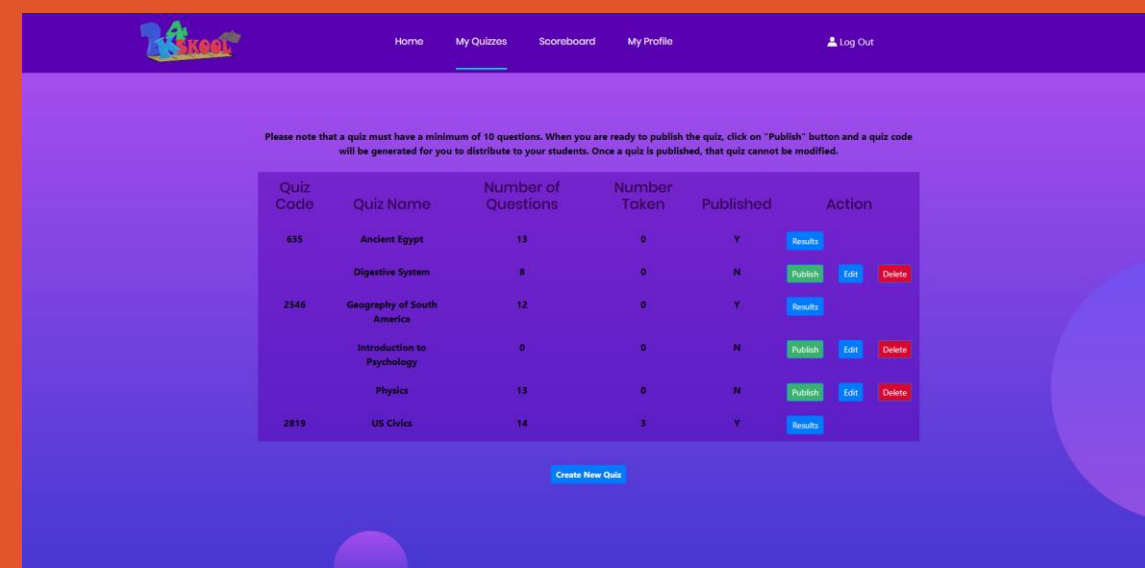
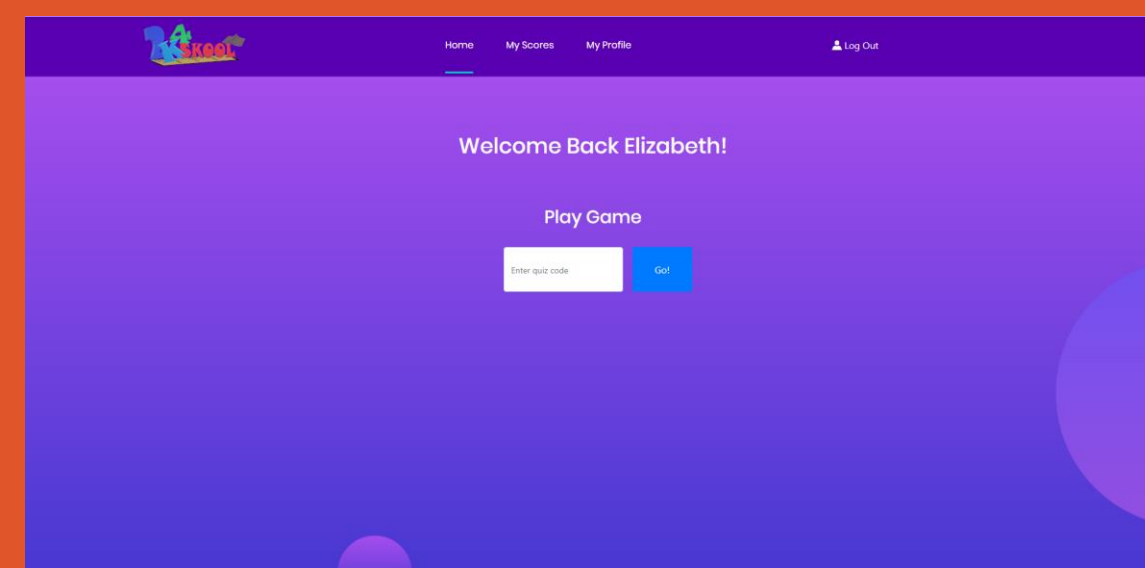


# 2KOOL4SKOOL ESCAPE ROOM QUIZ

There are two end users for this website: teacher and student. The website has a login page where a user can either login with an existing account or create a new one. Once logged in, both users are redirected to their respective home page.



For teachers, they can view recent activities, create and edit quizzes, view quiz scores, as well as edit their profile.



For students, they can enter a code to play the game, view and download their quiz scores, as well as edit their profile.



**Nelson Chan**  
**Thomas Lem**  
**NianJun Shi**

<https://osu-2kool4skool.herokuapp.com/>

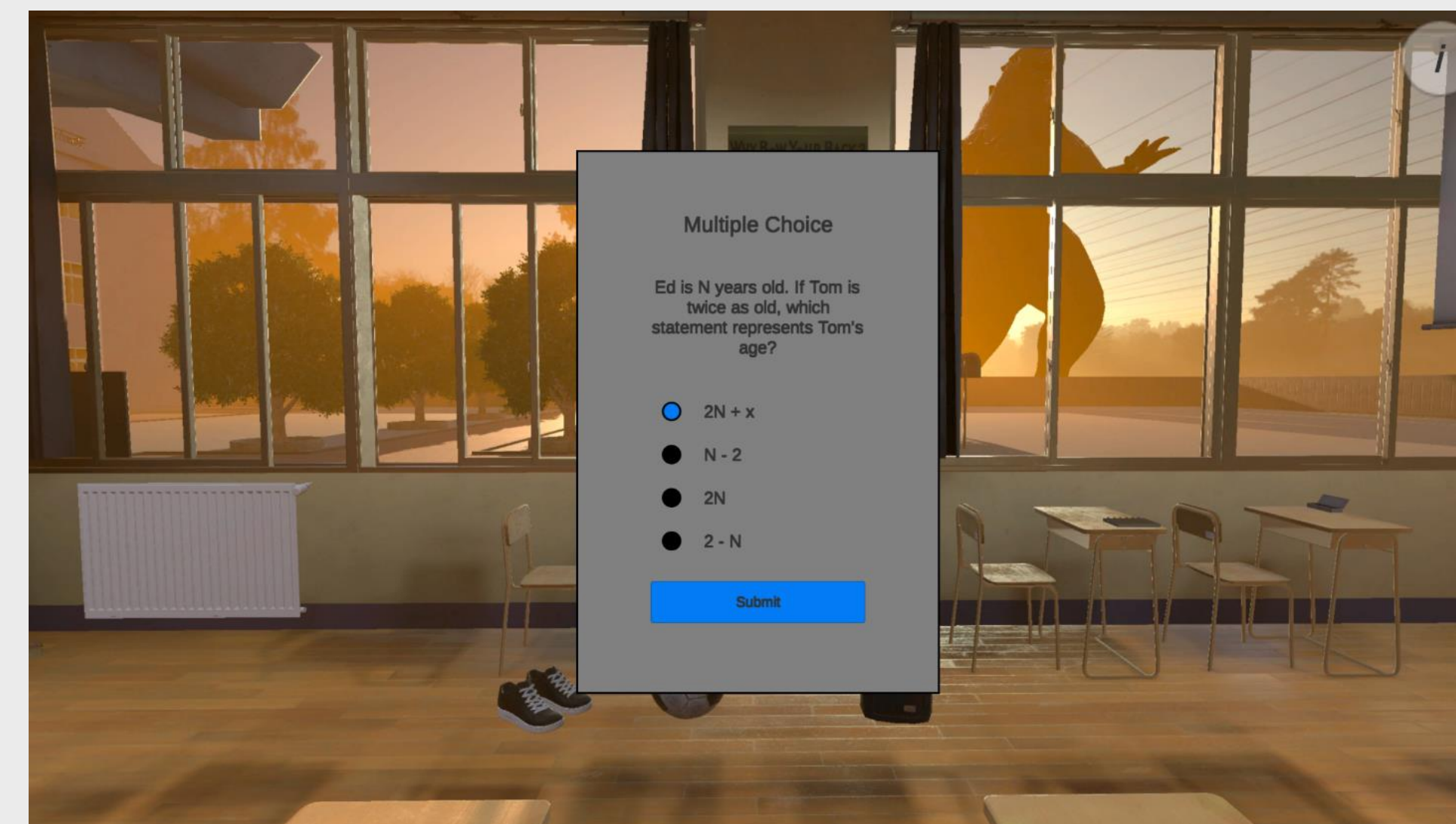


Sends user response  
from Unity to  
backend server

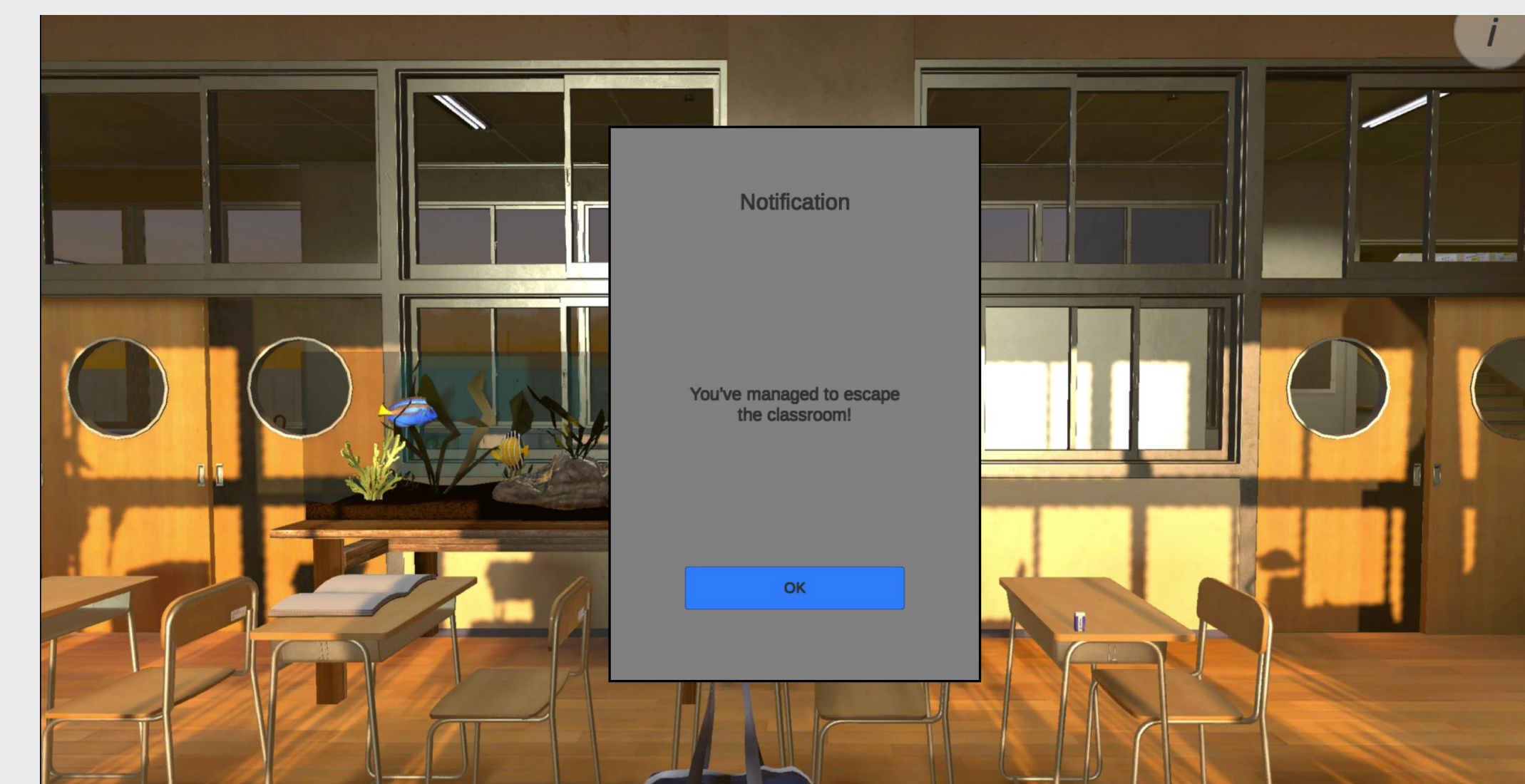
Displays form for user to edit question in Handlebars

Collects and sends user inputs as a query to create account

### Multiple choice question prompt in the game



## Backpack lock puzzle in the game

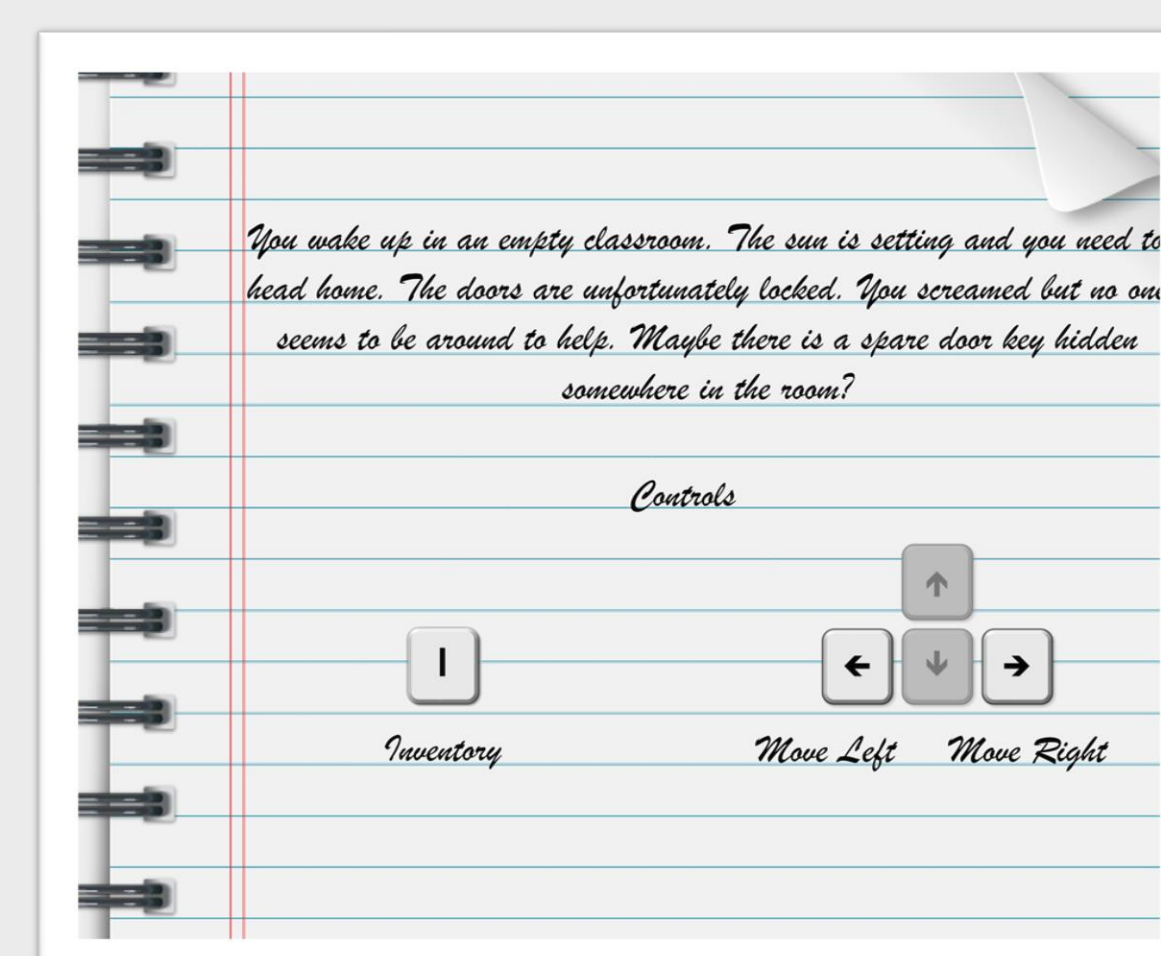


Victory message in the game

## SYSTEM INTERACTIONS

The player plays as a student that fell asleep during a class lecture. The student wakes up and discovers that they are locked in. The player needs to escape the classroom and can do so by answering a set of ten teacher-generated questions.

As the player progresses through the game and interacts with the many different objects, for each object they interact with, they will be prompted to answer a question (True/False, Multiple Choice, or Short Answer). After the question is answered, the player will obtain an inventory item or activate an event. The player can also access their inventory where they can view acquired items, questions remaining and elapsed time.



## MySQL Database ↔ Web Application:

The web-based user interface provides Create, Read, Update, and Delete (CRUD) functionalities for a small database—handled by the server (Heroku with Node.js).

## MySQL Database ↔ Unity Game:

The web application retrieves the questions from the database and sends those to the game by calling a Unity script function through JavaScript. After the user answers the ten questions, the game sends the user's responses to the backend server using Unity's `UnityWebRequest` API. Then, the server sends an insert query to the database.

