Tampere University of Applied Sciences

Snake Game

Sudikshya Bhattarai

BACHELOR'S
May 2024

CONTENTS

1 Features

1.1 Implemented Features

The Snake game feature includes:

1. **Grid of 20x20 Squares:** The game interface consists of a grid layout comprising 20 rows and 20 columsns, providing a structured environment for the snake and other game elements.

2. **Snake Movement with Timer Events:** The snake's motion is regulated by timer events, ensuring consistent and fluid movement throughout the game. Timer events trigger periodic updates to the snake's position, simulating its continuous traversal across the grid.

3. **Snake Growth to Target Length:** Upon initiation, the snake begins with a length of one square and gradually grows to its target length of five squares. As the game progresses and the snake consumes food items, its length increases accordingly.

4. **Apples Spawn Randomly:** Three apples are randomly spawned on the grid at first. When one apple is eaten again it respawns at random position, other remaining in original position. Each apple eaten by the snake increases the player's score.

5. **Scoring System:** The game interface displays the player's current score, reflecting the total number of points earned through apple consumption. Additionally, the lengths of the snake (both current and target) are continuously updated in real-time, providing valuable feedback to the player.

6. **Obstacles:** Obstacles are stationary objects placed on the grid that the snake must avoid. Obstacles are displayed as solid blocks on the grid. If snake head collides with thee obstacles game stops.

7. **Border Collision:** The snake collides with the borders of the grid, preventing it from moving outside the playable area. When the snake collides with a border, the game ends, and the player's score is displayed.

8. **Restart Button:** Players can restart the game by clicking the "Restart Game" button. Clicking the button resets the game state, including the snake's length, score, and position.

1.2 Features Missing

- Graphical representation.
- Game pause mechanism.
- frontend implementation should get via the API offered by the server.

1.3 Use of AI tools

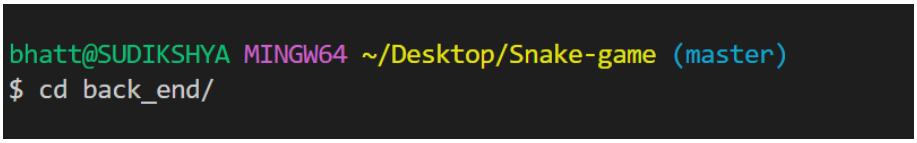
Chat-gpt has been used to understand and implement some of the features.

2 Backend Server Instructions

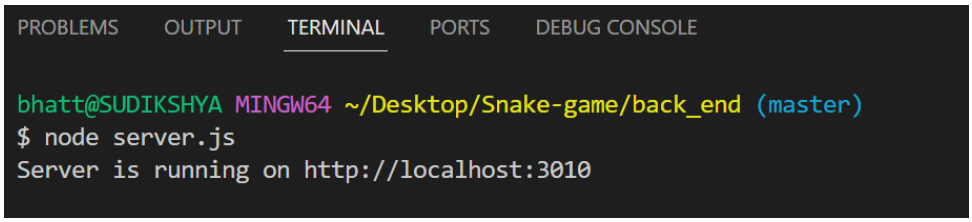
To facilitate the deployment and execution of the Snake game, a backend server has been provided. To run the backend server, follow these instructions:

1. **Open Terminal or Command Prompt:** Launch the command line interface on your system.

2. **Navigate to Backend Directory:** Use the **cd** command to navigate to the 'back_end' directory of the Snake game project.
   `cd back_end/`



3. **Run Server:** Execute the following command to start the backend server:
   `node server.js`



4. **Access Game:** Once the server is running, open a web browser and navigate to the following URL:
   http://localhost:3010

   This will direct you to the Snake game interface.

3 Working Hour

Task Description	**Hour (approx.)**
Analysing required features in project and creating project structure and files	10
Create grid and basic layout in HTML and CSS	8
Implement the snake's initial position and rendering, handling basic snake movement	8
Implement collision detection with the grid borders, with snake itself and added game over conditions.	10
Ensuring apple do not appear on snake or obstacles while respawning	5
Randomly generating apple and growth on snake eating an apple	8
Implement obstacles and handle collision on obstacles	9
Implement apple respawn logic and adding restart features	12
Adding score increasing and snake length logic	8
Implementing CSS on game and information page also	8
Backend Implementation and testing	8
Testing of implemented features and fixing bugs	18
Final testing and documentation	8
Total Hour	120