Air University Islamabad

Programming Fundamentals

Semester Project GDD

BSAI-F-24-B

Submitted By:

Name:Ayesha Siddiqa

Roll Number:241419

Submitted To:

Mam Umme Sadima

Sir Tahir Akram

Submission Date:

30/12/2024

Table of Contents

Fairyland Tetris Game Design Document 3
What is Tetris? 3
Tools and Environment 3
What I Used 3
Features I Completed 3
1. Splash Screen 3
2. Login and Signup System 3
3. Level Selection Menu 4
4. Game Theme 4
5. Basic Gameplay 4
6. Simple Functional Tetris 4
Challenges I Faced 4
Final Status 5
User Interface: 5
Simple Tetris: 7
What I Learned 8
Impact of the Game 8
Future Plans 9

Fairyland Tetris Game Design Document

What is Tetris?

Tetris is a simple and fun 2D puzzle game where players move falling shapes called tetrominoes to fit them into rows. When a row is completely filled, it disappears, and the goal is to clear as many rows as possible without the grid filling up.

I wanted to make a Tetris-inspired game called **Fairyland Tetris** with a magical theme. The idea was to create something unique and visually appealing, with a splash screen, login system, and different levels. However, due to limited time and some challenges, I managed to complete a basic version of the game along with a simpler, fully functional Tetris version without visuals.

This project was very helpful for my learning because I explored new tools, libraries, and concepts. It taught me how to build a project using bazel and sfml, handle technical challenges, and manage my time effectively.

Tools and Environment

What I Used

- **Programming Language**: I wrote the game in C++ because it's fast and reliable.
- **Library**: I used SFML (Simple and Fast Multimedia Library) to handle the graphics and game window.
- **Build System**: I used Bazel to build the project. It was my first time using it, and it took me a whole week to figure it out.
- **Compiler**: I used MSVC (Microsoft Visual C++) for compiling the game.
- **IDE**: I chose Visual Studio Code because it's lightweight and easy to use.
- **Version Control**: I used Git to manage my code, and the project is hosted on GitHub.

Features I Completed

1. Splash Screen

I created a splash screen that shows the game title "Fairyland Tetris," along with my name, roll number, class, and university. I also added some motivational messages to make it more engaging.

2. Login and Signup System

I made a login system where users can sign up and log in. For now, only one username and password are hardcoded to test the game. I planned to connect this to a database to save user records, but I couldn't do it due to time limitations.

3. Level Selection Menu

I added a menu where users can pick from six levels. Currently, all the levels are the same because I didn't have time to make them different. My plan was to make some levels locked until the user unlocks them by completing earlier levels.

4. Game Theme

I tried to give the game a magical Fairyland theme. The falling blocks (tetrominoes) are stars, and the score bar is represented by the moon.

5. Basic Gameplay

I implemented a feature where blocks fall every 300 milliseconds. The front-end looks nice, and I added win and loss screens for when the game ends. However, the main game functions like rotating blocks or clearing lines aren't working yet.

6. Simple Functional Tetris

Since the main Fairyland Tetris game wasn't fully functional, I created a simpler Tetris game without visuals. This version works perfectly and includes all the standard features like rotating blocks, clearing lines, and scoring.

Challenges I Faced

1. **Learning New Tools**
    - I had never used Bazel before, and it took me a whole week just to learn how to build my project with it.
    - I also had no experience with SFML, so it took a lot of time to understand how to use it.
2. **Time Constraints**
    - I had so many ideas for the game, but there wasn't enough time to implement everything. For example, I couldn't make the levels different or connect the login system to a database.
3. **Technical Issues**
    - While I managed to make the blocks fall, other important features like rotating blocks, clearing lines, and scoring aren't working in Fairyland Tetris.

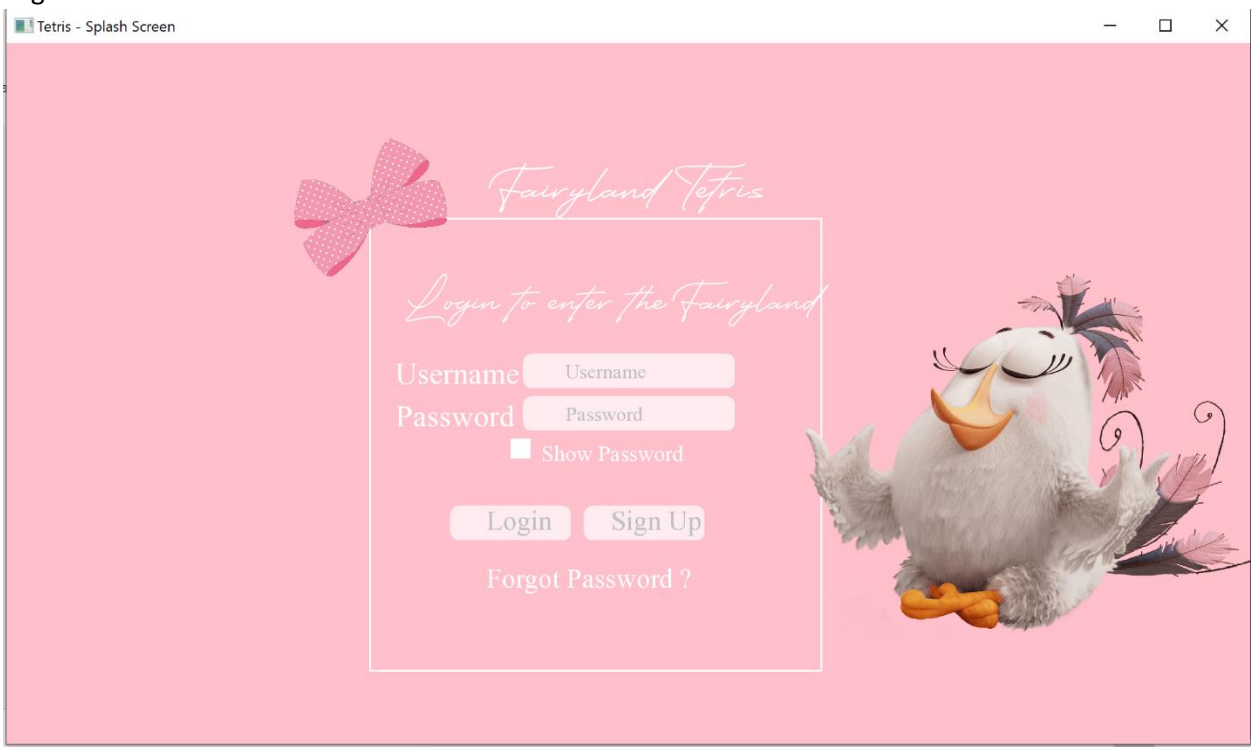
Final Status

- **Fairyland Tetris**: The game looks good with the splash screen, login system, and level selection, but it's not fully playable yet.
- **Simple Tetris**: I made a basic Tetris game that works perfectly, but it doesn't have any graphics or themes.

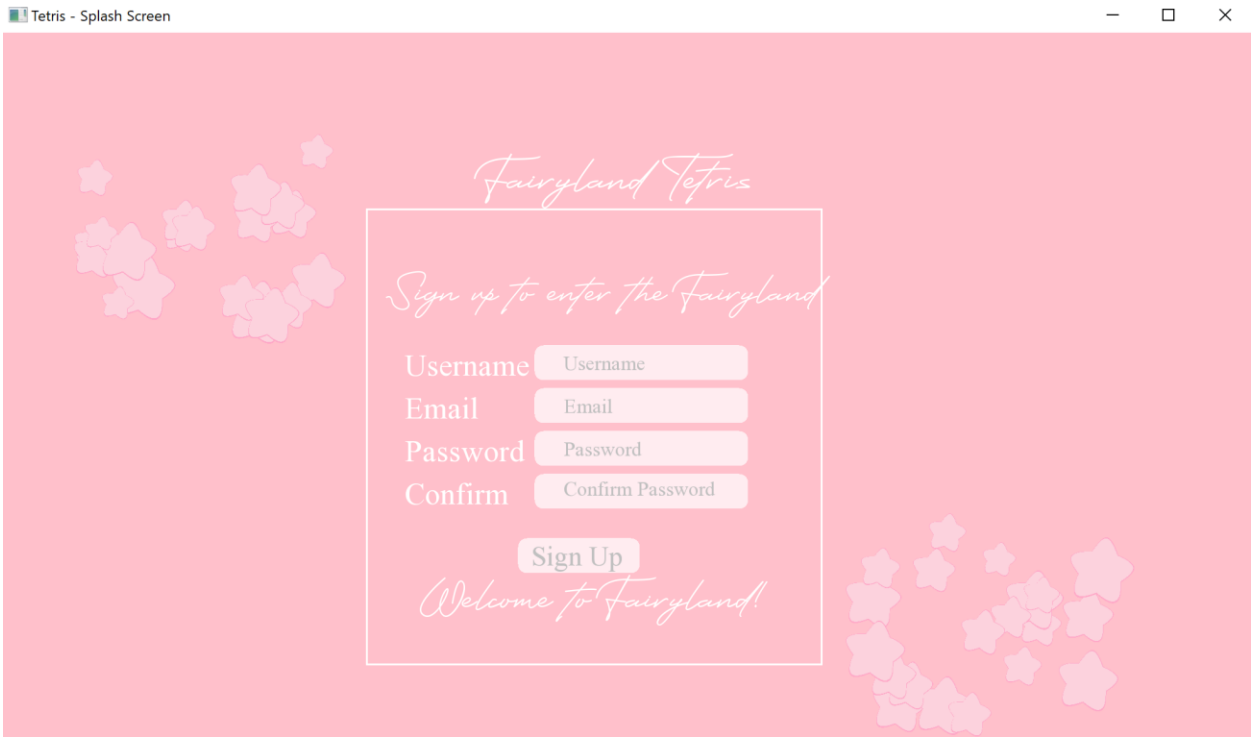
User Interface:

Splash screen:

Login screen:

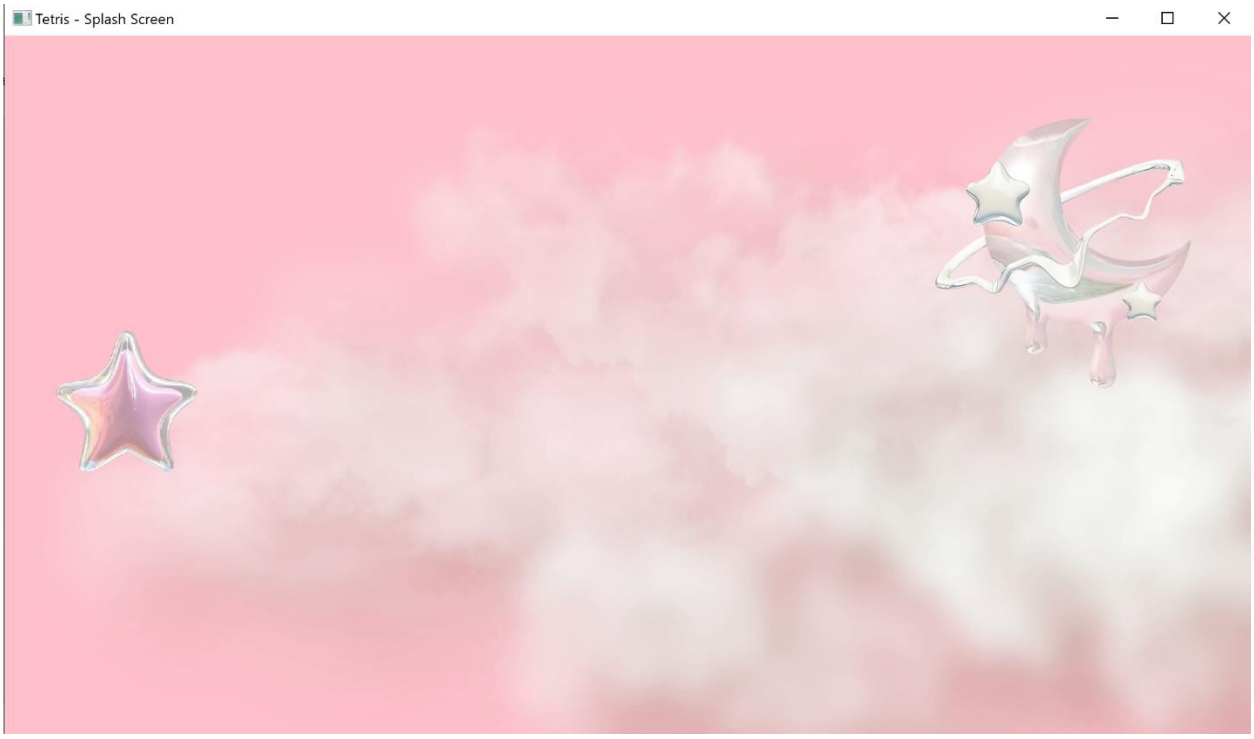


Signups screen:

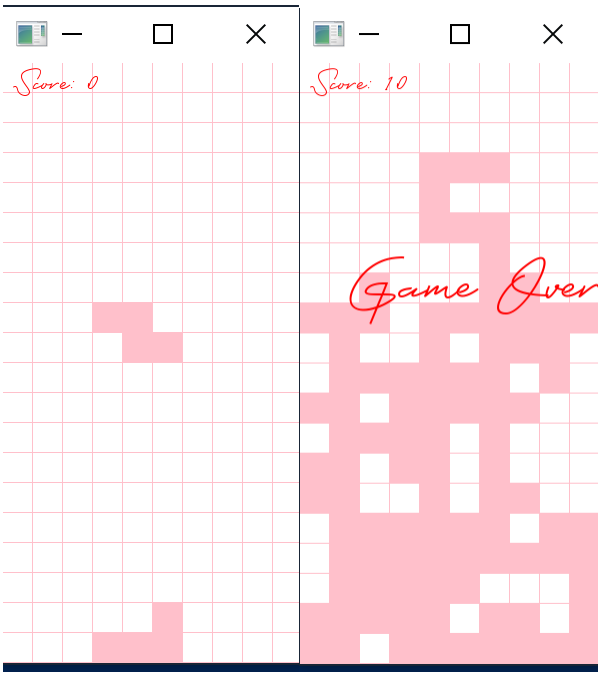


Levels menu:

Game running state:



Simple Tetris:



What I Learned

This project taught me a lot. I learned how to use SFML for creating graphics and how to build projects using Bazel, which was completely new to me. I also understood the importance of planning and time management, especially when working on larger projects. Debugging and problem-solving became key skills I improved during this process.

I also realized how important it is to break tasks into smaller, manageable steps. Although I couldn't implement everything I wanted, I'm happy with what I accomplished and the knowledge I gained.

Impact of the Game

This game can be helpful for improving problem-solving and quick-thinking skills. Tetris is known for enhancing focus and spatial awareness, and my fairyland version adds a magical theme that makes it more fun and engaging. Even though it's not fully functional yet, the idea of combining education and entertainment in games can be impactful and inspiring.

Future Plans

If I get more time, I want to:

1. Make each level unique and progressively harder.
2. Add features to lock and unlock levels.
3. Fix the main gameplay in Fairyland Tetris by adding rotation, line clearing, and scoring.
4. Connect the login system to a database to save user information.
5. Add more Fairyland-themed graphics to make the game look magical and fun.