

Team Name: Neurologica VR

VDS: Proposal

Product Name: VISUALIZE Data Structures

Team Members: Brendan Chase & Ken Orsini

Website: www.neurologicavr.com

Term: Fall 2021

12-10-2021

Table of Contents

Executive Summary.....	1
Standard Application Runtime Overview.....	1
Application Overview	2
Business Objectives.....	2
User Stories	3
Non-Functional Requirements.....	4
Functional Requirements.....	5
Technology & Terminology	6
Business Processes.....	7
Academic Use Case	7
Free Learning Use Case	9
Models	10
Database Diagram.....	10
Context Diagram	11
Level 1 Data Flow Diagram.....	12
Mockups.....	12
Unity UI Mockup	12
Login Menu	13
Create Account.....	13
Forgot Password	14

Reset Password	14
Verify Email	15
Main Menu.....	15
Lesson Menu (Lesson 1).....	16
Quiz Menu.....	16
Quiz Menu (Early Submission)	17
Log Out Confirmation (Primary).....	17
Log Out Confirmation (Quiz)	18
Unity Application Flow	18
Application Start	18
User Starting Area.....	18
Initial Level Access	19
Lesson Access.....	19
Lessons.....	19
Quiz Access & Submission.....	19
Unity Application Gallery	20
2D Floor Plan (Side).....	20
2D Floor Plan (Top)	21
Portal Room	21
Central Hall.....	22

Starting Account Sign-In / Creation	22
Portals to Lessons	23
Think Deck.....	23
Arrays Lesson: Sub-Lesson 1	24
Arrays Lesson: Sub-Lesson 2	24
Arrays Lesson: Sub-Lesson 3	25
Runtime Spawned Quiz User Interface	25
Conclusion.....	26
Issues.....	26
Results.....	26
References.....	27

Executive Summary

This document serves to host the specifications for the design and implementation of the “Neurologica VR: VISUALIZE Data Structures” application. The purpose of the application is to introduce the concepts of data structures as they relate to computer science through the unique and interactive platform of virtual reality. This solution addresses the lack of education in this topic and other similar topics to students in an academic setting. The application will offer an immersive learning experience for students or free learners, and a supplementary resource for educators to teach and grade their students. The goals of the project include:

- developing the application in Unity for execution on the Oculus Quest platform
- developing a Node.js backend application to manage web requests
- developing a website for the organization.

The Node.js backend application will be used for web requests to read from or modify the database from the unity application. The website will offer information about the application and organization and the ability to download application executables.

Standard Application Runtime Overview

During user account creation, the user will input their information on the Unity client which will be validated upon submission. The user information is then validated by the API to ensure the user email is not in the database. Successful account creation will generate an account on the database using the user account data; sensitive user information like the password will be hashed for user security and protection.

A verification code is generated to verify user email and stored in the database. This code is sent to the user email using MailGun services; this verification code sent to the user is identical to the

verification code on the database. When the user inputs the verification code into the Unity client, the flag, “Email Confirmed”, is set to “true”, which allows the user to successfully log in and operate within the Unity application.

Upon successful login, a JSON Web Token will be generated, sent, and stored on the user object within Unity so that all communication with the database can be validated and verified. This JSON Web Token, must be present within all request to the API for the request to be processed; this ensures that users can not modify their scores or access the database from anywhere but within the Unity client. Once logged in, users will have access to the main areas and features within the application. Users will be able to complete lessons and take respective quizzes.

When a user begins a quiz, the Unity client pulls user quiz data from the database to populate the users quiz history for that specific quiz. On the first attempt, the user will answer quiz questions. The users quiz score is recorded within Unity, and upon submission, is sent to the database to be recorded. The database stores the users quiz attempt and score. At this point, the database records the users first attempt score, second attempt score, average score, and latest attempt score. Additionally, the users quiz score and attempt number is sent to the users email using MailGun services; the quiz score and attempt number sent to the user is identical to the respective values in the database.

Upon successful logout or application quit, the JSON Web Token is destroyed from the user object.

Application Overview

Business Objectives

Key	Summary
VDS-87	Provide free learners with a resource to learn about data structures
VDS-88	Provide students with a unique environment to learn in
VDS-89	Provide educators with a new tool to teach their students
VDS-90	Allow educators to grade their students with the application

VDS-91	Research the use of virtual reality applications for educative purposes
--------	---

User Stories

Key	Summary	Linked Issues
VDS-103	As a User, I want to make an account, so I can save my progress and quiz scores.	VDS-125
VDS-105	As a User, I want to make an account from within the Unity Application, so I don't need to shut down the application.	VDS-125
VDS-107	As an Admin, I want to make changes to the database from my personal terminal via SSH.	VDS-126
VDS-108	As an Admin, I want CRUD operations to the database to be secure, so that data isn't manipulated.	VDS-127, VDS-128
VDS-109	As a User, I want information being stored on the server, so that my progress is saved.	VDS-129
VDS-110	As a User, I want information being stored on the server, so that my quiz scores are saved.	VDS-129
VDS-111	As a User, I want to see my progress information from within the Unity Application, so that I know what I should learn about next.	VDS-129, VDS-130
VDS-112	As a User, I want to be able to initiate a password reset from within the Unity Application, so that I can recover my account information.	VDS-132
VDS-113	As a User, I want to take quizzes from within the Unity Application, so I can better learn from the VR lessons.	VDS-133
VDS-114	As a User, I want to be alerted if a server error occurs in the Unity Application, so that I know the issues are not caused by me.	VDS-134
VDS-115	As a User, I want tips about lessons to be available to me, so that I can better understand the lessons.	VDS-137
VDS-116	As a Student User, I want my quiz results emailed to me after completing lessons, so that I can store or submit them to my teacher.	VDS-138
VDS-117	As a User, I want the ability to skip to sections of lessons after I've viewed them at least once, so I can save time.	VDS-140
VDS-122	As a User, I want to learn about the organization, so that I can understand what they offer.	VDS-145
VDS-123	As a User, I want to learn about the Unity Application from the website, so that I can understand its benefits.	
VDS-124	As a User, I want to download the Unity Application from the website without requiring a registered account, so that I can access it on my computer.	

Non-Functional Requirements

Key	Summary	Linked Issues
VDS-93	The website must contain terms & conditions with standards used for virtual reality applications.	
VDS-94	The unity application must be fully available without the need for user registration.	
VDS-95	The application must maintain a 60-90hz refresh rate when running on the Oculus Quest.	
VDS-97	The server application shall be deployed to and hosted by DigitalOcean	
VDS-98	The unity application shall support the Oculus Quest VR headset and Microsoft Windows operating system.	
VDS-99	Web requests made by the unity application shall handle all relevant HTTP error codes.	
VDS-100	The unity application shall host the same terms & conditions as the Oculus platform it is being developed for.	
VDS-101	The unity application shall host the same health and safety warnings as the Oculus platform it is being developed for.	
VDS-102	Users will be prompted to accept terms upon user registration or using the application without an account.	
VDS-187	All code and unity development processes must be documented and backed up.	VDS-183, VDS-184

Functional Requirements

Key	Summary	Linked Issues
VDS-92	The website must allow users to download the application.	
VDS-125	The system shall allow users to create an account through the Unity Application.	VDS-10, VDS-105, VDS-106
VDS-126	The Database shall allow administrators to make manual changes via a remote terminal.	VDS-107
VDS-127	Manual CRUD operations must be made by administrators only via the Server Application.	VDS-108
VDS-128	Automatic CRUD operations to the Database performed by the User must be contained to functionality within the Unity Application.	VDS-108
VDS-129	The Database shall store Registered Account information such as User ID, Password, Name, Progress, and Scores.	VDS-110, VDS-111, VDS-109
VDS-130	The Unity Application shall present the User with progress information via the UI.	VDS-111
VDS-132	The System shall allow Users to initiate an account password reset via the Unity Application.	VDS-112, VDS-131
VDS-133	The Unity Application shall offer quizzes related to each lesson for Users to complete via a UI panel within the application.	VDS-113
VDS-134	Users shall be alerted of server errors from within the Unity Application.	VDS-114
VDS-136	The Website shall alert Users of server errors.	VDS-135
VDS-137	The Unity Application shall display tips related to the current lesson via a UI panel.	VDS-115
VDS-138	Quiz score information including Score, current attempt number, and attempt resets, shall be generated, and emailed to the User after quiz completion.	VDS-116
VDS-141	Quizzes shall allow for a maximum of two attempts and will keep the highest score of those two attempts.	
VDS-145	The Website shall offer information about the organization and products, including the relative Unity Application.	VDS-122
VDS-169	The Unity application shall have an interactable UI	

Technology & Terminology

Blender – Computer graphics software used for 3D modeling

DigitalOcean – Web hosting company

DrawSQL – Platform used to create database entity relationship diagrams

ECMAScript – JavaScript standard developed by Ecma international meant to ensure the interoperability of web pages across web browsers

Express.js – Back-end web application and server framework for Node.js

Figma – User Experience Design Tool

GIMP – Open-source raster graphics editor for image manipulation and editing

HTTP – Application layer protocol used to manage requests and responses between computer systems via the World Wide Web

JavaScript – Lightweight, interpreted, or just-in-time compiled programming language which conforms to ECMAScript specification

Jira – Proprietary issue tracking product for bug tracking and agile project management

Mailgun – SMTP Service used to send emails to users

MongoDB – Document database system

Mongoose – NPM object-relational mapper used to work with MongoDB database

Node.js – Open-source back-end JavaScript runtime environment which executes JavaScript code outside of a web browser

NPM – Package manager for the JavaScript programming language

Oculus Quest 2 – Virtual reality headset developed by Oculus

StarUML 2 – Open-source software used to support modeling

UML – Modeling language used in the software engineering field

Unity – Game engine used for 2D and 3D application development

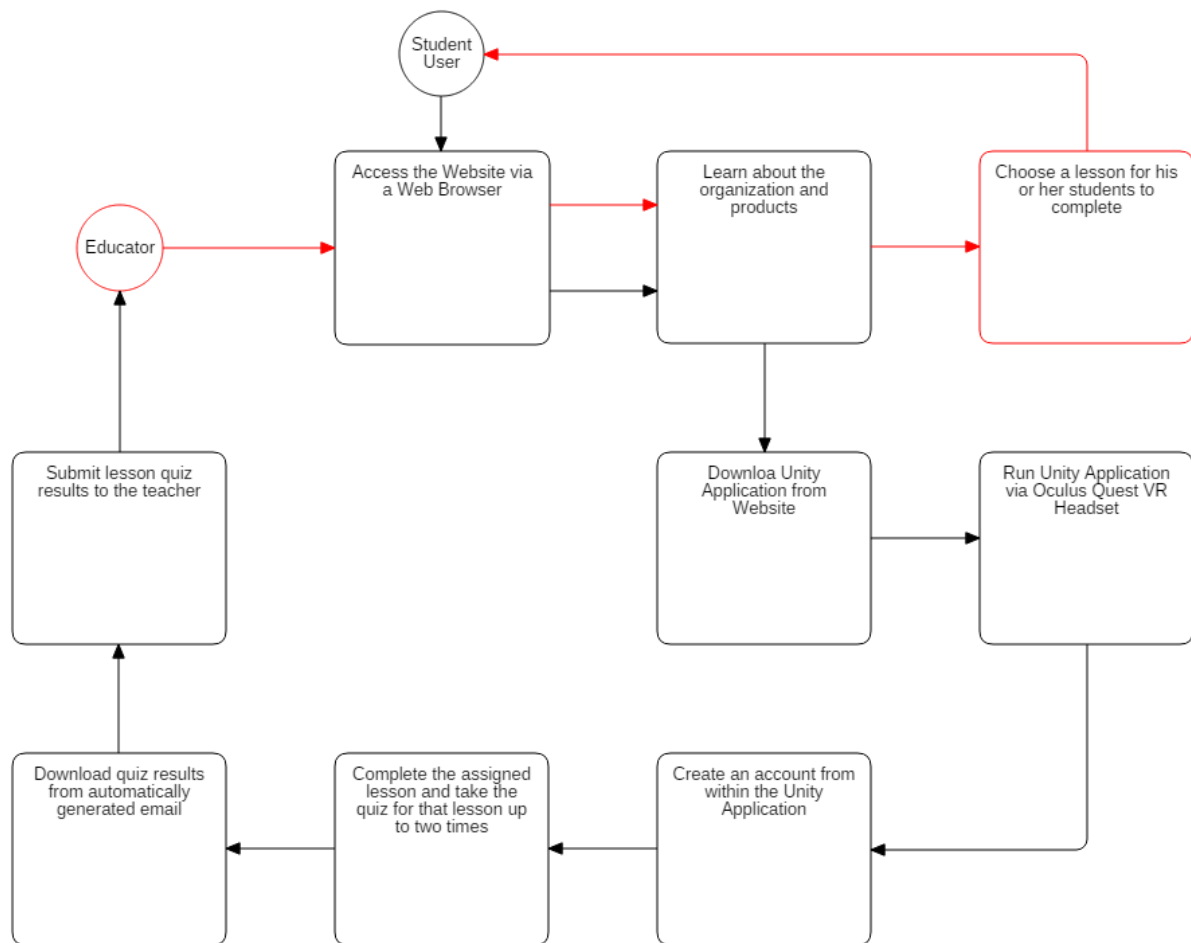
Visual Studio Code – Integrated development environment

Business Processes

Academic Use Case

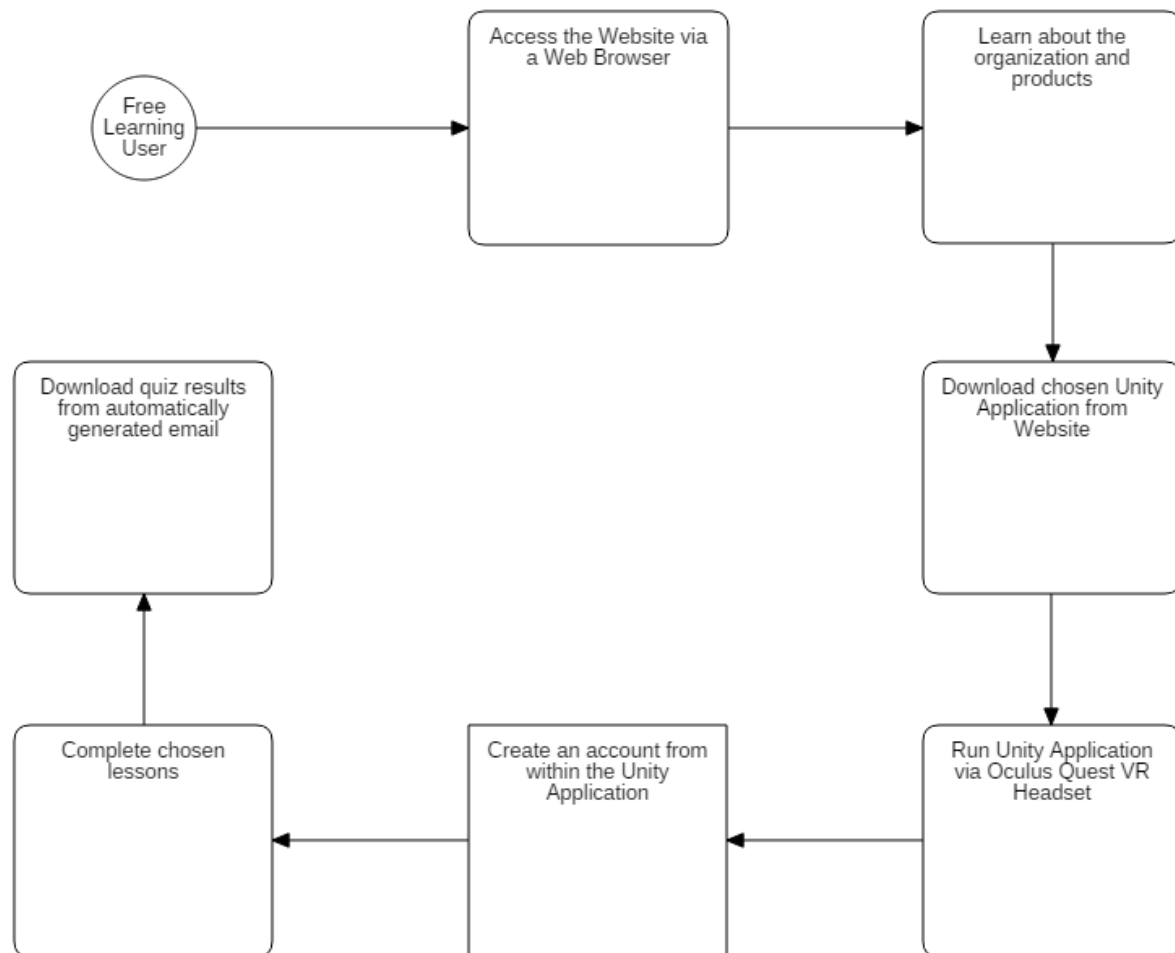
Current Business Process

The current business process for our new product would be focused on educators using the software to distribute lessons to their students; lesson material and assignments would all be housed within the software. Currently, in many cases, the educator has lessons and assignments in an archive that they've built on during their careers. Because STEM fields, specifically those related to computer science, evolve so rapidly, educators usually have stockpiles of lessons that are out of date and no longer relative. Our product would give supplementary resources to educators to teach new lessons to students and grade them on those lessons in a simple way. The product would also provide students with a unique way to experience their new lesson that their teacher would not be able to provide on their own.

New Business Process

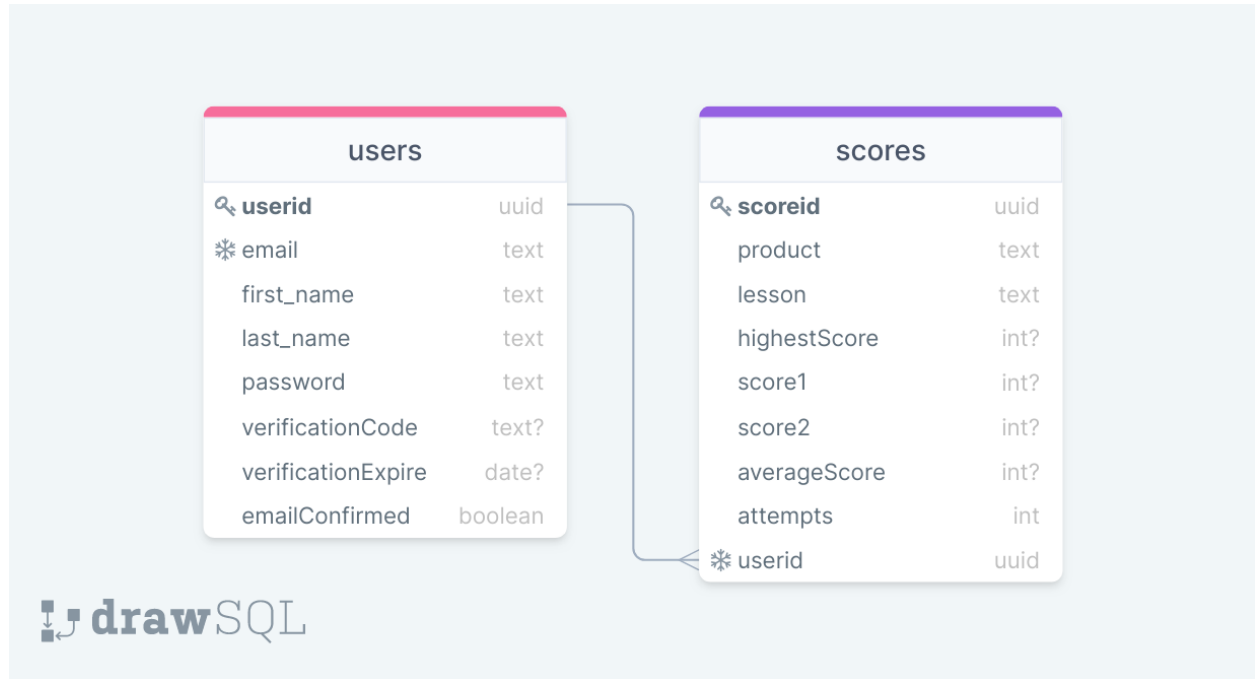
Free Learning Use Case

Proposed Business Process

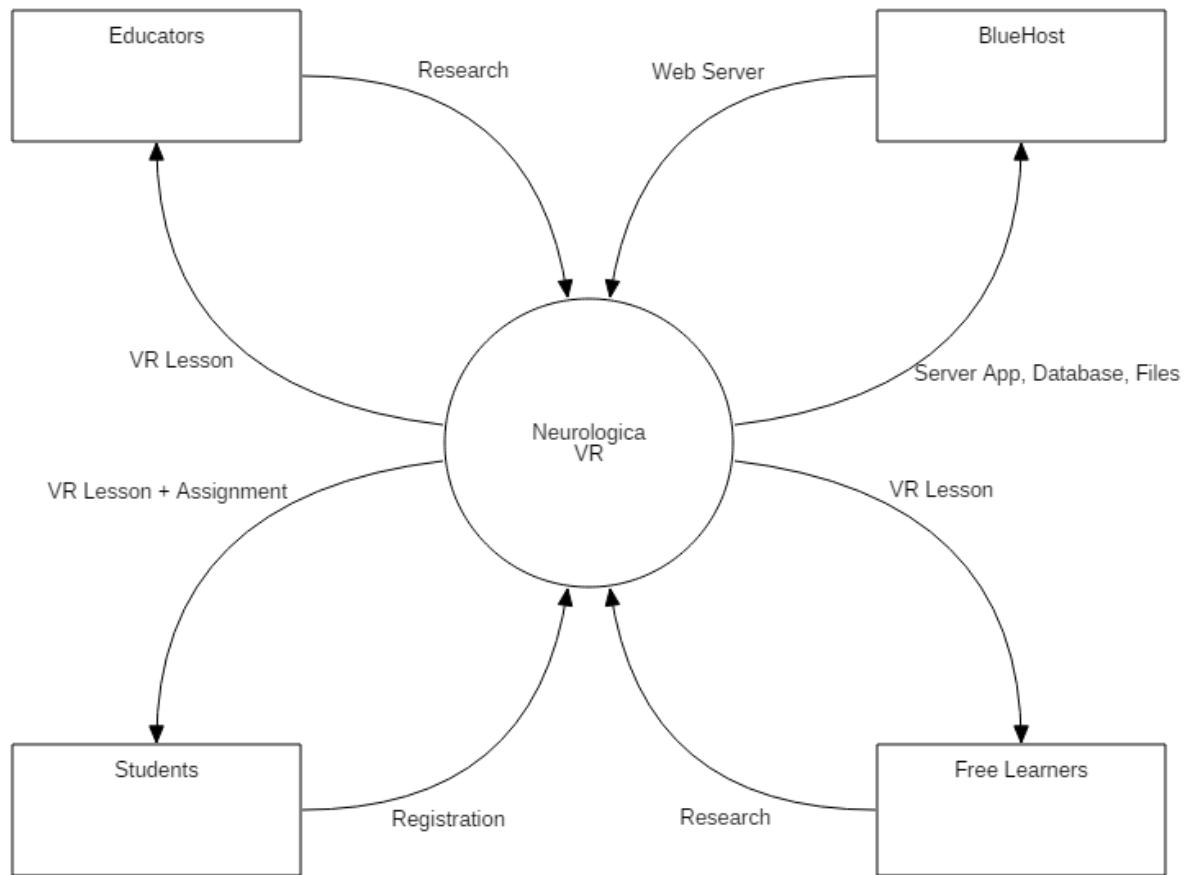


Models

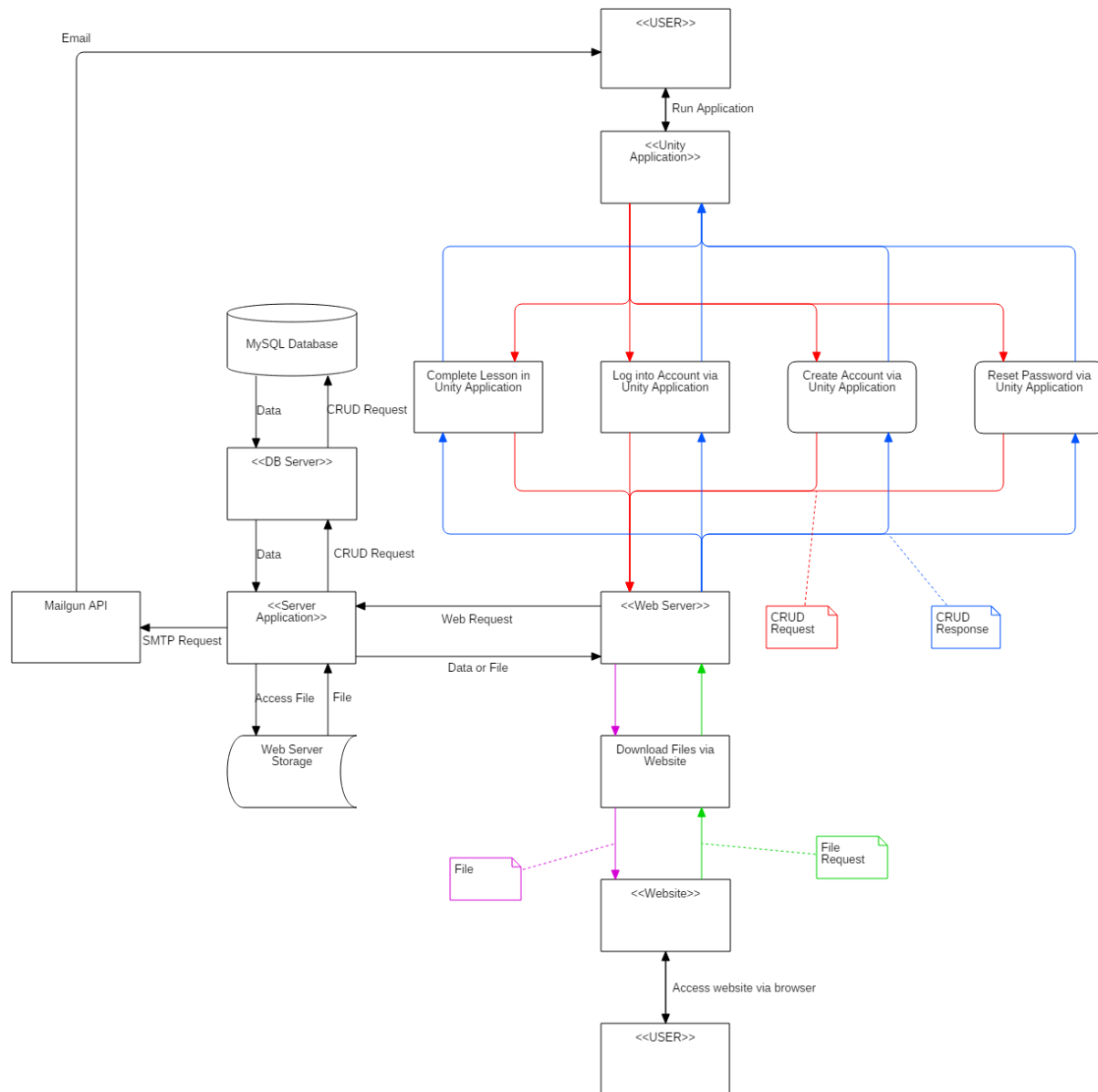
Database Diagram



Context Diagram



Level 1 Data Flow Diagram



Mockups

Unity UI Mockup

The white space around panels represents the scaled canvas size within the scene of Unity Engine and will be transparent in the application at run time.

Login Menu



The login menu for Neurologica VR features a pink logo at the top, consisting of three dots connected by lines. Below the logo is the text "Neurologica VR". The form includes input fields for "email" and "password", a link for "Forgot your password?", and buttons for "Create Account" and "Log In". At the bottom, there is a link to "Continue without an account" and a small text line stating "By continuing you agree to our terms".

Neurologica VR

email

password

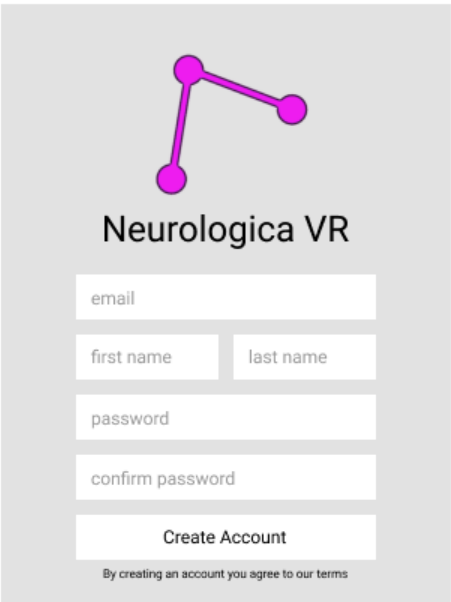
[Forgot your password?](#)

[Create Account](#) [Log In](#)

[Continue without an account](#)

By continuing you agree to our terms

Create Account



The create account form for Neurologica VR features the same pink logo at the top. Below the logo is the text "Neurologica VR". The form includes input fields for "email", "first name", "last name", "password", and "confirm password". A "Create Account" button is located at the bottom, followed by a small text line stating "By creating an account you agree to our terms".

Neurologica VR

email

first name last name


password

confirm password

[Create Account](#)

By creating an account you agree to our terms

Forgot Password




Neurologica VR

Please enter a registered email address

Reset Password


Reset Password



Neurologica VR

Set Password

Verify Email



Neurologica VR

Please enter the verification code sent to **\$(email)** to verify your account

Enter Verification Code

Resend Code

By creating an account you agree to our terms

Main Menu

Welcome Back **\$(firstname)**!

Progress


- > Lesson 1 | **Arrays** (You *completed* this lesson!)
- Lesson 2 | **Lists** (You *completed* this lesson!)

Quiz Score: 93%

Quiz Attempts: 1 of 2

Quiz Resets: 1

- > Lesson 3 | **Trees** (You *viewed* this lesson!)
- > Lesson 4 | **Heaps**
- > Lesson 5 | **Hash**

**VISUALIZE**
Data Structures

Options

Log Out

Lesson Menu (Lesson 1)

Lesson 1: Arrays

X << || 57%

Guide


Part 1: One Dimensional Arrays >>

Part 2: Multi-Dimensional Arrays >>

Go to quiz

Think about it

- What can you store in an array?
- What are objects in an array called?
- How do you know where things are in an array?

 **VISUALIZE**
Data Structures

Options

Log Out

Quiz Menu

Quiz: Arrays

Question 1: You can store different types of items in the same array. False True


Question 2: Arrays can increase in size after being made. False True

Question 3: The name for an item being stored in an array is an *object*. False True

Question 4: Items in an array are given a number for their position called an *index*. False True

Question 5: An array that stores other arrays is called a big-storage array. False True

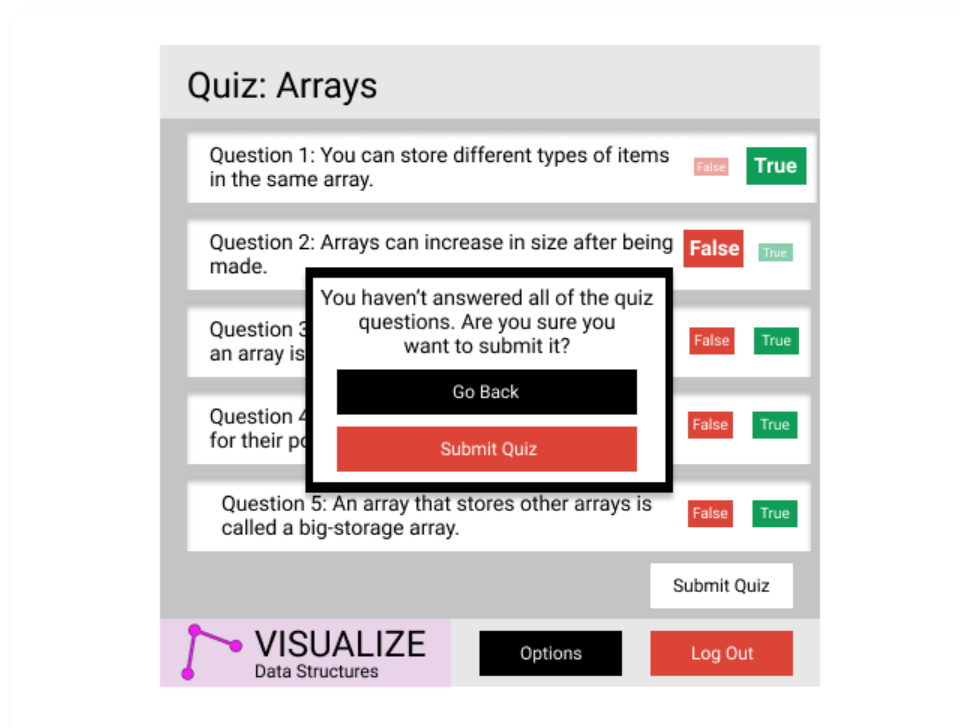
Submit Quiz

 **VISUALIZE**
Data Structures

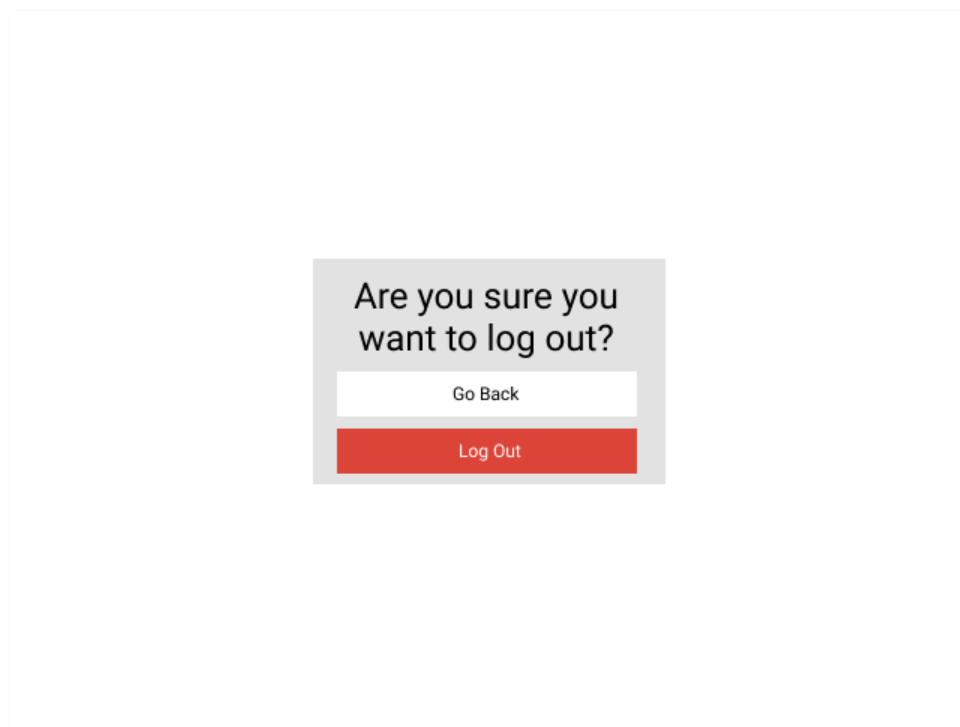
Options

Log Out

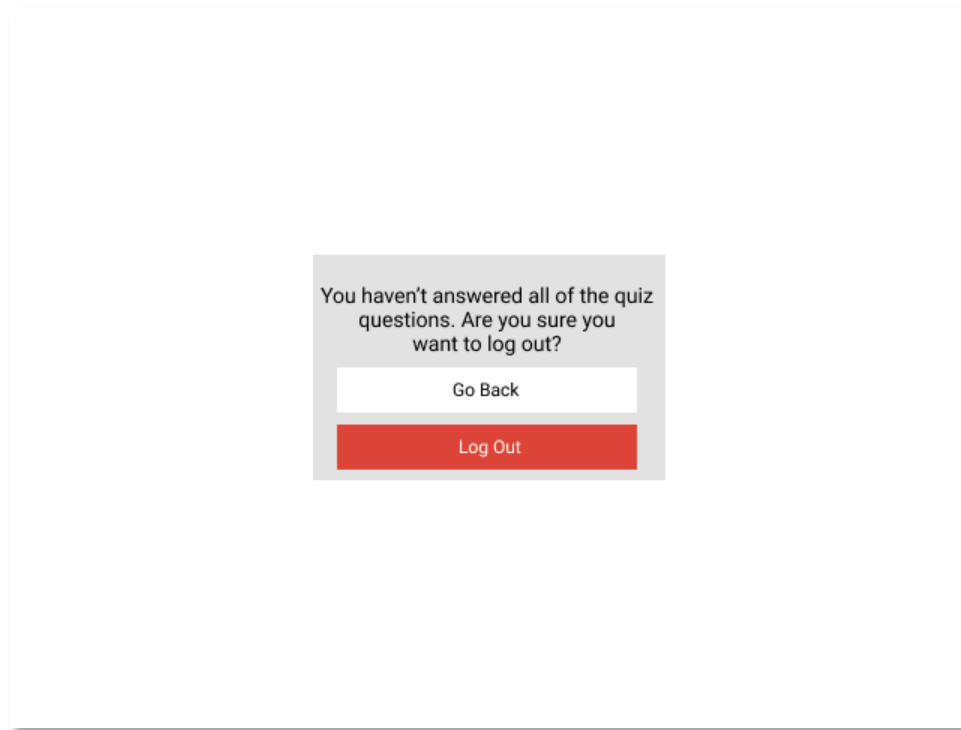
Quiz Menu (Early Submission)



Log Out Confirmation (Primary)



Log Out Confirmation (Quiz)



Unity Application Flow

Application Start

The application will be stored on the users PC as an executable file that will be opened within a connected Virtual Reality headset.

Alternatively, the application can be downloaded directly to standalone Virtual Reality headsets, like the Oculus Quest 2, in which it can be opened from the Oculus built-in application library.

User Starting Area

The user will start inside the "Start Area", a small area within the VR application that is isolated from the rest of the scene. The user will have access to a UI panel that will allow them to handle all account operations.

Initial Level Access

After successfully logging in to an account, the user will be given access to the main scene of the application. The main scene is made up of the Portal Room, Think Deck, and Central Hall. Each area is designed to provide the user with an immersive and interactive educational experience.

Lesson Access

Lessons can be accessed via the portals in the Portal Room. Each portal terminal has a short description of the lesson it is connected to allowing the user to easily access any desired lesson or sub-lesson.

Lessons

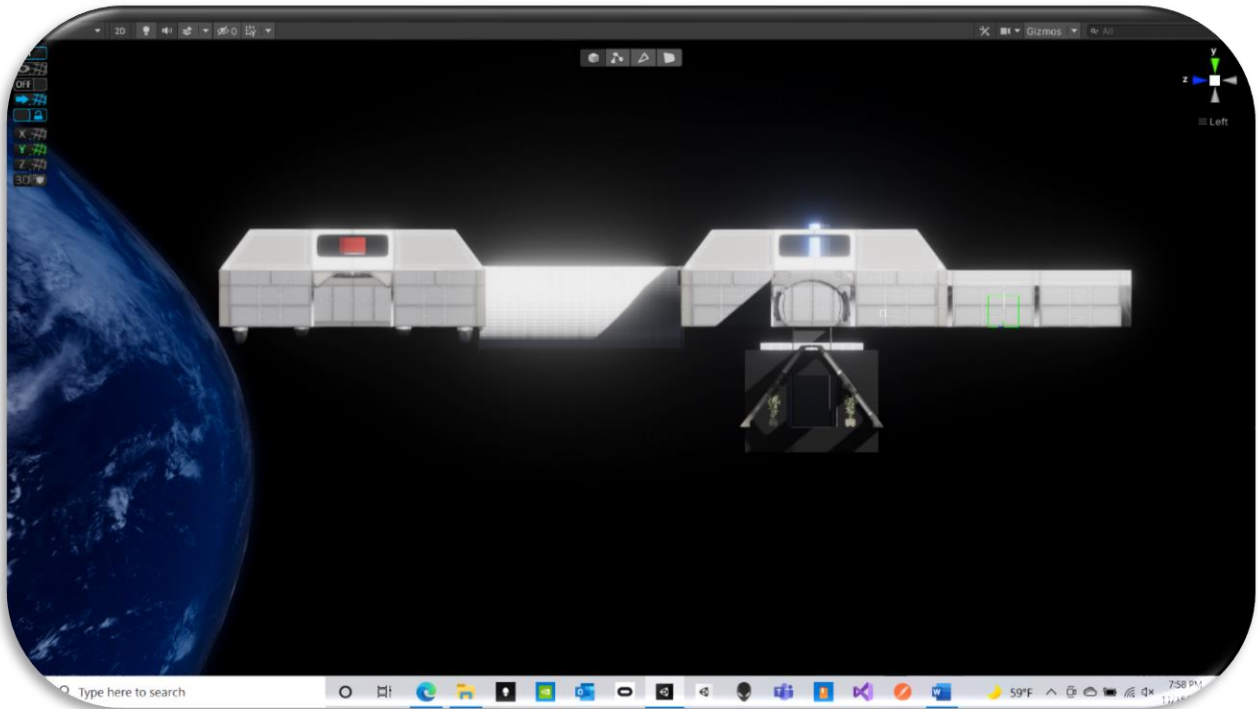
Once a user has accessed a lesson via a lesson portal, they will have access to multiple sub-lessons that will cover all the complex topics contained within that lesson. The information will be taught to the user through the implementation of games that utilize the interactivity to VR, drastically increasing retention rate.

Quiz Access & Submission

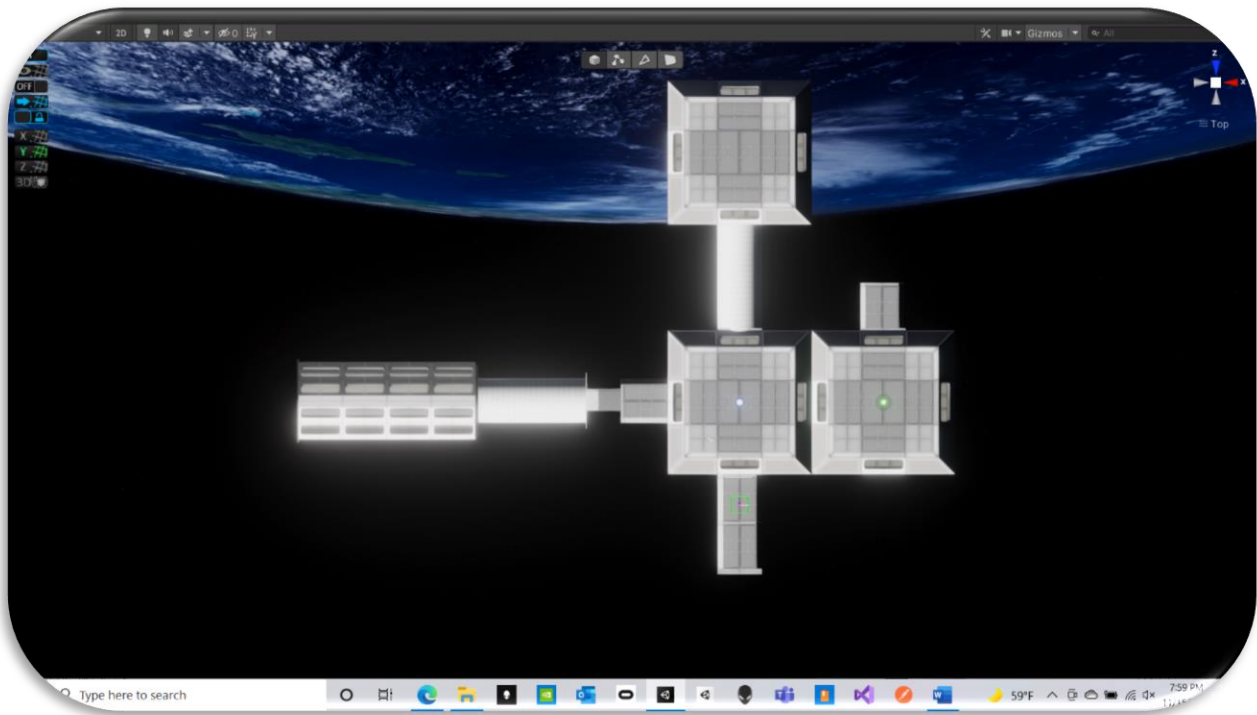
When a user is comfortable with specific lesson content, they will have the option to deploy the Quiz User interface system. This system can be deployed wherever the user is within a scene. The user will be able to select and attempt any available quiz; each quiz has set of true & false questions. As the user attempts a quiz, the users attempts, and scores are recorded and passed to the database for storage and future access.

Unity Application Gallery

2D Floor Plan (Side)



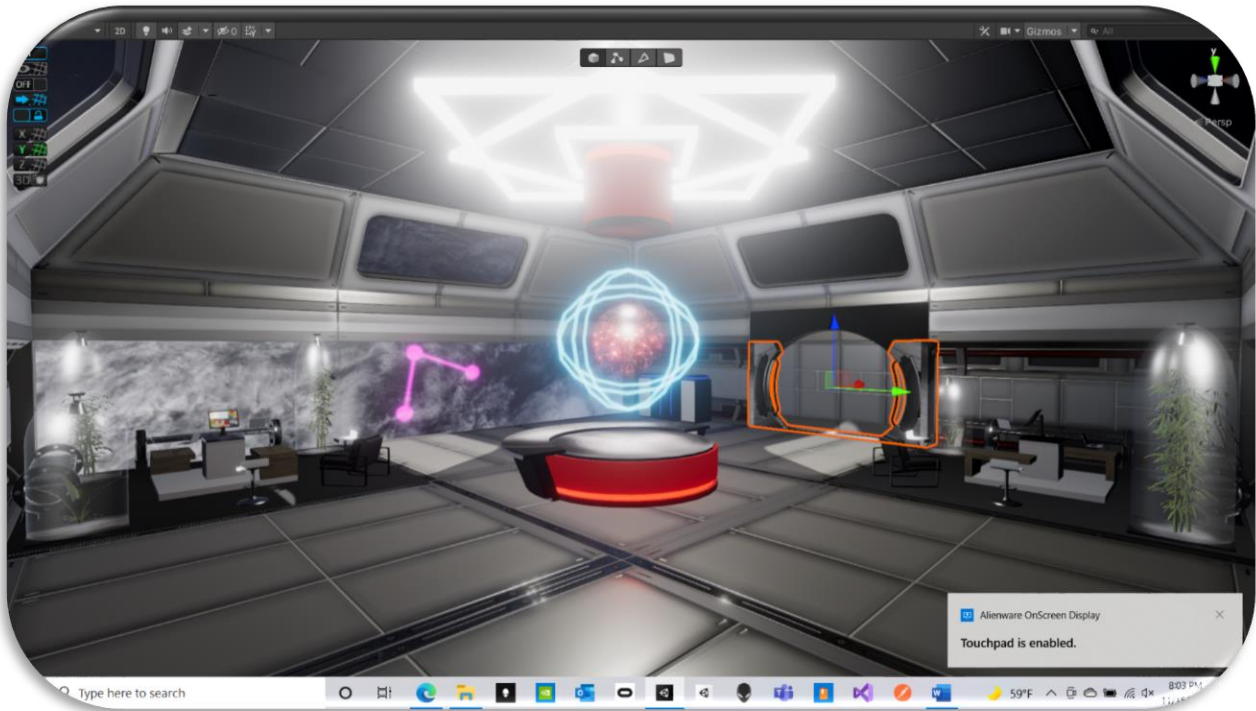
2D Floor Plan (Top)



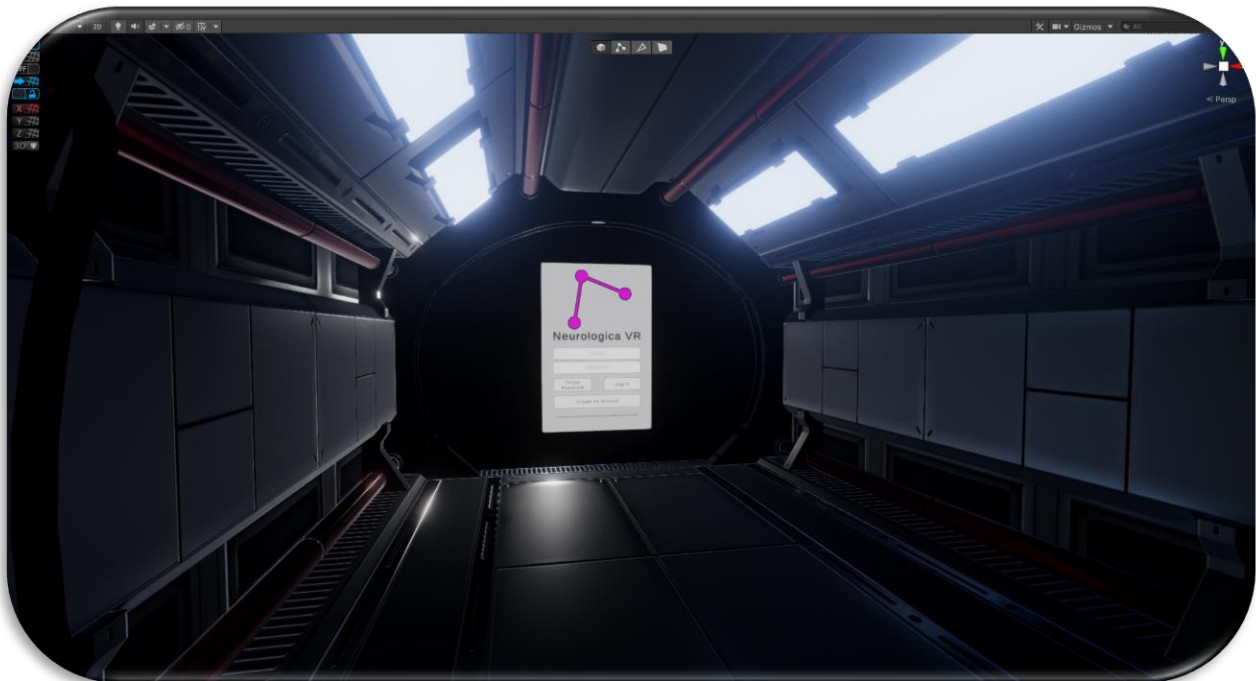
Portal Room



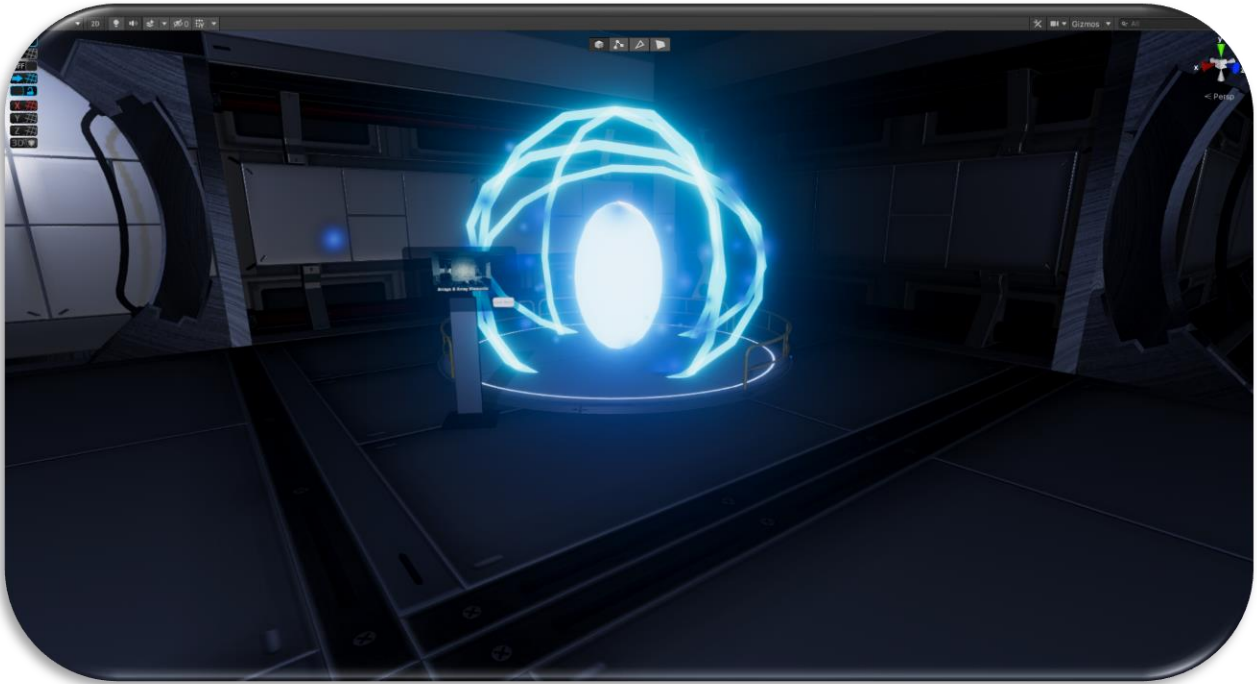
Central Hall



Starting Account Sign-In / Creation



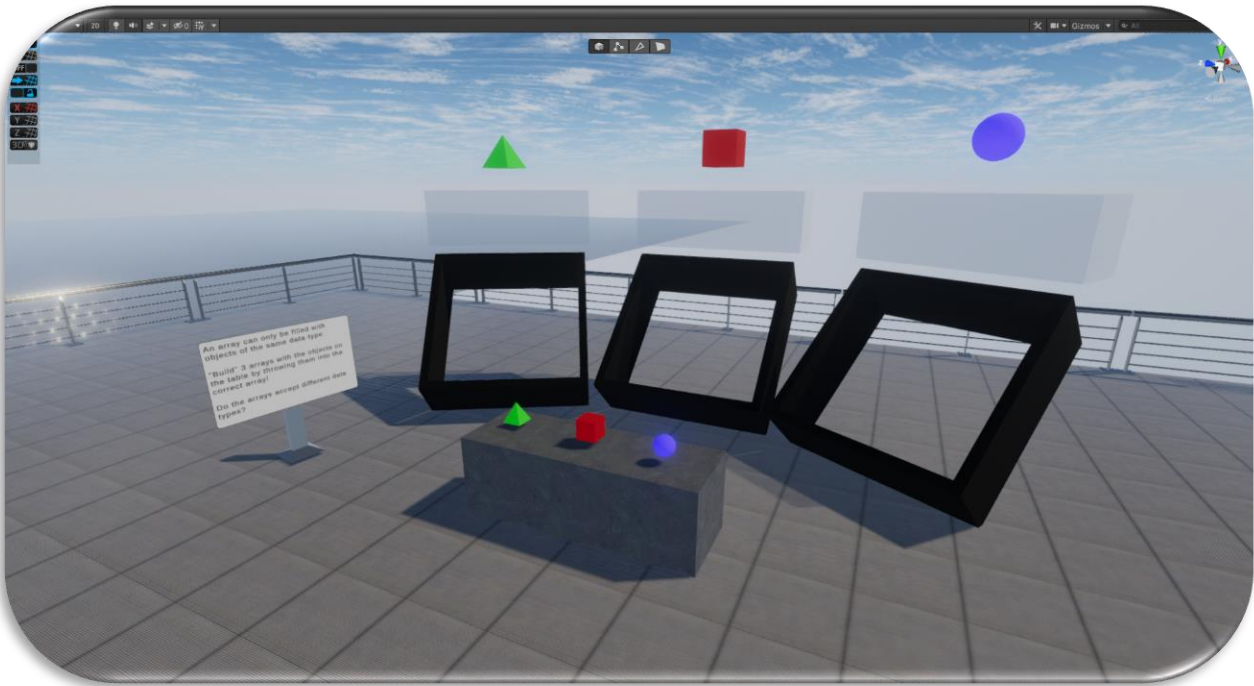
Portals to Lessons



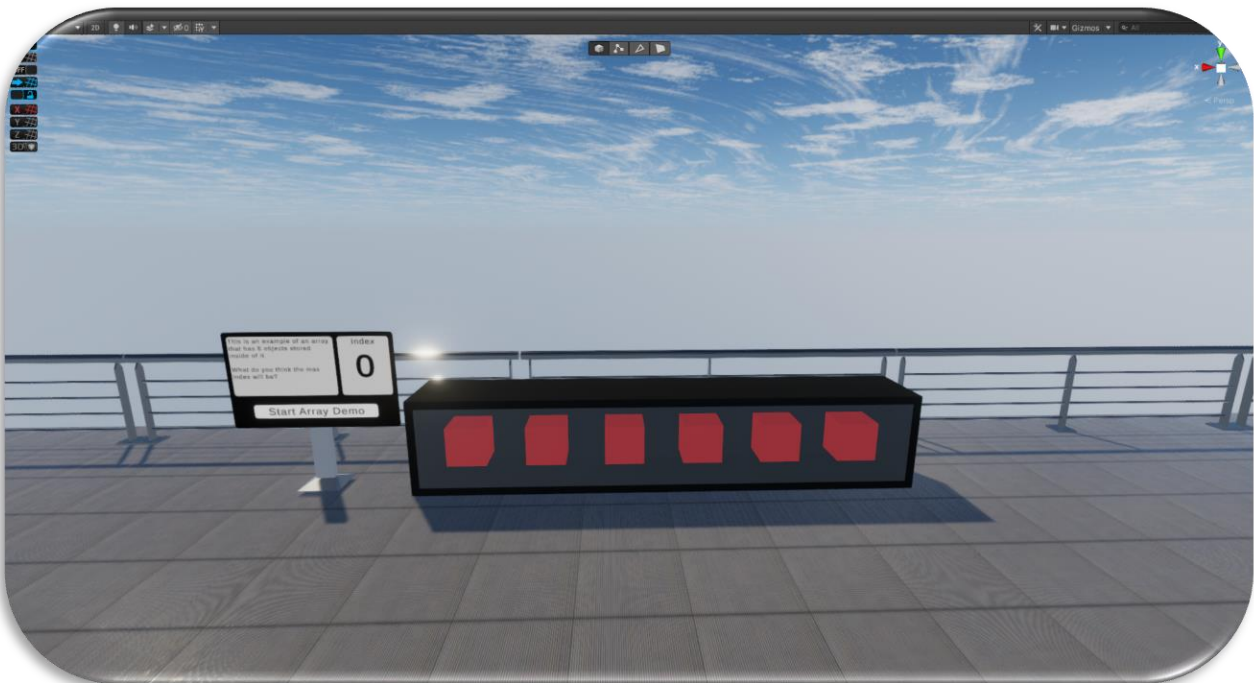
Think Deck



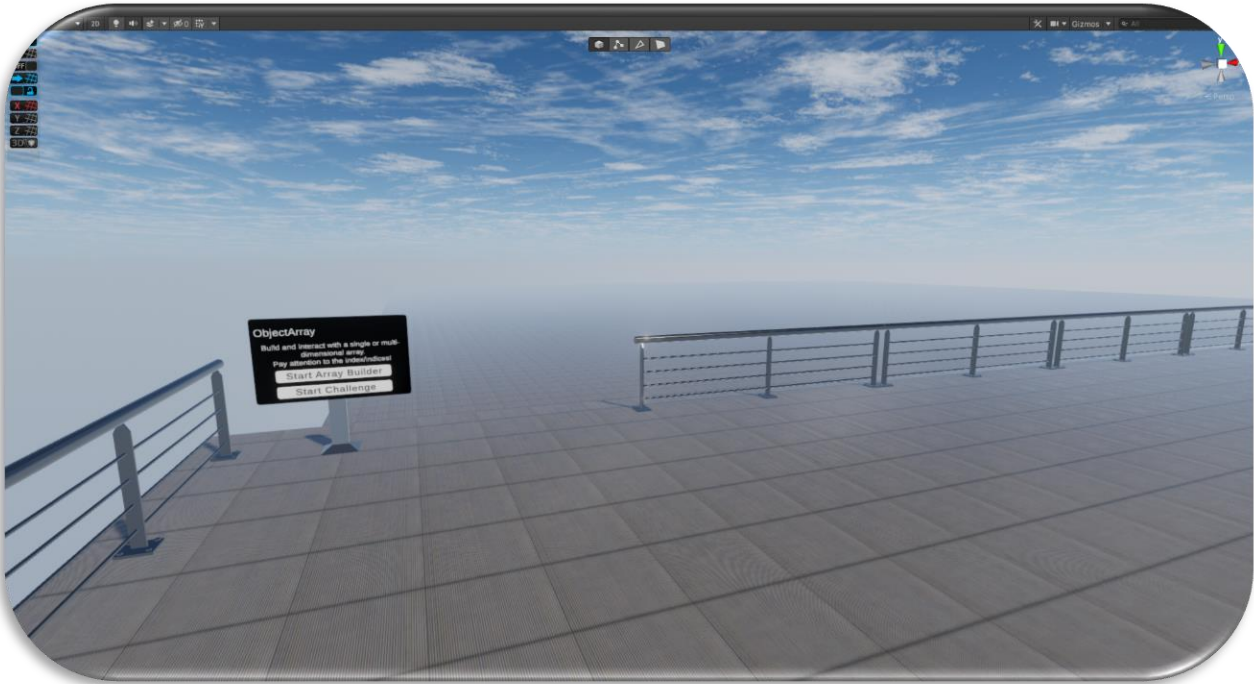
Arrays Lesson: Sub-Lesson 1



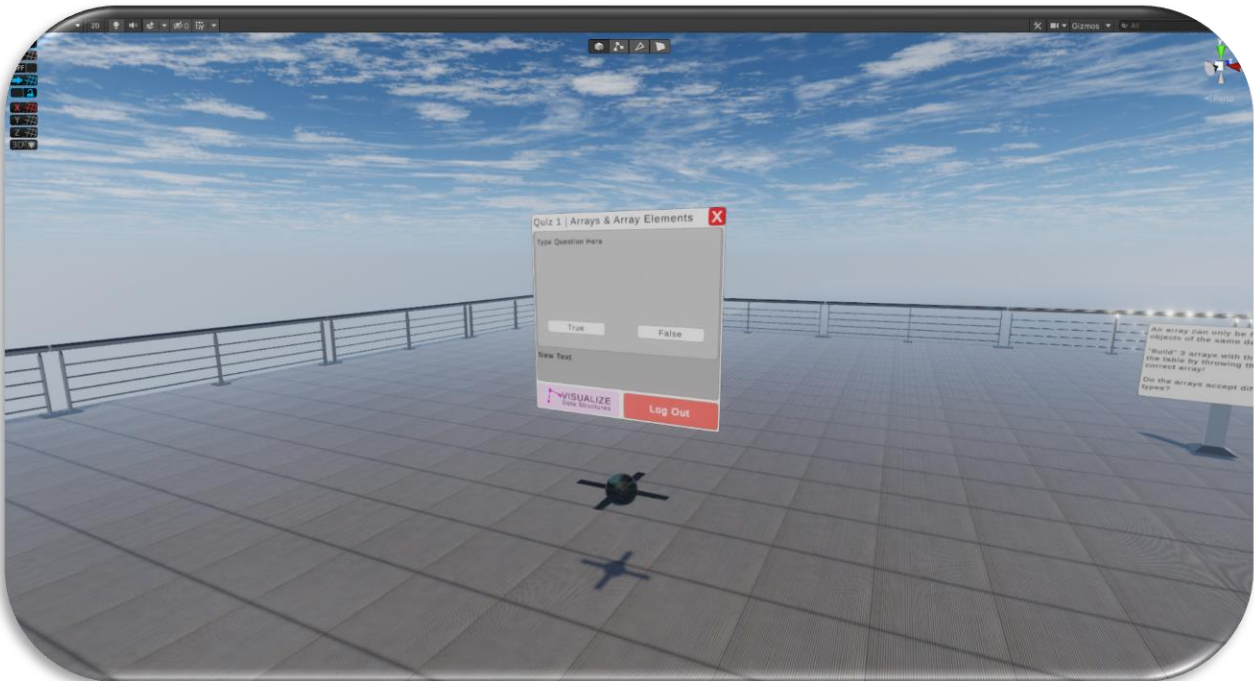
Arrays Lesson: Sub-Lesson 2



Arrays Lesson: Sub-Lesson 3



Runtime Spawned Quiz User Interface



Conclusion

Issues

We ran into various issues throughout the development of the project in both the server application and unity application. Some issues were solved while others were pushed until later iterations.

Issues Solved:

- Successful deployment of the server application to our web server
- Communication between the unity application and our server application
- Server responses digestible and useable from within the unity application
- Rate limiting and content security policies successfully implemented in the server application

Issues Left Unresolved:

- No server application security to check that requests are coming from the correct platform
- Unnecessary user password selected in response from server
- Bugged portable menu spawning from within the unity application
- Small and various collision bugs within the unity application

Results

Overall, the team is very happy with the results of the project. The technical goals of the project were met which including creating a unity VR application, server application, and organization website with various features as stated in our requirements. The business objectives we laid out were also reached which included providing a unique environment for users to learn in and give educators a new tool to teach and grade their students with. In the future, we'd like to add more content to this initial product, and we have created the infrastructure to add content through new products about other topics. Future goals also include allowing student users to view their grades via our organization's website.

References

Bcryptjs Documentation: <https://openbase.com/js/bcrypt/documentation>

Express.js Documentation: <https://expressjs.com/>

Helmet.js Documentation: <https://helmetjs.github.io/>

Mailgun Documentation: <https://documentation.mailgun.com/en/latest/>

Mailgun.js Documentation: <https://github.com/mailgun/mailgun.js>

Mongoose.js Documentation: <https://mongoosejs.com/docs/api.html>

Node.js Documentation: <https://nodejs.org/en/docs/>

Oculus Terms of Service: <https://www.oculus.com/legal/terms-of-service/>

Unity Documentation: <https://docs.unity3d.com/Manual/index.html>