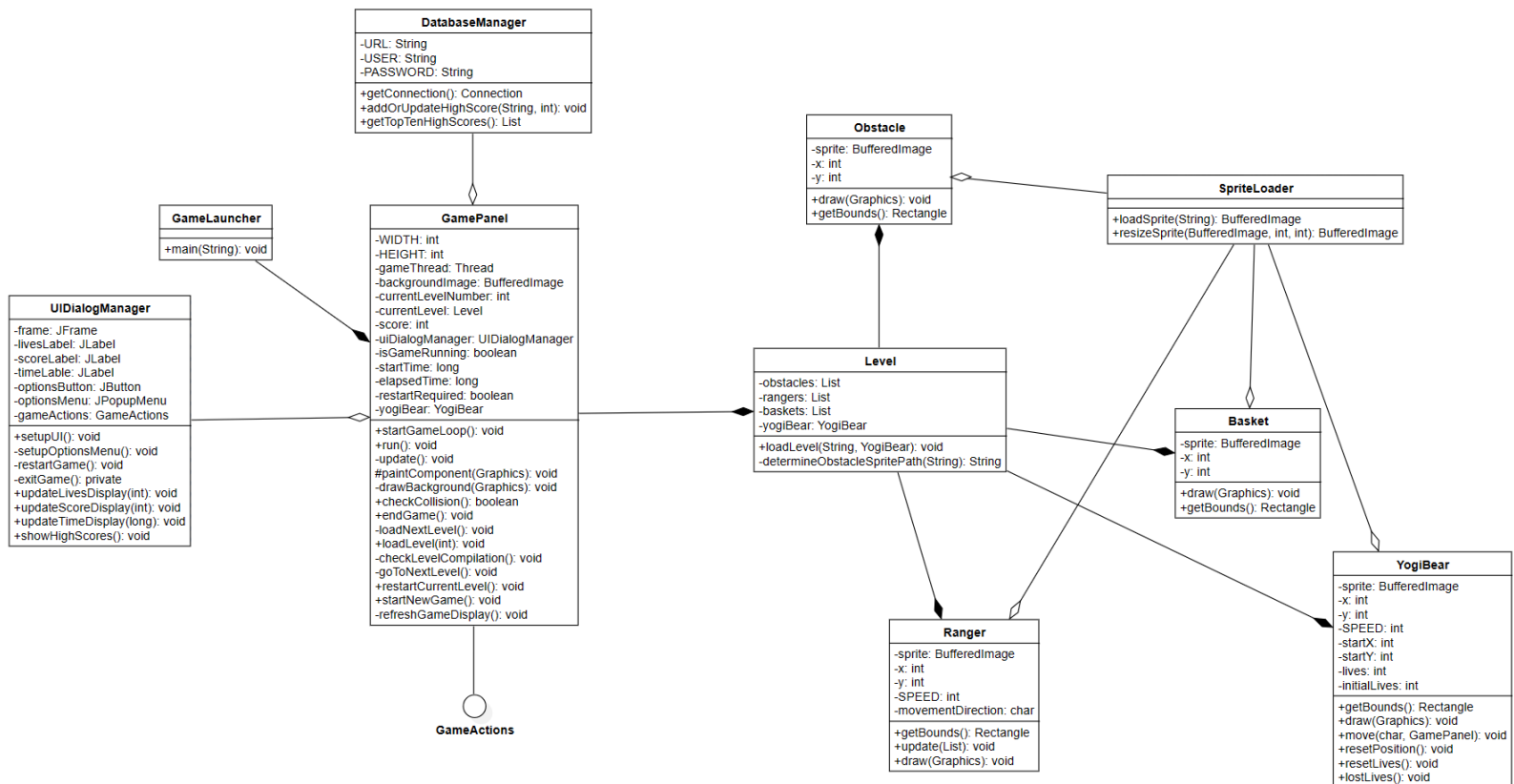
Plan

The game's architecture is built around several key Java classes, each with specific responsibilities to provide a smooth and interactive gaming experience.

- GameLauncher is the entry point of the game, responsible for initializing the main game window and starting the game.
- GamePanel serves as the control center for the game's graphics and mechanics. It manages the game loop, processes user inputs, and updates the game's state. Important methods in this class include run(), which contains the game loop, update(), which updates game logic, and paintComponent(), which handles all the drawing on the screen.
- Level class handles loading and managing the game levels from files. It organizes game elements such as obstacles, rangers, baskets, and the player's character, YogiBear. The loadLevel() method is crucial as it populates the level with these elements.
- YogiBear, the player's character, is managed by its own class which controls its movement, drawing, and interaction with other game elements. The move() method updates the position based on player input, and loseLife() manages the player's life count.
- Obstacle, Ranger, and Basket classes represent different elements within the game. Obstacle defines static objects like trees, Ranger defines moving enemies with update() method adjusting their position, and Basket represents collectible items.
- DatabaseManager handles data persistence, storing and retrieving high scores from a database. The methods addOrUpdateHighScore() and getTopTenHighScores() are used to manage the high scores list.
- UIDialogManager takes care of the game's user interface outside the main gameplay area, such as displaying lives, scores, and menus. Methods like updateScoreDisplay() and updateLivesDisplay() update the UI in response to game events.
- SpriteLoader is a utility class that loads and resizes images (sprites) used in the game, ensuring that visual elements are displayed correctly.
- The interface GameActions defines actions like starting a new game or restarting the current level, providing a contract that the GamePanel class fulfills.

JUnit Tests

1) LevelTest.java ➜ testLoadLevel(); Tests if a level is successfully loaded with all of the components
2) RangerTest.java ➜ testHorizontalMovement(); tests if a ranger moves horizontally successfully
   testCollosionWithObstacles(); tests if the collision of ranger and obstacles is successful
3) YogiBearTest.java ➜ testInitialPosition(); tests if the initial position of yogi bear in each level is successfully place
   testInitialLives(); tests if the initial number of lives of yogi bear is 3

1) The game starts with level1.txt game design and yogi bear has 3 lives throughout the game, 0 scores (baskets) initially and the timer starts from counting from 0:

2) Each basket is worth 1 score and every time the yogi bear collects a basket the Scores bar increases. Once all the baskets within a level are collected, the game proceed with a new level
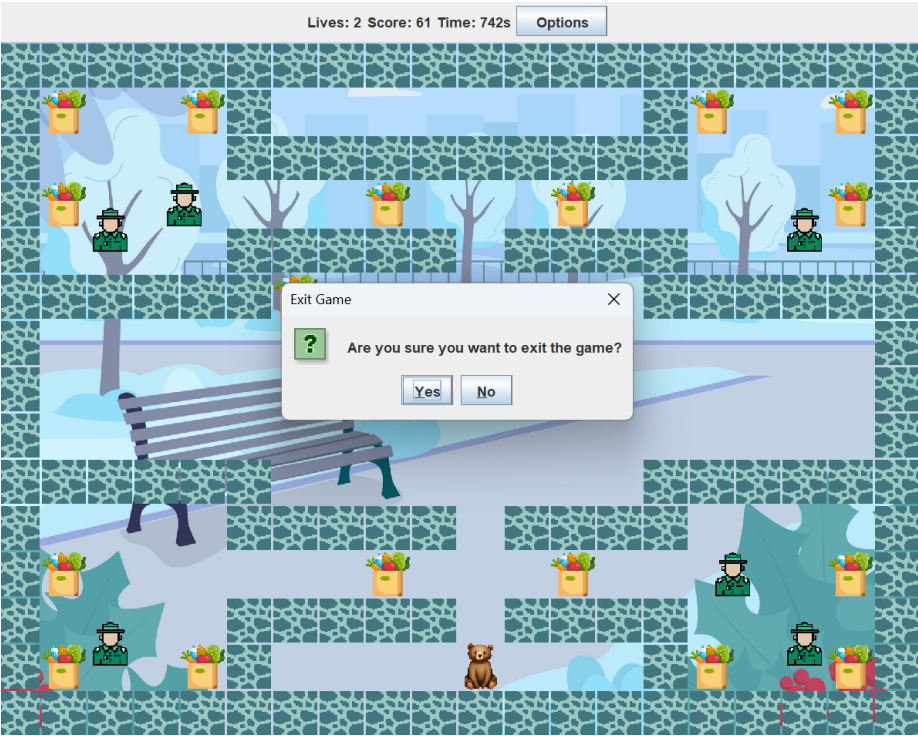
3) Trees are obstacles, neither yogi bear, nor rangers can bypass them. When a ranger collides with a tree, it moves back in the opposite direction. If yogi bear collides with a ranger, its life decreases by one.

4) The Options button leads to for options: New Game – Starts the game from the beginning, Restart – restarts the current level, High Score – displays the high score table that shows the top 10 high scores (Name and Scores), Exit – exits the game, but before asks for a confirmation.

Lives: 2 Score: 61 Time: 688s [Options]

High Scores

Name	Score
bbbbb	78
lalalal	53
abla	11
murad	11
aa	10
bb	10
dddd	6
llllll	2
bllablla	1

Lives: 2 Score: 61 Time: 742s [Options]

Exit Game

Are you sure you want to exit the game?

[Yes] [No]

5) If either yogi bear runs out of lives or the last level (level10.txt) is completed successfully, the game finishes and the game asks for the name of the player to store the score in the table.