

Query Quest

Project Documentation

- Course: CIS4951 – Capstone II
- Semester: Spring 2026
- Instructor: Seyedmasoud Sajadi
- Team Members:
 - Areesh Chowdhury
 - Victoria Vilar
 - Katherine Castilblanco

Table of Contents

Table of Contents.....	2
Executive Summary.....	3
Problem & Requirements.....	4
System Overview & Architecture.....	5
Implementation Notes.....	7
Security, Privacy, Accessibility & Compliance.....	8
Limitations & Future Work.....	9
Installation.....	10
User Manual.....	13
Scrum Evidence Index (links to Sprint Planning, Dailies, Reviews, Retros).....	17
Poster (36"×48" horizontal), Slides, and Video Links (Intro, Demo).....	19

Executive Summary

Many students attending school and beginners who want to get into programming only really learn about the basics of programming languages, such as Java and Python. There are almost no resources for younger audiences to learn about Database Management and SQL. There are very few games, and they might not captivate the attention of the younger audience.

Our solution was to make an interactive platformer that people can enjoy. It is to supplement learning basic SQL, to help them memorize the topics they have chosen to learn.

Problem & Requirements

Functional Requirements:

- Level Progression
- Enemy interactions
- Collision Detection
- Quiz Mechanics
- Menus
 - Main Menu
 - Pause Menu
- Player Movement
 - Jump
 - Attack
 - Interact

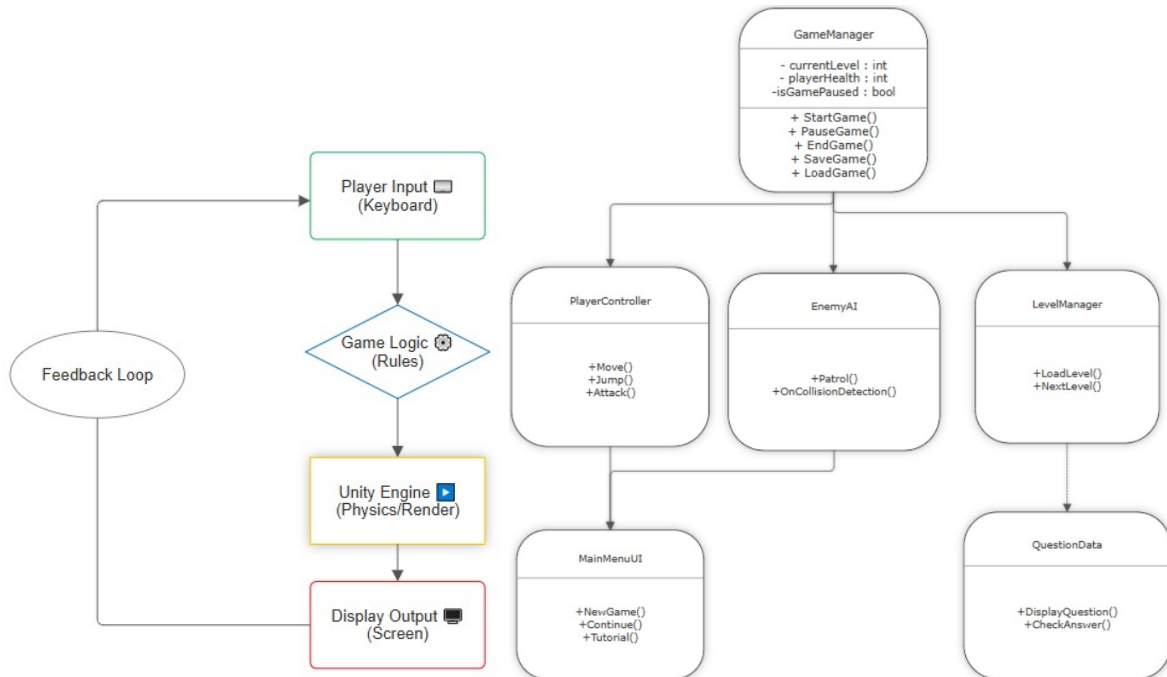
Non-functional Requirements:

- Save/load

Include a prioritized backlog (top 10–15 items) and a link to your full backlog board.

- User Story 6: Figure out how to incorporate SQL questions to the platformer
- User Story 7: Learn how to push the code from unity to GitHub
- User Story 8: Design levels to create for the game
- User Story 10: Create notes around each level to help teach the user about SQL
- User Story 13: Add a game over screen
- User Story 14: Add a pause button
- User Story 15: Add checkpoints
- User Story 16: Create the quiz screen
- User Story 17: Create the notes screen
- User Story 18: Add a Main Menu Screen

System Overview & Architecture



Core Technologies

- **Unity (2D Game Engine)**
 - Handles rendering, physics, scenes, animations, and game loop
- **C#**
 - Used for scripting gameplay logic (player, enemies, UI)
- **Unity Physics Engine**
 - Collision, gravity
- **Unity UI System**
 - Canvas, Buttons, Text
- **Scene Management System**
 - Levels
- **Input System**
 - Keyboard controls

Development Tools

- **Visual Studio**
Used to write and debug C# scripts
- **Unity Editor**
Used for designing scenes, prefabs, and game objects

Data/Storage Overview

- PlayerPrefs (Unity)
 - Stores:
 - Unlocked Levels
 - Player progress
 - Local storage on the user's machine

Data is stored locally using Unity's PlayerPrefs system for saving game progress and settings.

API Surface

This project does not utilize APIs such as OpenAPI or GraphQL. All of the game logic and data processing are handled locally within the Unity engine.

Implementation Notes

Repository Structure

- Organized by Unity's default project layout:
 - Assets/ – scripts, scenes, prefabs, UI elements
 - Scenes/ – main menu, levels, tutorial
 - Scripts/ – core gameplay logic (player, enemies, UI)
 - Settings/ – engine configuration

Coding Conventions

- Written in C# using object-oriented principles
- Consistent naming:
 - Classes: PascalCase (PlayerController)
 - Variables: camelCase (playerSpeed)
- Methods are modular and focused on single responsibilities

Notable Design Patterns

- Component-Based Design
 - Scripts attached to GameObjects
- Event-Driven Behavior
 - Uses Unity lifecycle methods (Start(), Update(), collisions)

Tradeoffs

- Used PlayerPrefs instead of a database for light data storage
- No backend reduces the complexity, but it also limits scalability
- Focused more on making modular scripts for an easier debugging process and future expansion.

Security, Privacy, Accessibility & Compliance

Summarize how the solution addresses ethical, legal, and societal impacts; include security, privacy, and accessibility considerations.

Security

- Game runs locally within Unity
- No use of external servers or APIs, which reduces the risk of data breaches

Privacy

- No personal user data is collected
- All game data is stored using PlayerPrefs
- No account creation, tracking, or online data storage

Accessibility

- Keyboard-based input
- Clear UI design for easy navigation
- A tutorial area is added to help users know what the controls are

Societal Impact

- Helps students learn about SQL in a more engaging way

Ethical Impact

- Designed to provide an engaging and fair gameplay experience
- Encourages problem-solving and interactive learning

Limitations & Future Work

Limitations:

Only 10 levels that cover a limited number of topics.

Lack of personal assets.

Future Work:

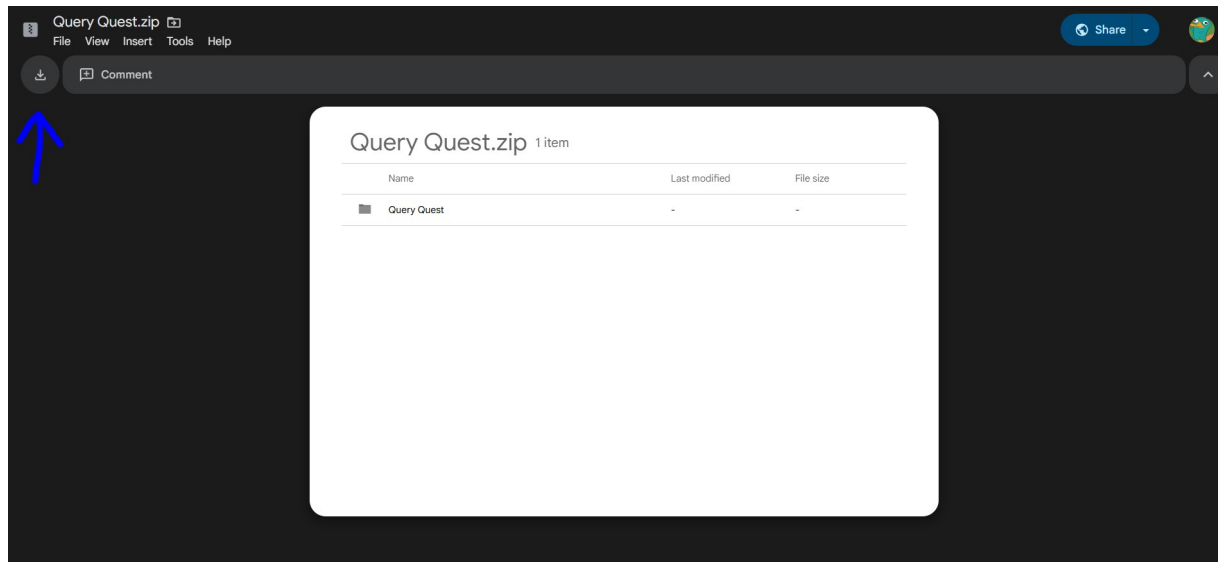
In the future, we would like to create more levels as well as have our own personal assets to use for the whole game.

Installation

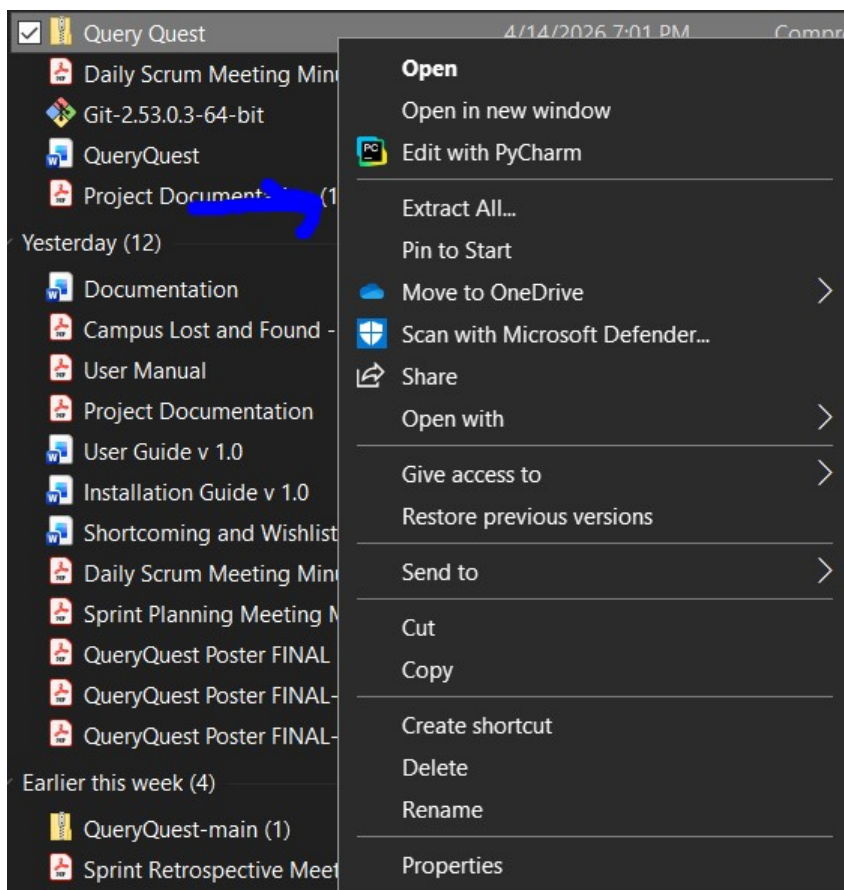
Click or copy and paste the link below and it will send you to a zip file

<https://drive.google.com/file/d/1tWskWJio0kn0syJjW7Gmf0aW1FO5V9xK/view?usp=sharing>

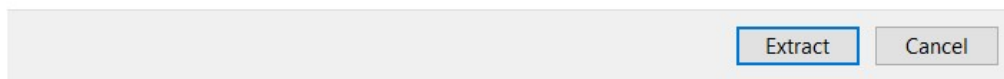
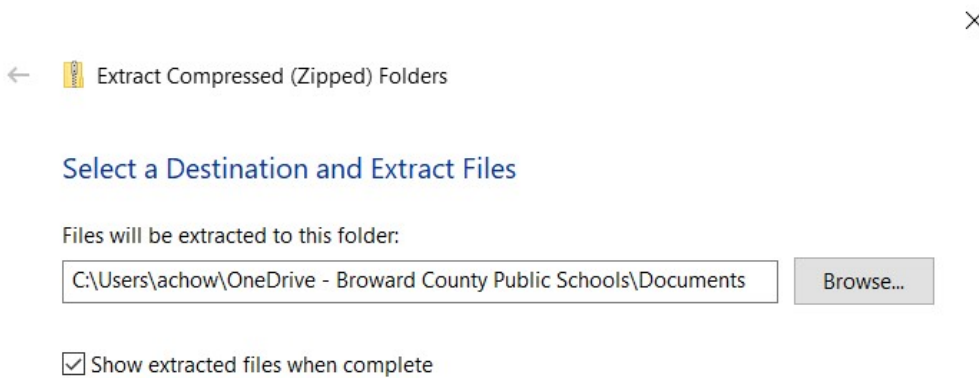
It will take you to this file to where you will need to download it



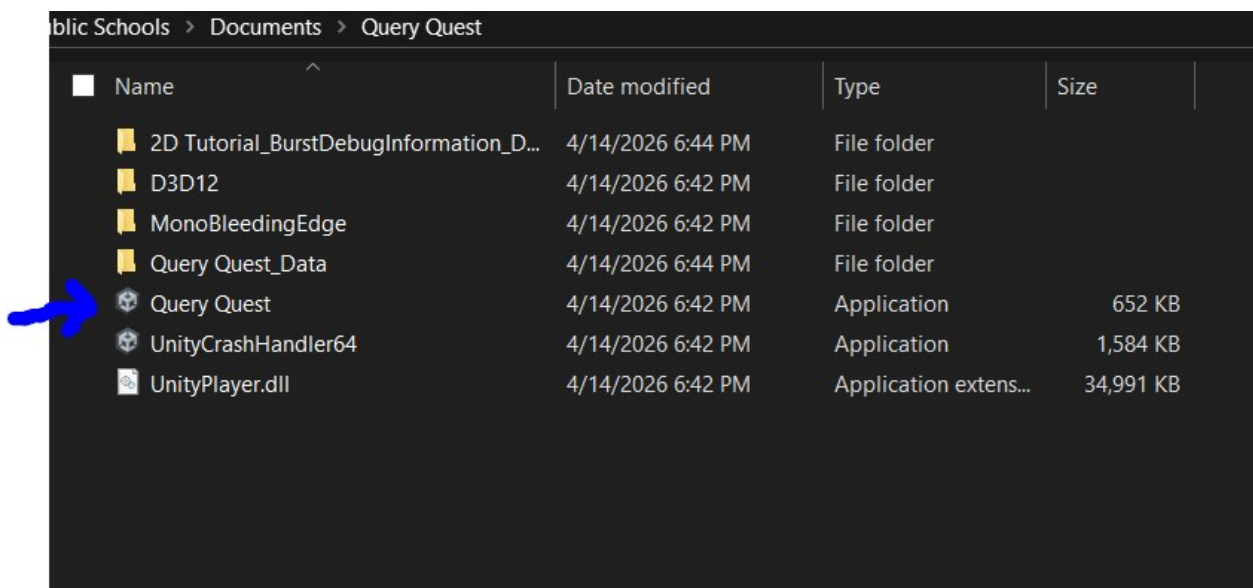
Click on Extract All



Choose the location and click on extract



Click on the application named Query Quest when opening the Query Quest Folder



Enjoy the game



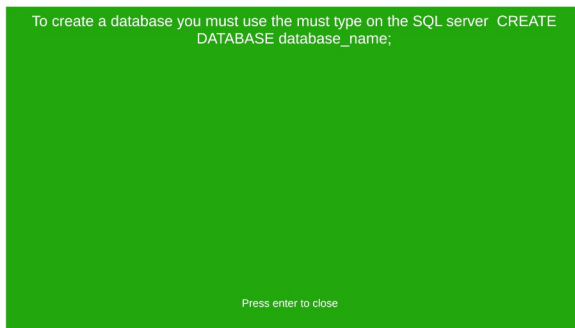
User Manual

Menu Interactions

To interact with the main menu, pause menu, level complete menu, you use “E” to select your option.



Press “ENTER” to close the notes menu.

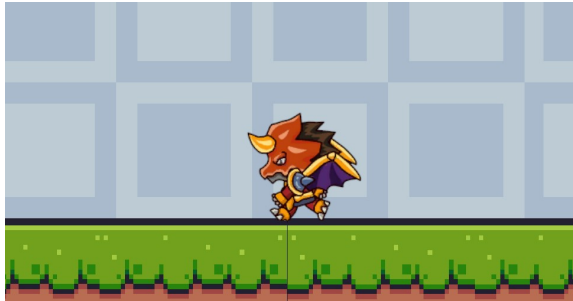


Use “UP” and “DOWN” arrow keys to highlight your answer choice. Press “ENTER” to select your answer on the quiz menu.



Character Movement

Use “A” to make the character move left.

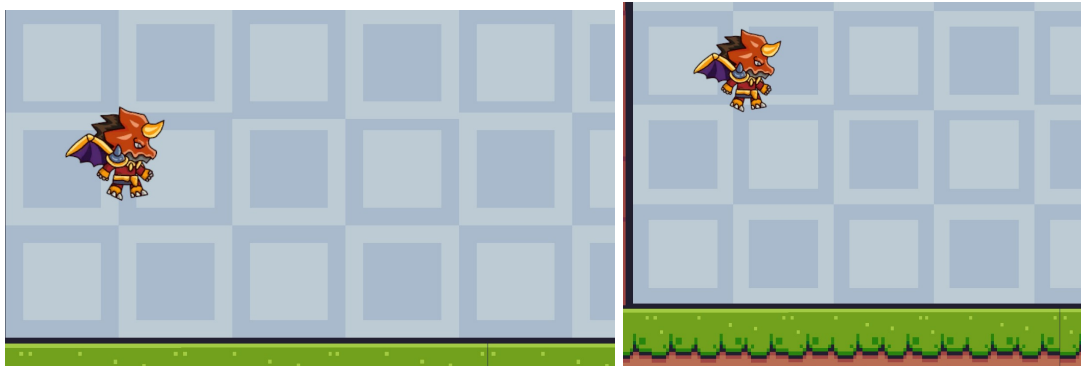


Use “D” to make the character move right.

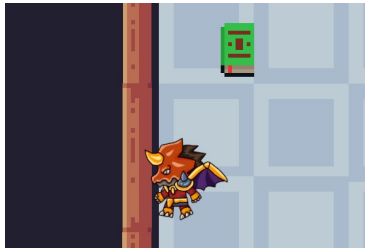


Use “SPACE BAR” to make the character jump.

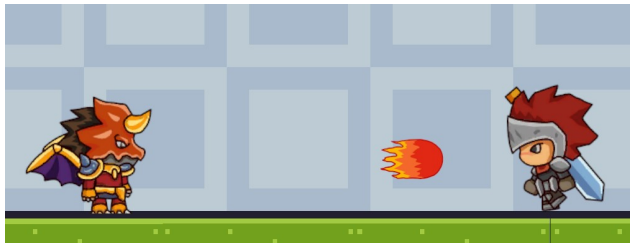
Double tapping the “SPACE BAR” makes the character double jump.



You can also jump off of walls.



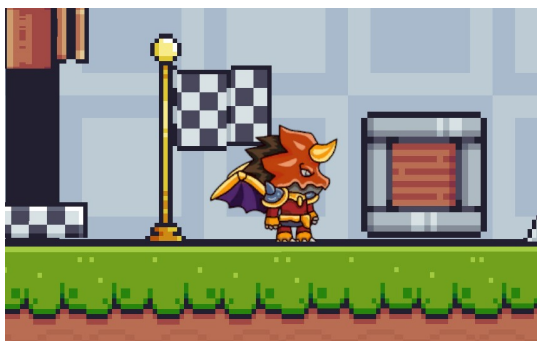
Use “Left Mouse Click” to shoot out fireballs.



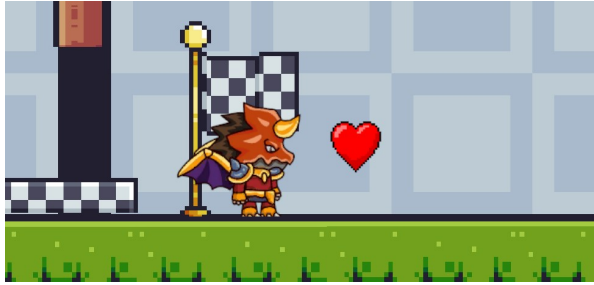
Use “E” to interact with the notes and quiz notebooks.



Walk into the flag poles to get a checkpoint.



Walk into hearts to regain health.



To complete a level, press “E” to interact with the trophy. This will take you to the level completed menu.



Scrum Evidence Index (links to Sprint Planning, Dailies, Reviews, Retros)

Link to where all Scrum Documents are Located:

https://drive.google.com/drive/folders/1sWXXu1o3Upe4m0o0qcHJAahA7EK_I3wq

Sprint 1:

- Sprint Planning Meeting 1
- Daily Scrum Meetings 1-9
- Backlog Grooming 1
- Sprint Retrospective Meeting 1
- Sprint Review Meeting 1

Sprint 2:

- Sprint Planning Meeting 2
- Daily Scrum Meetings 10-19
- Backlog Grooming 2
- Sprint Retrospective Meeting 2
- Sprint Review Meeting 2

Sprint 3:

- Sprint Planning Meeting 3
- Daily Scrum Meetings 20-29
- Backlog Grooming 3
- Sprint Retrospective Meeting 3
- Sprint Review Meeting 3

Sprint 4:

- Sprint Planning Meeting 4
- Daily Scrum Meetings 30-39
- Backlog Grooming 4
- Sprint Retrospective Meeting 4
- Sprint Review Meeting 4

Sprint 5:

- Sprint Planning Meeting 5
- Daily Scrum Meetings 40-49
- Backlog Grooming 5
- Sprint Retrospective Meeting 5

- Sprint Review Meeting 5

Sprint 6:

- Sprint Planning Meeting 6
- Daily Scrum Meetings 50-59
- Backlog Grooming 6
- Sprint Retrospective Meeting 6
- Sprint Review Meeting 6

Sprint 7:

- Sprint Planning Meeting 7
- Daily Scrum Meetings 60-69
- Backlog Grooming 7
- Sprint Retrospective Meeting 7
- Sprint Review Meeting 7

Poster (36"×48" horizontal), Slides, and Video Links (Intro, Demo)

Slides: https://drive.google.com/file/d/1vqfHAc3_wKtmA8flrmbnbgO63MCWSOzU/view

Poster: https://drive.google.com/file/d/1QQXcFI1CI7a2e5WbH_u0Yo7eh71efwL/view

Video Link for Intro: <https://youtu.be/R9ja6nq19YU>

Video Link for Demo: <https://youtu.be/JkTW7-vGZLw?si=0CmjuYwu9qXNRn2j>