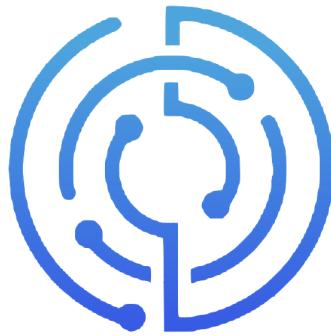


AutoPath: System Design Document



Bikram Chatterjee (b_chatterjee@u.pacific.edu)

Hung Le (h_le21@u.pacific.edu)

Lalit Vaidya (l_vaidya@u.pacific.edu)

<https://github.com/comp195/senior-project-spring-2022-autopath>

Last Date of Revision: 26/02/2022

Table of Contents

About AutoPath	3
System Architecture	4
Software Modules	4
External Software Systems	6
Hardware	7
User Interface	7
Hardware, Software, and System Requirements	10
External Interfaces	11
Software Design	13
Design Considerations	17
User Interface Design	18
Glossary of terms	23
References	25

About AutoPath

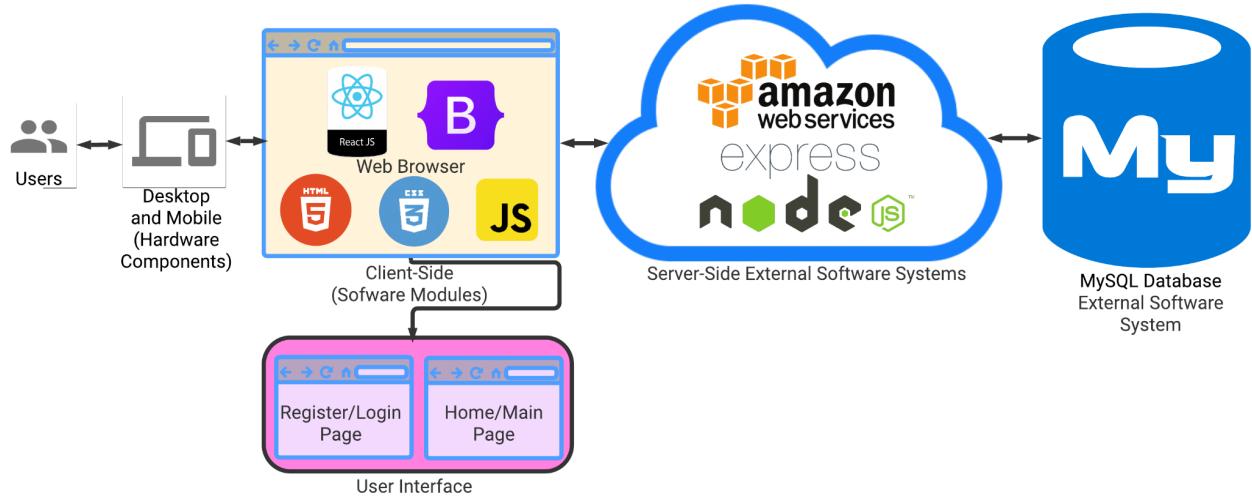
AutoPath is a web and VR application that visualizes pathfinding algorithms in action. At its core, a pathfinding algorithm seeks to find the shortest path between two points. It gives users the ability to choose from a randomly generated maze, maze presets, or simply build what they want. AutoPath offers algorithms such as Greedy Best-First Search, Dijkstra's Algorithm, A* Search, Breadth-First Search, and Depth-First Search for users to choose from. The application also allows users to create an account to save their custom mazes for access later or sign in as a guest for quicker access.

Features

- Algorithms
- Mazes & Patterns
- Add Weight
- Visualize
- Clear Board
- Clear Walls & Weights
- Clear Path
- Speed
- Nodes

System Architecture

Web Application



Block Diagram of all High-level System Components for the Web App

Software Modules

- **HTML** - HyperText Markup Language is the most basic building block of the Web. It defines the meaning and structure of web content.
- **CSS** - Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web.

- **JavaScript** - JavaScript is a scripting or programming language that allows devs to implement complex features on web pages. JavaScript is probably involved every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc.
- **React.js** - React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.
- **Bootstrap** is an HTML, CSS & JS Library that focuses on simplifying the development of informative web pages. The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project.
- **MySQL** is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder

Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

External Software Systems

- **Node.js** is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command-line tools and for server-side scripting running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web application development around a single programming language, rather than different languages for server-side and client-side scripts.
- **Express** is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

- Amazon Elastic Compute Cloud is a part of Amazon.com's cloud-computing platform, Amazon Web Services, that allows users to rent virtual computers on which to run their own computer applications or in this case servers. This project is going to use AWS to host the website and its server.

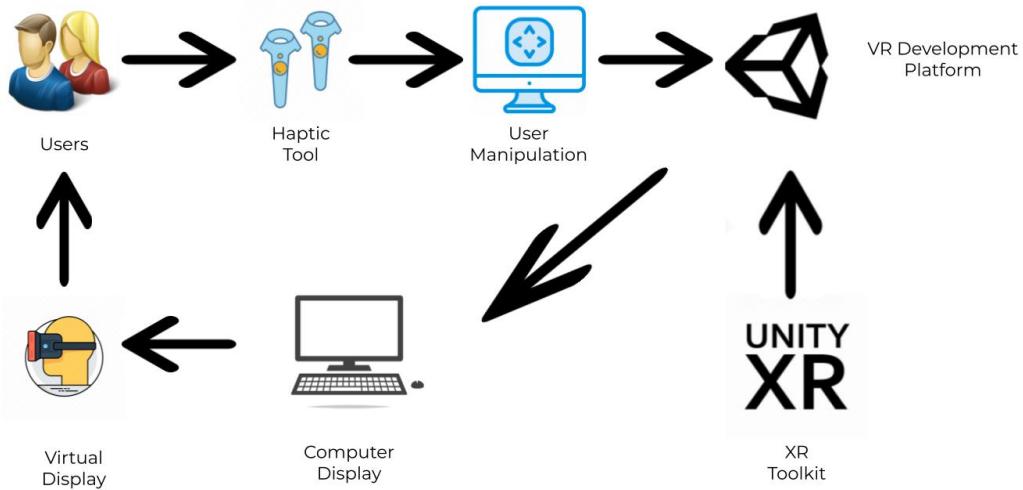
Hardware

The web implementation of AutoPath requires access to any device such as a computer or a phone that has a web browser installed on it and can connect to the internet.

User Interface

- **Login/Registration:** This page allows users to register as a new user, sign in to an existing account, or continue as a guest
- **Home/Main Page:** This page is where the users can set start and end points in a 2D grid, select from a list of various pathfinding algorithms, add walls and bombs, generate random mazes, visualize the selected algorithm, save their custom maze, and load their save mazes.

VR Application



Block Diagram of System Components for the VR App

Software Modules

- **XR ToolKit:** The XR Interaction Toolkit package is a high-level, component-based, interaction system. It provides a framework that makes 3D and UI interactions available from Unity input events. The core of this system is a set of base Interactor and Interactable components, and an Interaction Manager that ties these two types of components together. It also contains helper components that you can use to extend functionality for drawing visuals and hooking in your own interaction events.
- **VR Development Platform:** We will be using Unity in order to develop the VR application. Unity is a real-time development platform that allows the user to develop applications in VR. Unity provides a base API

for VR development as well as direct connection without the need for external plugins.

- **Oculus App:** We will be using Oculus App to connect the VR Headset to a computer through Airlink or USB-C

External Software System

We don't need any external software system to deploy for the VR application, everything is included in Unity.

Hardware

The VR application is used for Oculus VR sets. It requires a VR headset and touch controllers to view and control the application. The VR also needs to be connected to a computer to view the application. The Internet is not required to access or use the VR application.

User Interface

The VR application will have a 360-degree view of the AutoPath maze in a 3D grid. The VR application uses the same concept of the web application but without the ability for users to save or load the maze maps. The application doesn't allow users to log in so they can go straight to the main screen and check out the pathfinding algorithms.

Hardware, Software, and System Requirements

Hardware Requirements: In terms of hardware requirements, the user needs to have access to a computer and a VR device.

Minimum Computer Specifications with VR:

Processor	Intel i3-6100 / AMD Ryzen 3 1200, FX4350 or greater
Graphics Card	NVIDIA GTX 1050 Ti / AMD Radeon RX 470 or greater
Memory	8GB+
Operating System	Windows 10
Disk Space	10GB
Wifi	Yes

Minimum Computer Specifications without VR:

Processor	1.9 gigahertz (GHz) x86- or x64-bit dual-core processor with SSE2 instruction set
Graphics Card	None
Memory	2GB+
Operating System	Windows 10
Disk Space	1GB
Wifi	Yes

Software Requirements: Unity, XR Toolkit

System Requirements: Any Operating System with internet browser access

External Interfaces

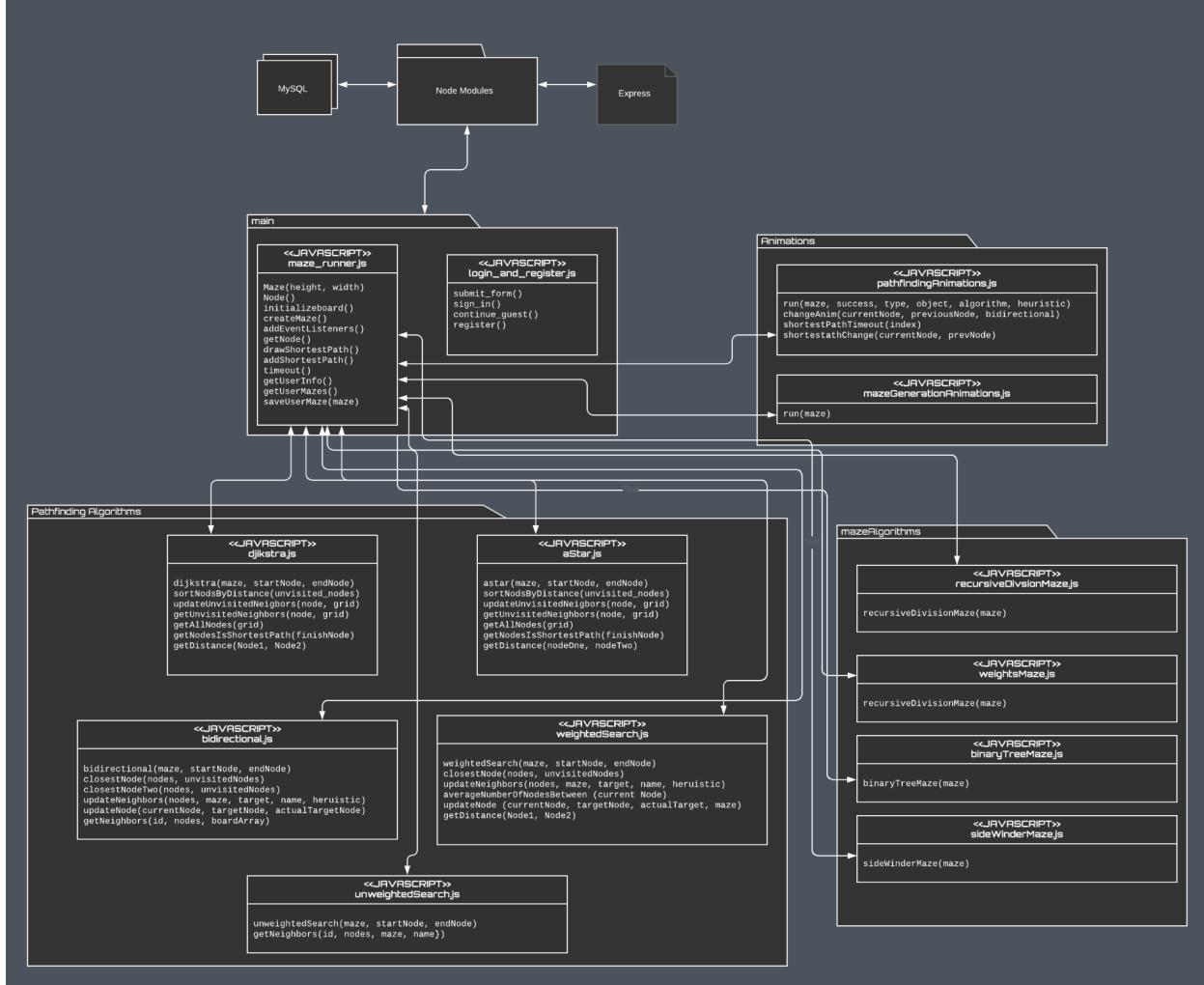
- XR Interaction Toolkit: The XR interaction toolkit is a library package that allows the user to create VR experiences. It provides the basic necessities for creating 3D and UI interactions through the use of a VR controller. It includes basic toolkits like controller inputs, basic object hover, select, and grab, haptic feedback, and visual feedback to indicate interactions. This package can be imported using Unity by going to the scene and clicking on the GameObject menu. Once the user clicks on the GameObject menu, there is an option called XR and within that menu, there is an option called the XR Rig.
- Another application needed to run the VR application is the oculus app. The Oculus app allows the user to sync their VR device to the computer via Airlink or a USB C cable. After connecting the device, the user should be able to access the Unity desktop on the VR device and can run the application from the device itself.
- **Node.js** is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a

"JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

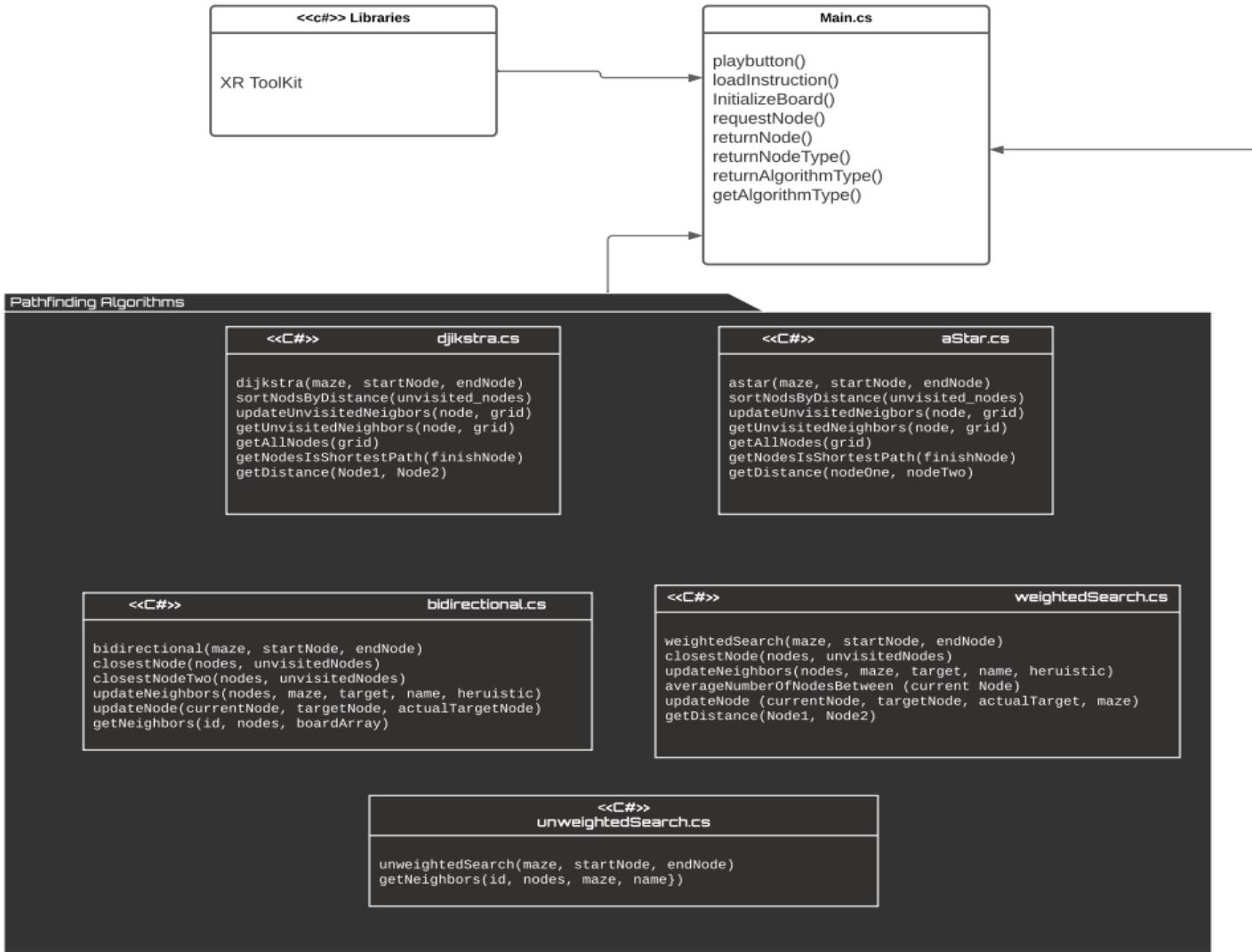
- **Express** is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications
- **Amazon Elastic Compute Cloud** is a part of Amazon.com's cloud-computing platform, Amazon Web Services, that allows users to rent virtual computers on which to run their own computer applications or in this case servers. This project is going to use AWS to host the website and its server.

Software Design

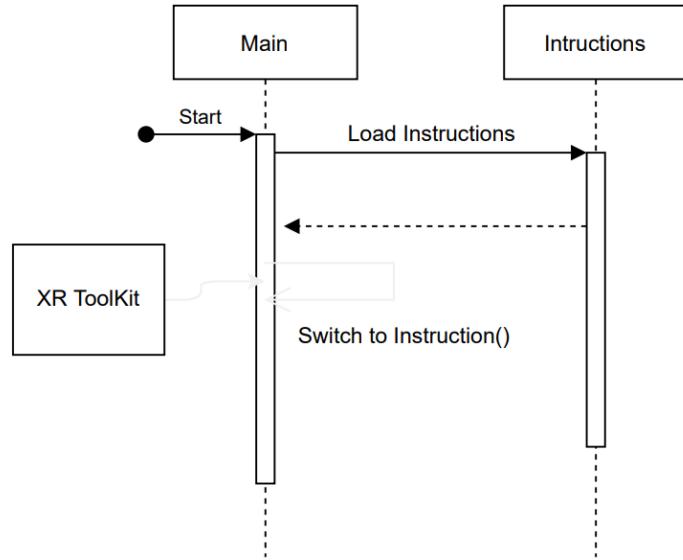
Web UML Diagram



VR UML Diagram

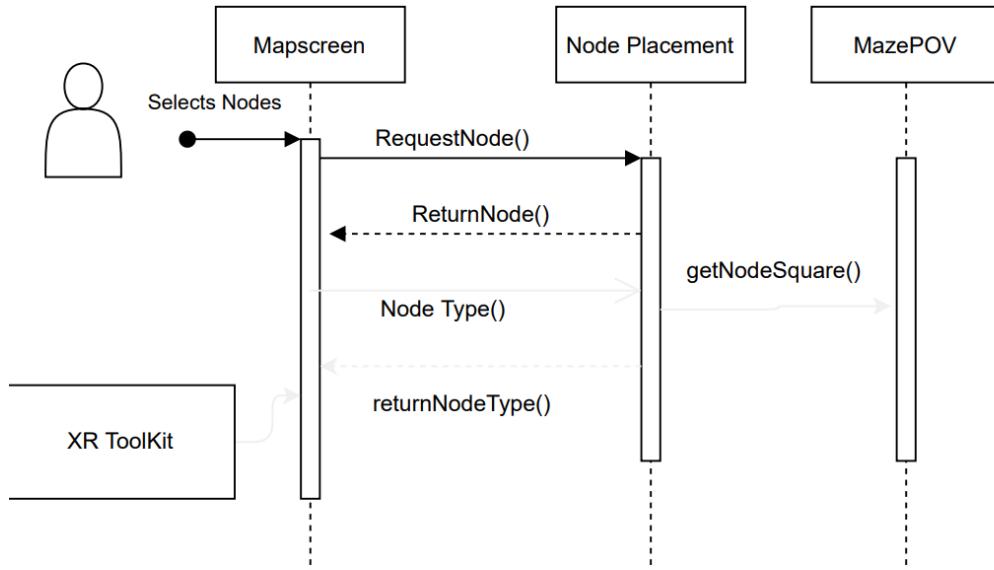


Interaction Diagrams



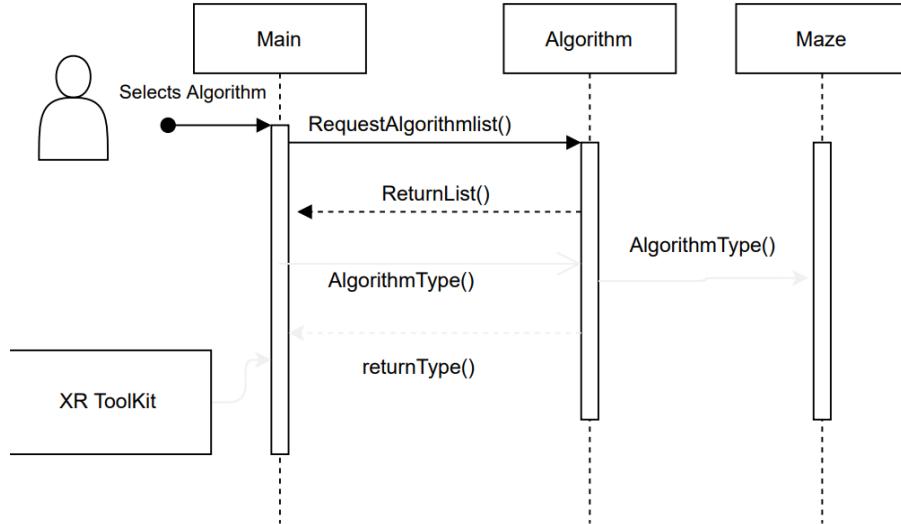
User Interaction Diagram for Starting Application

Selecting Nodes



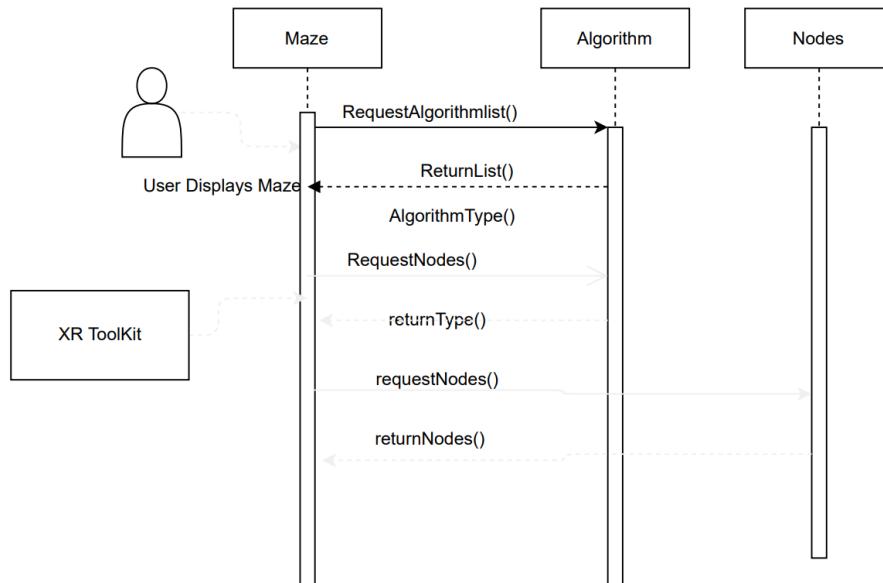
VR User Interaction Design for Selecting Nodes

Selecting Algorithm



User Interaction Design for selecting Algorithm

Display Maze



User Interaction Design for Displaying Maze

Design Considerations

The important design principle we are going to use is how to implement the algorithms into the maze. It is the biggest challenge we are going to face as we are using 8 algorithms for the users to choose from. VR graphic design is the next important principle we need because it will make our application more user-friendly and easy to interact with.

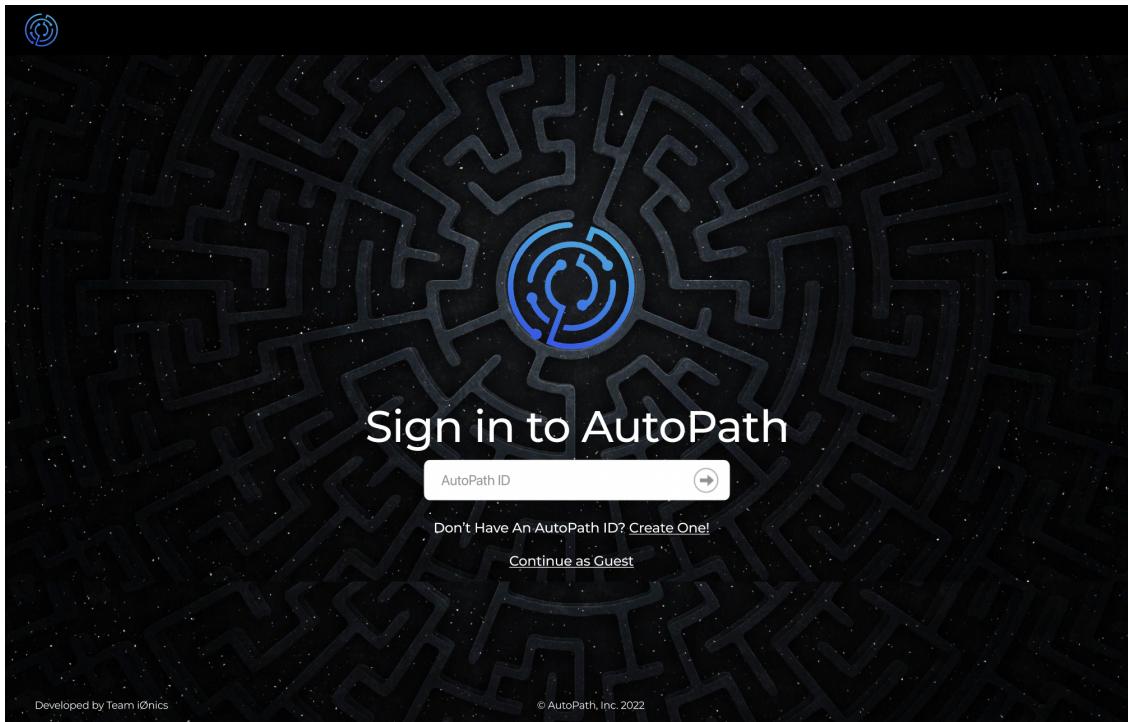
Another challenge is implementing the actual visualization animations in a way that users can learn how the various pathfinding algorithms work.

User Interface Design

Web Application

AutoPath offers users an interactive Pathfinding Visualizing. It allows the users to instantly see the shortest path from start to the target node and also recreate different paths in mazes by dragging nodes. The UI mockups are provided below:

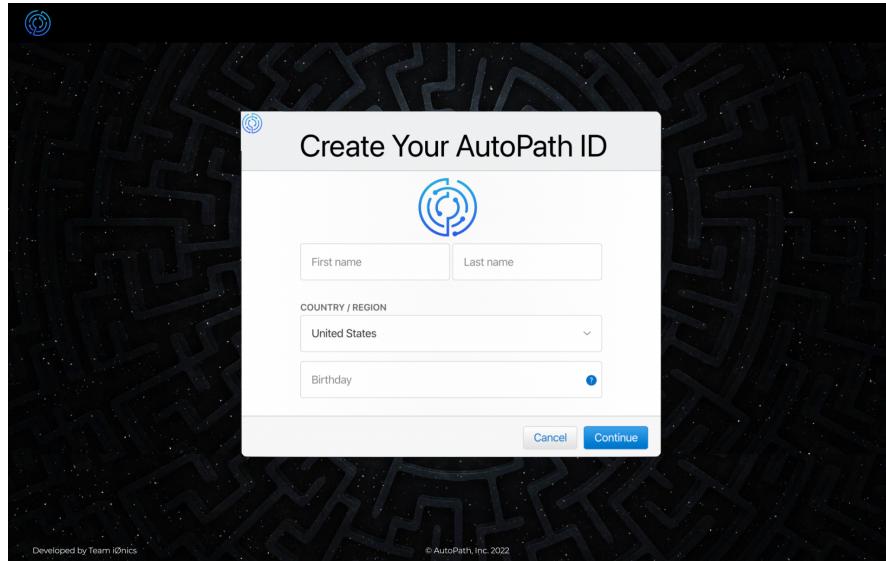
Login Screen



Login Screen

After Users enter the AutoPath Web Application through its link, they will be introduced to a welcome page, this is where they will get to choose if they want to sign-in or sign-in as a guest account for faster access.

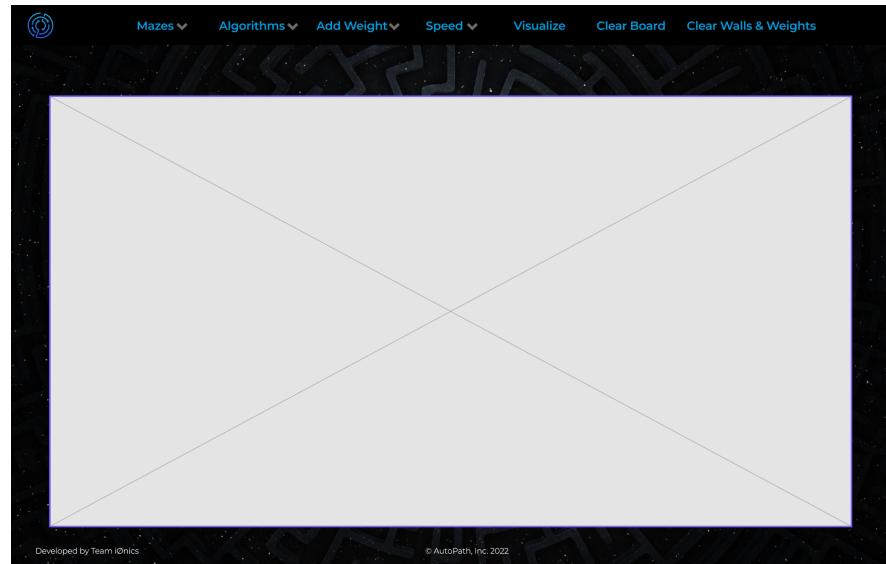
Register page



Register Page

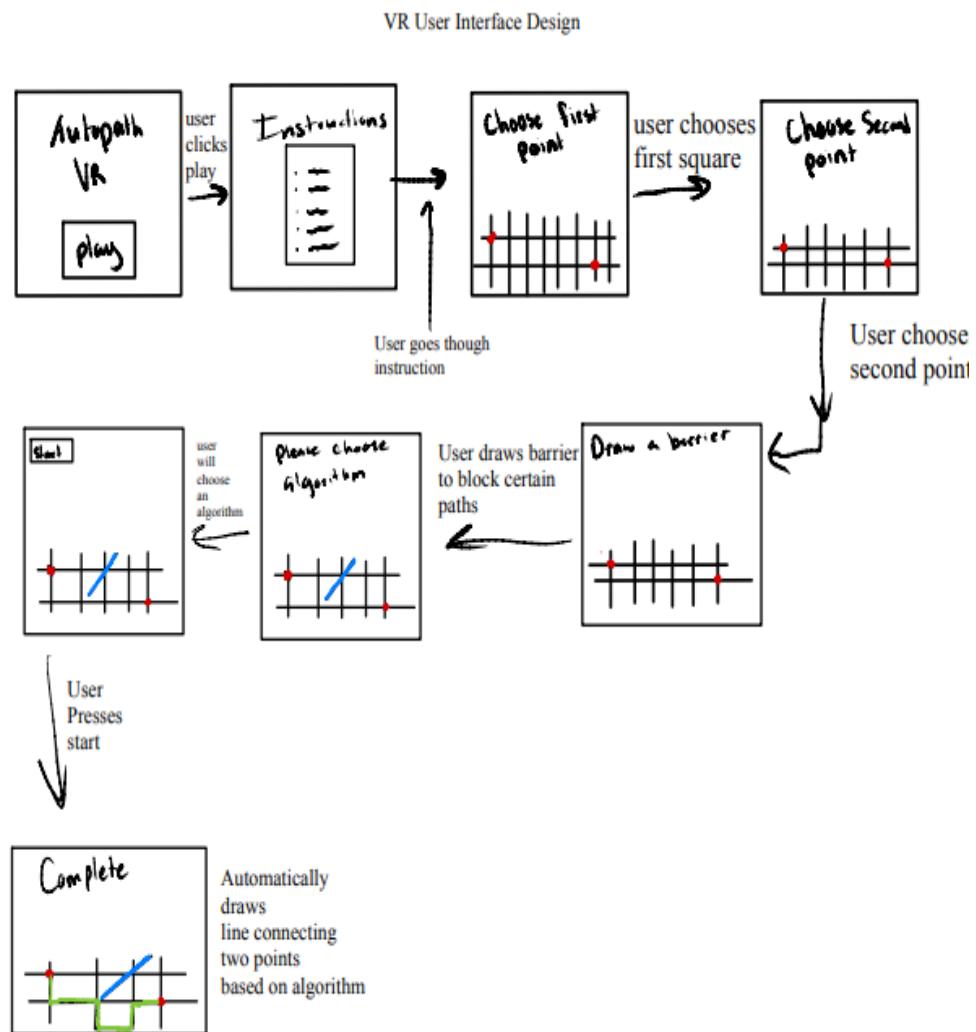
Register page allows users to insert their information to register for an account with AutoPath. Data includes name, email, and country.

Home Page



This is the page where the users can get to select the type of algorithm and maze they want or else recreate a maze of type. They can also add wall, bomb, and weighted nodes for the maze. After the maze has been created, the users can select the button to start visualization and see the magic unfold! Later on, they have the option to save their maze and access them again in the future if they have created an account.

VR Application



AutoPath will take place in a 360 view island where users have the freedom to walk around in the virtual space. They can manually place the nodes in the maze and recreate anything they want. Autopath provides them a camera option where they can change the view of the camera and see the maze at any position they want.

- Home Screen

This screen will have the same contents as the homepage for the Autopath web application. Except the users don't have the option to create an account, they can directly go to the application after getting through the welcome screen.

- Introduction

All the users will get the same tutorial information as the web application to show the features and it will be in the pop up screen. They will have the option to skip the tutorial or see through the features.

- Main Screen

For this application, since the users don't have the ability to create an account. The save maze option isn't available for the AutoPath VR Application. Other than that, the application is going to have the same features and ideology as the web application.

Glossary of terms

- Dijkstra's Algorithm: The father of pathfinding algorithms; guarantees the shortest path

- A* Search: Arguably the best pathfinding algorithm; uses heuristics to guarantee the shortest path much faster than Dijkstra's Algorithm
- Greedy Best-first Search: A faster, more heuristic-heavy version of A*; does not guarantee the shortest path
- Swarm Algorithm: A mixture of Dijkstra's Algorithm and A*; does not guarantee the shortest-path
- Bidirectional Swarm: Swarm from both sides; does not guarantee the shortest path
- Breadth-first Search: A great algorithm; guarantees the shortest path
- Depth-first Search

A very bad algorithm for pathfinding; does not guarantee the shortest path
- Start Node: The users can place the start node at any point in the maze and it will be the starting point of the path. They can also drag the node after the algorithm is finished running to see different paths possible.
- Target Node: The node can be placed anywhere in the maze and that will be the ending point of the path.
- Bomb Node: The node is placed anywhere in the maze and after running the algorithm, the path has to go through the bomb node and then later find the shortest path to the target node. The user has the freedom to add, remove, and change the position of the bomb node.
- Weighted Node: The node is placed anywhere in the maze that acts as a barrier but they are not impassable. It is more simply tough to move through it.

- Visited Node: When the visualization gets started, the visited nodes pop up to show that the algorithm is searching for the target node. So, using the visited nodes, the shortest path nodes can be created.
- Unvisited Node: The node is an empty node where the user can place wall, bomb, start and target. It is the base of the maze.
- Shortest-Path Node: The node is created when the algorithm is done searching the target node. It is used to show the shortest path in the maze and the user can drag the node to see if there is another possible path or not.
- Wall Node: The wall node is placed anywhere in the maze that acts as a barrier. The path that leads from the starting node to the end node must not intersect with the wall node. The path would have to go around the wall node.
- Airlink: Air Link allows the user to connect your quest device wirelessly to your PC using a secure Wifi network.

References

Express - Node.js web application framework. (n.d.). Express JS.

<https://expressjs.com/>

Mihaiescu, Clément. (2019, August 27). *The Projects That Got Me Into Google (tips for software engineering projects [Video])*. YouTube.

https://www.youtube.com/watch?v=n4t_-NjY_Sq&t=0s

Jakupi, R. (2022, January 6). *How to Deploy a Node.js application in AWS EC2 - AWS in Plain English*. Medium.

<https://aws.plainenglish.io/deploying-a-nodejs-application-in-aws-ec2-c1618b9b3874>

XR Interaction Toolkit | XR Interaction Toolkit | 0.9.4-preview. (2018). Unity.

<https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@0.9/manual/index.html>

Wikipedia contributors. (2022, February 6). *Node.js*. Wikipedia.

<https://en.wikipedia.org/wiki/Node.js>