

Query Quest



Continue

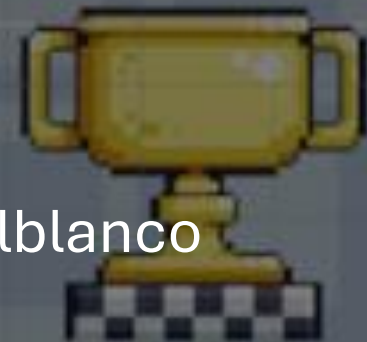
New Game

Tutorial

Query Quest



Areesh Chowdhury, Victoria Vilar, Katherine Castilblanco



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:

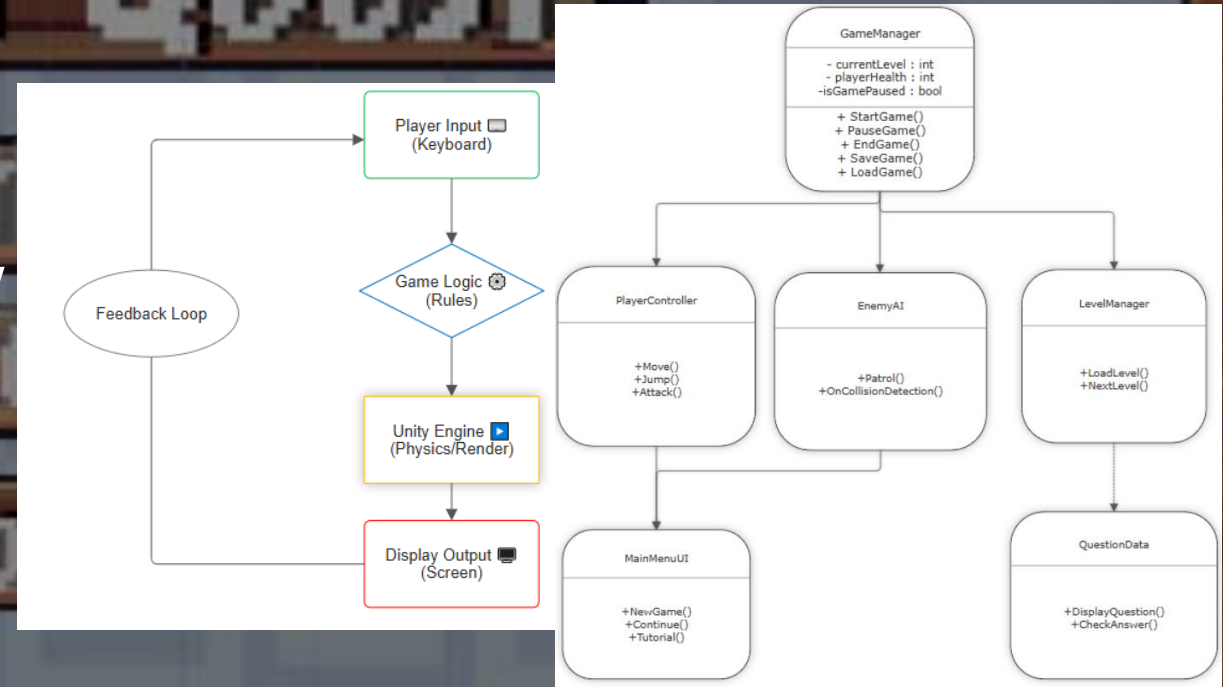
- Save/load



Important User Stories

- 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



Continue

- **Visual Studio**

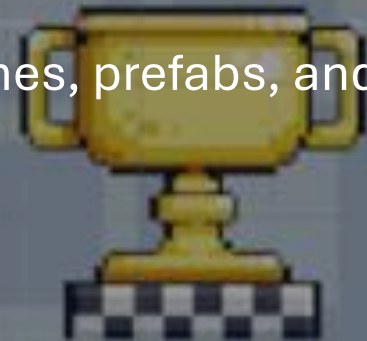
Used to write and debug C# scripts

New Game

- **Unity Editor**

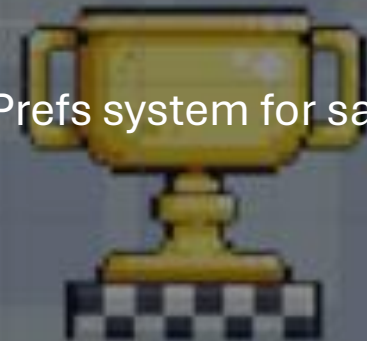
Used for designing scenes, prefabs, and game objects

Tutorial



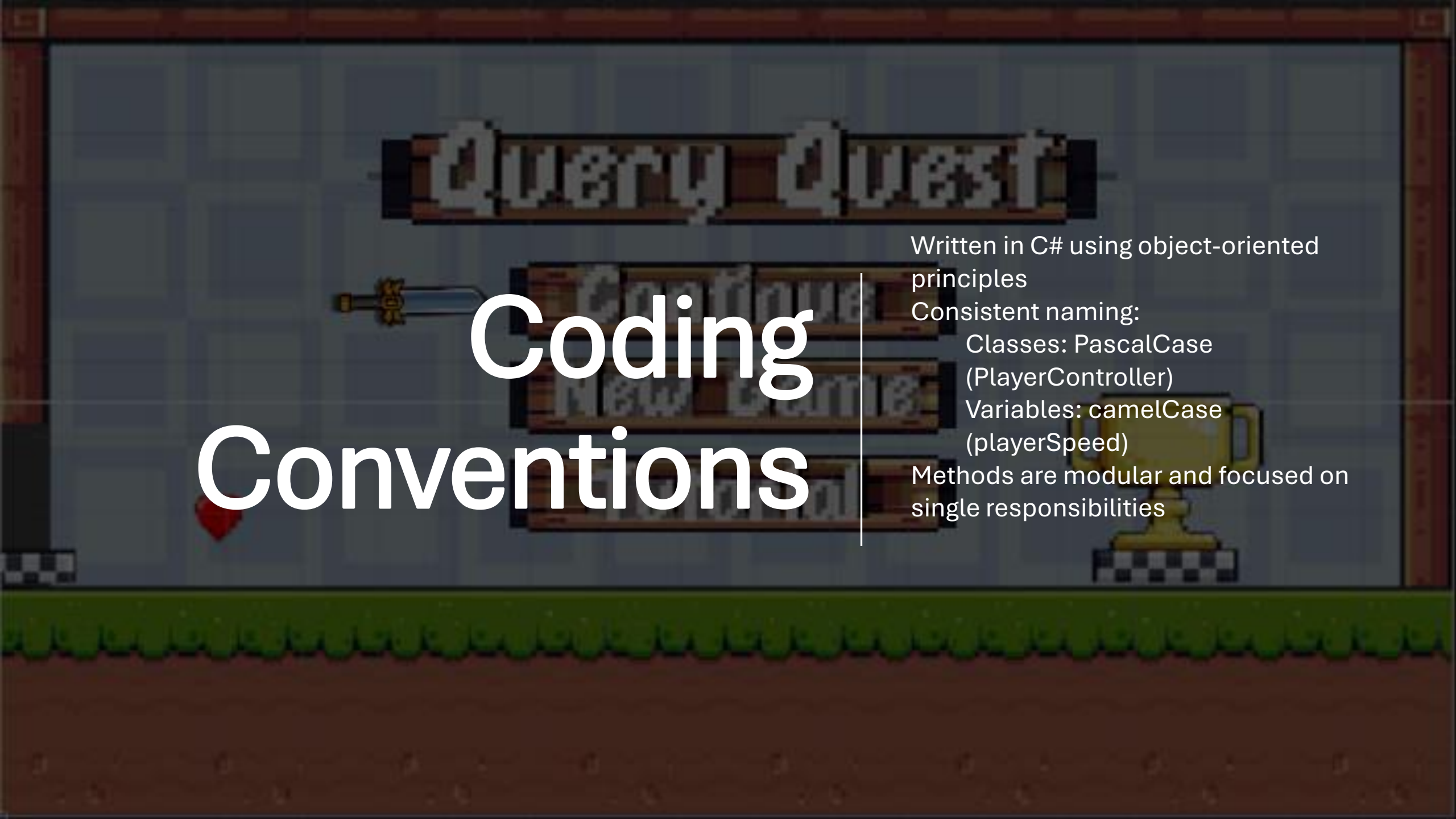
Data/Storage Overview

- 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.



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

- 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

Query Quest



Continue

New Game

Tutorial



Accessibility

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



The background is a pixelated Super Mario Bros. title screen. It features a blue sky with a light blue checkered pattern, a brown brick border at the top, and a green grassy ground at the bottom. In the center, there are three wooden signs stacked vertically: 'Query Quest' at the top, 'Continue' in the middle, and 'New Game' at the bottom. To the left of the 'Continue' sign is a sword icon, and to the right is a trophy icon. A red heart is visible on the left side of the screen.

Query Quest

Continue

New Game

Helps students learn about SQL in a more engaging way

Ethical Impact



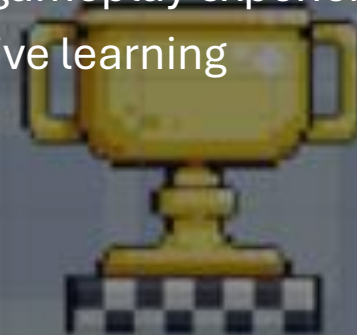
Query Quest

Continue

New Game

Tutorial

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



Limitations and 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.