

Snake Game Design-Specs

Manos Vourgourakis

1. Game Features

1.1 Customization Options

- **Number of Food Items:** The player can select the number of food items (1-10) that appear on the screen.
- **Game Speed:** The speed of the game can be set to "Slow," "Regular," or "Fast," affecting how quickly the snake moves.
- **Snake Color:** The player can choose the snake's color from a range of options (Red, Orange, Yellow, Green, Blue, Purple, Pink).
- **Food Color:** The player can choose the color of the food items from a range of options (Red, Orange, Yellow, Green, Blue, Purple, Pink).
- **Game Mode:** The player can choose between two game modes:
 - **Classic Mode:** The game plays as the original Snake game.
 - **Obstacles Mode:** Obstacles are randomly placed on the grid, making the game more challenging.

1.2 Gameplay

- **Controls:** The player controls the snake using the arrow keys (Up, Down, Left, Right).
- **Snake Movement:** The snake continuously moves in the current direction. The player can change its direction using the arrow keys. The snake's head moves first, and its body follows.
- **Food:** The snake eats food when its head collides with a food item. Each time the snake eats food, the player's score increases by 1, and the snake's length increases by one segment.
- **Obstacles (optional):** In Obstacle mode, random obstacles are placed on the grid. If the snake's head collides with any obstacle, the game ends.
- **Collisions:** The game ends if the snake collides with:
 - The walls of the game area.
 - Itself (by running into its body).
 - Obstacles (in the Obstacles mode).

1.3 Game Loop

The game progresses in a loop that includes the following steps:

1. **Clear Canvas:** Every frame, the game canvas is cleared to prepare for new drawings.
2. **Draw Background:** A light blue checkerboard background is drawn on the canvas.

3. **Draw Snake:** The snake is drawn on the canvas, with its body segments being displayed in the selected color.
 4. **Draw Food:** The food items are drawn on the canvas in the selected color.
 5. **Move Snake:** The snake's head moves in the chosen direction, and the body follows.
 6. **Check Collisions:** The game checks if the snake collides with walls, itself, or obstacles (if in Obstacle mode).
 7. **Update Score:** The score is updated each time the snake eats a food item.
 8. **End Game:** The game ends if any of the collision conditions are met.
-

2. Game Flow

2.1 Start Screen

- **UI Elements:**
 - A welcome message ("Welcome to Snake Game!").
 - Input fields to adjust the number of food items, game speed, snake color, food color, and game mode.
 - A "Start Game" button that triggers the start of the game.
- **Player Interaction:**
 - The player selects the game settings from dropdown menus and input fields.
 - The player clicks "Start Game" to begin playing.

2.2 Game Screen

- **UI Elements:**
 - A score display showing the current score.
 - A canvas where the game is rendered.
 - The game environment (background, snake, food, and obstacles if applicable).
- **Player Interaction:**
 - The player uses the arrow keys to control the snake.
 - The game continues until the snake collides with an obstacle, wall, or itself.

2.3 Game Over Screen

- **UI Elements:**
 - A message indicating the game is over ("Game Over :(").
 - The final score.
 - A "Play Again" button that reloads the page and restarts the game.
- **Player Interaction:**

- The player can view their final score and choose to play again by clicking the "Play Again" button.
-

3. Art and Aesthetics

3.1 Art Style

- **Background:** The background consists of a light blue checkerboard pattern, with alternating shades of light blue (#ADD8E6 and #B0E0E6).
- **Snake:** The snake is represented by square segments drawn on the canvas. The player can customize the color of the snake from a list of options.
- **Food:** Food items are also square and can be drawn in a custom color selected by the player.
- **Obstacles:** In the Obstacles mode, obstacles are randomly placed squares on the canvas in black.

3.2 User Interface

- **Fonts:** The game uses Arial, a sans-serif font for readability and clarity. The font size for the score is 24px, and the font size for general text is 12px.
-

4. Technical Specifications

4.1 Languages and Tools

- **HTML5:** Structure and markup of the game page.
- **CSS3:** Styling for layout, buttons, and UI elements.
- **JavaScript:** Game logic, event handling (keyboard input), and rendering on the canvas.
- **Canvas API:** Used for drawing the game elements (snake, food, obstacles) and handling game rendering in real-time.

4.2 Canvas Dimensions

- **Canvas Size:** 400px x 400px
- **Box Size:** 20px (each segment of the snake, food, and obstacles).

4.3 Game Variables

- **Snake Movement:** Managed by tracking the position of the snake's head and updating its direction each frame.
- **Collision Detection:** The game checks for collisions with walls, the snake's own body, and obstacles.
- **Food Generation:** Food items are placed randomly on the grid, and the snake grows when it consumes food.
- **Obstacles:** Obstacles are randomly placed if the game mode is set to "obstacles."

4.4 Event Handling

- **Keyboard Input:** Arrow keys are used to control the snake's direction.
- **Start/Restart:** The game starts upon the "Start Game" button press and can be restarted with the "Play Again" button.

4.5 Performance

- The game uses `setInterval` to control the frame rate of the game (dictated by the selected game speed). This allows for smooth gameplay at different speeds.
-

5. Game Modes

5.1 Classic Mode

- In this mode, the game follows the classic Snake rules:
 - The snake grows longer as it eats food.
 - The game ends if the snake collides with the walls or itself.

5.2 Obstacles Mode

- In this mode, random obstacles are placed on the game grid.
 - The snake must avoid colliding with these obstacles.
 - The game ends if the snake collides with an obstacle.
-

6. Monetization (Optional)

- **Ad Revenue:** Ads could be displayed on the start and game-over screens.

- **Premium Features:** The game could include optional cosmetic upgrades for snake and food customization, available for purchase.
-

7. Target Audience

- **Casual gamers** who enjoy retro arcade games with modern twists.
 - **Players who enjoy customizing their gaming experience** through options like speed, color, and difficulty.
-

8. Testing

- **Unit Testing:** Tests should be conducted on key functions such as collision detection, food generation, and score updates.
 - **User Testing:** Playtesting with different configurations to ensure the game is enjoyable and the customization options are easy to use.
-

9. Future Updates

- **Leaderboard:** Implement a leaderboard system to track high scores.
- **Mobile Support:** Optimize the game for mobile devices, adding touch controls for direction.
- **Sound Effects:** Add sound effects for eating food, colliding, and game-over events.