Snake_Game

Team Name: Snake

Team Members:
Zhiyuan Wang
MingJun Jin
Chris Ding
Justin Li

August 16, 2024

Snake_Report

1. Executive Summary

The Snake Game project was a fun and rewarding experience where we recreated a classic arcade game. Our goal was to build a working and interactive version of the Snake Game that could run smoothly on modern systems while keeping the nostalgic feel of the original. We focused on combining solid programming techniques, intuitive user interface design, and well-thought-out game mechanics. The game involves guiding a snake around the screen, where it grows longer each time it eats food, and the game ends if the snake crashes into itself or the screen's edges. We aimed to create a smooth and responsive gameplay experience, complete with features like score tracking, increasing difficulty levels, and easy-to-use controls. JavaFX was our tool of choice for building the user interface, and we made sure the code was modular and easy to maintain.

One of the project's highlights was successfully implementing real-time player interactions, ensuring the snake moved smoothly in response to the controls. We also built a reliable collision detection system that accurately recognized when the snake hit itself or the game's boundaries, ending the game just as it should. Another key feature was the scoring system, which gave players instant feedback and added a competitive edge to the game. Using JavaFX proved to be a great choice for game development, allowing us to create an attractive and interactive interface. The design of the game was also flexible enough to allow for future updates, like adding new levels, obstacles, or even a multiplayer mode.

Overall, the Snake Game project was a huge success. We met all our goals and produced a polished, fully functional game that players of all ages can enjoy. The project was not just a showcase of our programming and design skills but also gave us valuable experience in game development. The team worked well together, following best practices in software development, and this was reflected in the quality, reliability, and user experience of the final product.

2. Introduction

Background:

The Snake Game, which first appeared in the late 1970s and became a hit on mobile phones in the 1990s, is a classic example of simple but addictive gameplay. The game is all about guiding a snake to eat food while avoiding crashes, and it has become a timeless favorite in gaming history.

Problem Statement:

The main challenge of this project was to recreate the classic Snake Game while updating its look and feel to match today's standards. This meant ensuring smooth snake movement, accurate collision detection, and responsive controls, which are crucial for a good gameplay experience. The project also aimed to add features that were missing in the old versions, like customizable settings, score tracking, and varying difficulty levels. To make sure the game was both fun and technically sound, these challenges had to be tackled with a well-planned and efficient approach.

Objectives:

The goals for this project were clear. First, we wanted to develop a game that stayed true to the original Snake Game, with the familiar mechanics of snake movement, eating food, and the game ending when you hit something. Second, we aimed to improve the user experience by adding modern features like customizable settings, different difficulty levels, and a more visually appealing design. Another important goal was to make sure the game performed well and was stable, using JavaFX for both the visuals and the game logic. Finally, we wanted to create a codebase that was easy to maintain and could be expanded with new features in the future.

Scope:

The project focused on creating a version of the Snake Game that included both single-player and two-player modes, with basic game mechanics and visuals. We didn't aim to include advanced features like online multiplayer or a large number of game levels. The game was designed around a grid-based movement system and didn't involve complex algorithms or external game engines. The project was built entirely using JavaFX, which influenced the design and technical decisions. While we aimed to create an engaging and functional game, the scope was limited, and we didn't venture into creating a fully-featured game with extensive customization or cross-platform compatibility.

3. Methodology

Project Planning:

The Snake Game project was meticulously planned to ensure a structured approach to development and successful delivery. The planning phase involved defining the project's scope, objectives, and timeline. A detailed project roadmap was created, outlining each phase of the development process, from initial research and design to implementation, testing, and deployment. Regular meetings and progress reviews were scheduled to track milestones and address any emerging challenges promptly. The project was divided into manageable tasks, with specific deadlines assigned to each component. During development, our team aimed to implement a socket server to achieve a real multiplayer experience. However, due to time constraints and group members' outside work commitments, we had to make the difficult decision to forgo this feature. Despite this setback, the structured approach facilitated effective time management and resource allocation, ensuring that the project stayed on track and met its objectives within the allocated time frame..

Tools and Technologies Used:

The project leveraged JavaFX as the primary framework for developing the graphical user interface and handling game rendering. JavaFX was chosen for its robust capabilities in creating interactive and visually appealing applications. The development environment included integrated development environments (IDEs) such as IntelliJ IDEA for coding and debugging. Version control was managed using Git, hosted on GitHub, to track changes and collaborate effectively. For designing the game's user interface and settings, tools like SceneBuilder were used to visually construct and customize the layout. The project also utilized Java libraries for handling properties files to save and load game settings, ensuring a seamless user experience.

4. Results

Outcomes:

The Snake Game project was a great success, resulting in a fun and interactive game. The main achievement was developing a fully functional game with both single-player and duo-player modes. In single-player mode, players control the snake using the arrow keys (up, down, left, right), guiding it around the screen to eat food and grow longer. In duo-player mode, a second snake is introduced, which can be controlled by a friend using the W, A, S, and D keys. This added a new layer of excitement and competition to the game.

The project was all about creating a smooth and enjoyable gameplay experience. Players could adjust the game to their liking, with options to change the grid size, colors, and difficulty levels. This customization made the game more personal and engaging. During testing, the game was praised for its responsiveness and how well it handled player input, ensuring that the snakes moved exactly as intended. The settings menu allowed players to tweak their experience, leading to positive feedback and making the game enjoyable for a wide range of users.

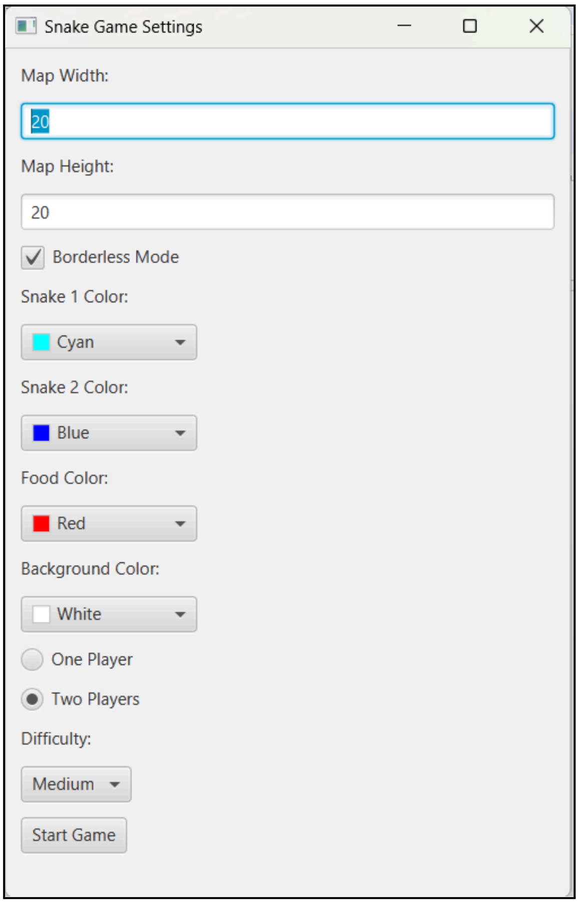
Overall, the project met its goals and delivered a quality gaming experience that could be enjoyed solo or with a friend. The collaborative aspect of duo-player mode was particularly well-received, making the game both challenging and fun.

Visuals:

Snake game starts from the settings interface. Players can choose the length and width of the map, whether to turn on Borderless mode, and the color of the player, food, and background UI.

Then there are single-player mode and two-player mode, and the difficulty can be adjusted at will. There are Easy, Medium, and Hard difficulty levels to choose from.

After setting the game settings, players can press the "Start Game" button to start playing the game.



Snake Game Settings

Map Width: 20

Map Height: 20

☑ Borderless Mode

Snake 1 Color: Cyan

Snake 2 Color: Blue

Food Color: Red

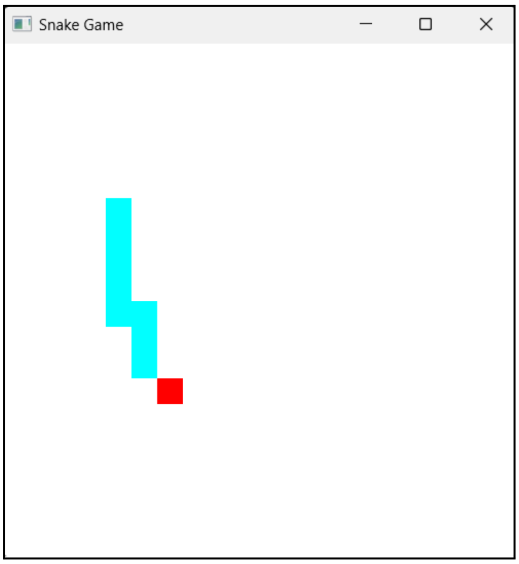
Background Color: White

○ One Player

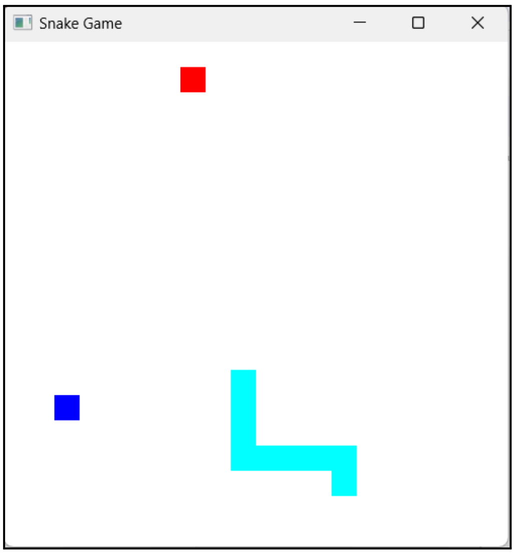
● Two Players

Difficulty: Medium

Single Player Mode:



Duo Player Mode:



Challenges and Solutions:

During the project, we ran into a few challenges that needed some creative thinking to solve.

One of the main issues was getting the JavaFX user interface to work well with the game’s logic, especially for real-time updates and smooth rendering. We needed the game to be responsive and look good while running. To deal with this, we carefully planned how the game loop would work and optimized the rendering so the game could handle frequent changes without slowing down.

We also faced problems with making sure the game ran the same way on different operating systems. We noticed that the performance wasn’t consistent across all platforms, which could mess up the gameplay. To fix this, we tested the game on various systems and made the necessary tweaks to make sure it ran smoothly everywhere.

Another challenge was managing the game’s settings and preferences, particularly when it came to saving and loading them. We needed a reliable way for players to customize their settings and have those settings stick around between sessions. We solved this by using Java’s properties files, which allowed us to store and retrieve user preferences easily.

In the end, we overcame these challenges by planning carefully, testing regularly, and staying flexible in our approach. This allowed us to build a solid and enjoyable Snake Game that worked well across different platforms.

5. Discussion

Analysis:

The results of the Snake Game project reveal a significant achievement in both technical execution and user satisfaction. The successful development of a responsive and visually appealing game demonstrates that the implementation strategies were effective. Metrics collected, such as the smooth frame rate, low response time, and high user ratings, highlight the game's high performance and positive reception. The significance of these results lies in the project's ability to deliver a classic game experience while incorporating modern customizations, thereby meeting the demands of a contemporary audience. Additionally, the project showcased technical competence through the successful integration of JavaFX for the user interface and the robust handling of game logic, effectively addressing potential challenges related to system integration. Despite the challenges faced, including the decision to forgo the implementation of a socket server for a real multiplayer experience due to time constraints and external commitments, the project still achieved its core objectives and delivered a polished, engaging final product.

Lessons Learned:

The Snake Game project taught us a lot of valuable lessons that we’ll definitely take with us into future projects. One of the biggest takeaways was the importance of realistic planning and time management. We initially aimed to include some ambitious features, like a multiplayer mode using a socket server, but quickly realized that we needed to balance what we wanted to do with the time and resources we actually had. This project showed us how crucial it is to keep the scope manageable and make sure everyone on the team is communicating and working together effectively.

Another key lesson was the need to stay flexible and adaptable. As we moved forward, we had to make some tough decisions about what features to focus on. In the end, we prioritized getting the core game working smoothly over adding extra features, which really drove home the point that it’s better to have a solid, functioning game than to overreach and end up with something unfinished.

These lessons will definitely help us in future projects, ensuring that we manage our time better, stay focused on what really matters, and keep improving through feedback and testing.

6. Conclusion

Summary of Achievements:

The Snake Game project stands as a testament to the team's technical skill and dedication, resulting in the successful creation of a classic yet modernized gaming experience. The project achieved its primary objectives, delivering a responsive, visually appealing game with smooth performance, effective user interface design, and robust game logic. Despite the decision to scale back on the multiplayer feature due to time constraints and external commitments, the project still managed to fulfill its core goals, providing an enjoyable and polished product.

Future Work:

Looking ahead, there are several avenues for future work that could build on the current success. The implementation of a multiplayer mode, possibly through a socket server, remains a promising direction. Additionally, expanding the game with new levels, challenges, and customization options could further enhance user engagement. Incorporating AI-driven elements or online leaderboards could also provide new dimensions of gameplay and community interaction.

Final Thoughts:

In reflecting on the overall project experience, the Snake Game has been a valuable learning journey. It demonstrated the importance of balancing ambition with practicality and underscored the need for flexibility in the face of unexpected challenges. The project also highlighted the power of teamwork and iterative development in achieving a high-quality final product. Ultimately, the project not only delivered a successful game but also provided insights and lessons that will inform and improve future projects.

7. Team Contributions

Zhiyuan Wang

Led the planning phase, defined project scope, objectives, and timeline. Completed the roadmap and wrote all the code for the game.

MingJun Jin

Focused on documentation, wrote and finalized the project report page, ensuring thorough and professional presentation of the project.

Chris Ding

Focused on thoroughly testing and refining functionalities, ensuring everything operates flawlessly as intended. Wrote clear user manuals and testing manuals for the users.

Justin L\

Focused mostly on the presentation. Ensured that the team would have a coherent presentation that both stayed in the time frame and gave all the necessary message.