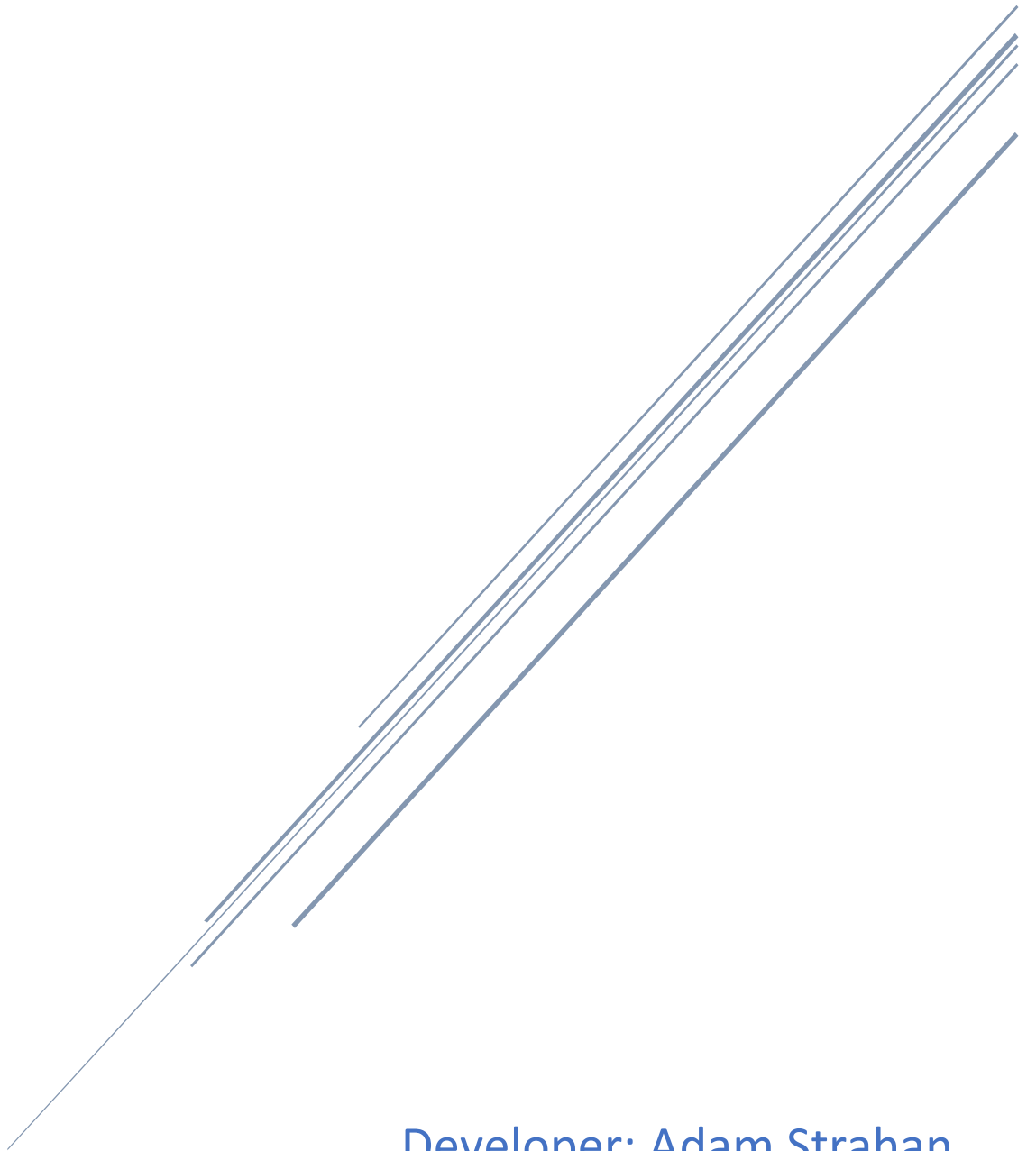


INTERACTIVE ITC MAP

RESEARCH DOCUMENT



Developer: Adam Strahan
Supervisor: Oisín Cawley

Table of Contents

Introduction	2
Coding Language	2
JavaScript	2
Python	3
Java	4
React Native	5
Flutter	5
Xamarin	5
Decision	5
Development Environment	6
Android Studio	6
Visual Studios	6
Expo	7
Decision	7
Hardware	8
Laptops/Computers	8
Smartphones	8
Algorithms	8
Dijkstra's Algorithm	8
Example	9
A* Algorithm	9
Example	10
Decision	10
Similar Products	11
Penn State Map	11
MazeMap	11
Conclusion	12
References	13
Table of Figures	13
Bibliography	13
Appendix	15
Plagiarism Form	15

Introduction

For my fourth-year final project I am developing an interactive map of the Institute of Technology Carlow campus. This map must have pathfinding to help students find their classrooms, ideally using the shortest routes. It would also be a good idea to provide information within the application about the various buildings on the campus. In order to develop this application, I will need to research a number of topics. These topics include:

- What coding language should I use?
- What development environment should I use?
- What hardware should I develop the application for?
- What algorithm should I use for the pathfinding?
- Are there any similar applications available?

I intend to document my research for the project in this document.

Coding Language

There are many languages that I can use to write this application. The most commonly used for this type of application are JavaScript, Python and Java.

JavaScript

JavaScript is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled and multi-paradigm. It has dynamic typing, prototype-based object-orientation and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. Over 97% of websites use it client-side for web page behaviour, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on the user's device.[8]

```
1 // Declare vars
2 const x = 11, y, z;
3
4 const greaterThan = () => console.log('greater than 10');
5 const lessThan = () => console.log('less than 10');
6 const doSomething = () => console.log('SOMETHING');
7 const checkScope = () => console.log(`The scope is: ${this}`);
8
9 if (x>10) greaterThan(), doSomething()
10 else lessThan();
11
12 const answer = x > 10 ? greaterThan()+doSomething() : lessThan();
13
14 const dbHost = process.env.DB_HOST || 'localhost';
15 console.log(`DBHOST ${dbHost}`)
16
17 // loops in plain english
18 const _ToDoList = ['get bread', 'get meat', 'get dressing'];
19 _ToDoList.forEach((item) => console.log(item) );
20
21 // Default Function Parameter Values
22 volume = (l, w = 3, h = 4) => (l * w * h);
23
```

Figure 1. JavaScript Code Example. Source: medium.com

There are a lot of third-party libraries that can be utilized with JavaScript to create a map application. One of these libraries is Leaflet.

Leaflet is the leading open-source JavaScript library for mobile-friendly interactive maps. Weighing just about 39 KB of JS, it has all the mapping features most developers ever need.

Leaflet is designed with simplicity, performance and usability in mind. It works efficiently across all major desktop and mobile platforms, can be extended with lots of plugins, has a beautiful, easy to use and well-documented API and a simple, readable source code that is a joy to contribute to.[9]

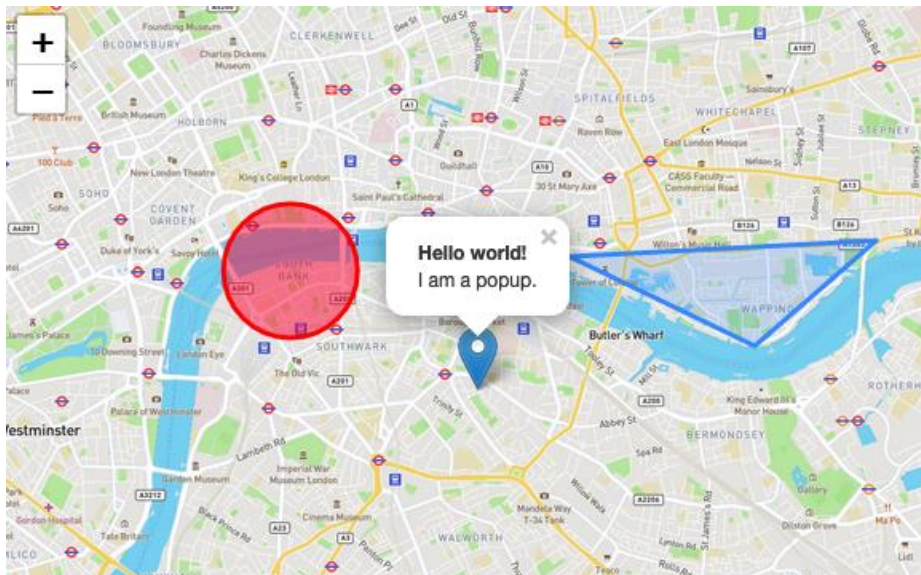


Figure 2. Leaflet Map Example. Source: leafletjs.com

JavaScript also allows for the use of the Maps API. The Maps JavaScript API lets you customize maps with your own content and imagery for display on web pages and mobile devices. The Maps JavaScript API features four basic map types (roadmap, satellite, hybrid, and terrain) which you can modify using layers and styles, controls and events, and various services and libraries.[10]

Python

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.[11]

```
PythonExample.py x
1  # Program published on https://beginnersbook.com
2
3  # Python program to calculate the sum of n Natural Numbers
4
5  # n denotes upto which number you want to calculate the sum
6  # for example, if n is 5 then the sum of first 5 natural numbers
7  num = int(input("Enter the value of n: "))
8  hold = num
9  sum = 0
10
11 if num <= 0:
12     print("Enter a whole positive number!")
13 else:
14     while num > 0:
15         sum = sum + num
16         num = num - 1
17         # displaying output
18     print("Sum of first", hold, "natural numbers is: ", sum)
```

Figure 3. Python Code Example. Source: beginnersbook.com

Python is one of the most popular coding languages in the world. As a result there is an endless amount of tutorials available online to help me develop my application should I choose to use Python.

Java

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine regardless of the underlying computer architecture. The syntax of Java is similar to C and C++ but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.[12]

```
package rentalStore;
import java.util.Enumeration;
import java.util.Vector;

class Customer {
    private String _name;
    private Vector<Rental> _rentals = new Vector<Rental>();

    public Customer(String name) {
        _name = name;
    }
    public String getMovie(Movie movie) {
        Rental rental = new Rental(new Movie("", Movie.NEW_RELEASE), 10);
        Movie m = rental._movie;
        return movie.getTitle();
    }
    public void addRental(Rental arg) {
        _rentals.addElement(arg);
    }
    public String getName() {
        return _name;
    }
}
```

Figure 4. Java Code Example. Source: researchgate.net

Java is also one of three languages used for Android Studio, along with Kotlin and C++. If I choose to use Android Studio, I will have to use Java for my application.

Java is more common for web development, where Python is more useful for data science.[13]

React Native

React Native is an open-source JavaScript framework, designed for building apps on multiple platforms like iOS, Android, and web applications, utilizing the very same code base. It is based on React, and it brings all its glory to mobile app development.

React Native was created by Meta Inc. It was created to solve the company's issue of maintaining two code bases for Facebook.

The benefits to using React Native include code reusability, faster development and better performance, although it comes with the drawback of being a new technology. This can lead to difficulty with finding support online.[14]

Flutter

Flutter is an open-source framework by Google for building natively compiled, multi-platform applications from a single codebase.[15]

Flutter consists of two important parts:

1. An SDK (Software Development Kit): A collection of tools that help develop applications. This includes tools to compile code into native machine code (code for iOS and Android).
2. A Framework (UI Library based on widgets): A collection of reusable UI elements (buttons, text inputs, sliders, and so on) that can be personalised for your own needs.

To develop with Flutter, you will use a programming language called Dart. The language was created by Google in October 2011, but it has improved a lot over these past years. Dart focuses on front-end development, and you can use it to create mobile and web applications.

Dart is a typed object programming language. You can compare Dart's syntax to JavaScript.[16]

Xamarin

Xamarin is open-source software that extends the .NET developer platform with tools and libraries specifically for building apps for Android, iOS, tvOS, watchOS, macOS, and Windows (UWP) primarily with C# in Visual Studio. Developers can re-use their existing C# code and share significant code across device platforms.[17]

Decision

After researching a variety of languages, I have decided to use React Native for my project. I think it will allow me to develop my application for as many devices as possible without increasing the workload as I will only have to maintain one code base.

Development Environment

There are many different independent development environments available. I managed to narrow it down to a few options for my application.

Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android app development. Java is still supported, as is C++.[18]

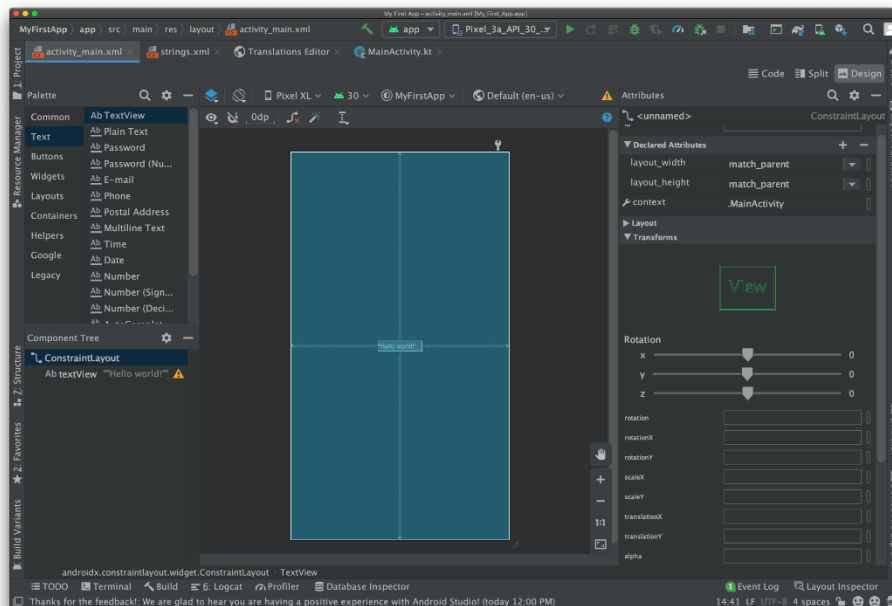


Figure 5. Android Studio User Interface. Source: developer.android.com

Android Studio would be a good choice for my application as it would allow me to develop for widely available, highly portable devices. A major drawback is that it would not allow me to develop for iOS.

Visual Studios

Microsoft Visual Studio is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class

designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle.

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists.[19]

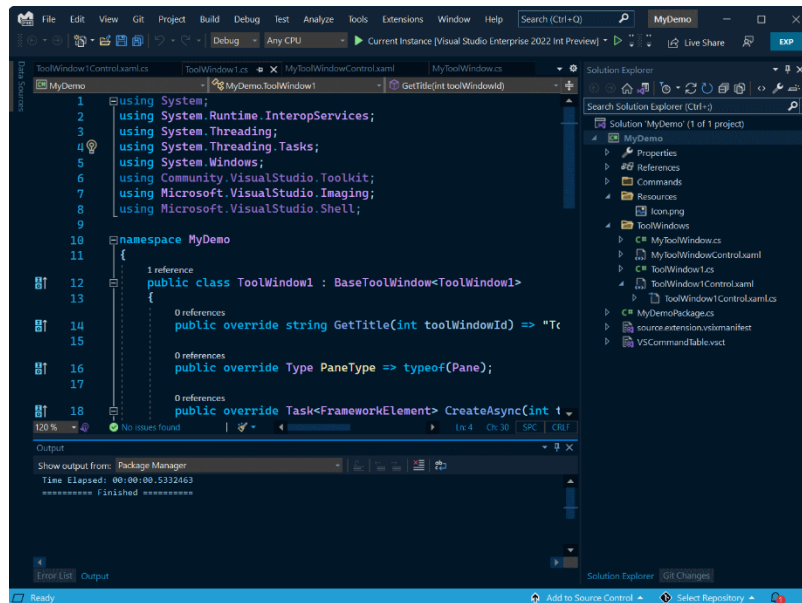


Figure 6. Visual Studios User Interface. Source: devblogs.microsoft.com

Using Visual Studios as my IDE would allow me to choose from a wider selection of programming languages than Android Studio.

Expo

Expo is a framework and a platform for universal React applications. It is a set of tools and services built around React Native and native platforms that help you develop, build, deploy, and quickly iterate on iOS, Android, and web apps from the same JavaScript/TypeScript codebase.[20]

Decision

I have decided to use Expo for my application. As I have decided to use React Native to develop the app, Expo feels like the correct choice to build the application as smoothly as possible.

Hardware

Laptops/Computers

While developing an application that runs on computers and laptops would allow for more processing power, it would be inconvenient for users to utilize tracking. Having students carry an open laptop around the campus would be too much of an inconvenience for most users and would alienate a large percentage of them.

Smartphones

Smartphones are the smarter option, as most of the staff and students have one. Putting the map application on phones will also allow for easier tracking if the user agrees to it. It would also allow for a smaller, faster, more focused application.

Algorithms

Dijkstra's Algorithm

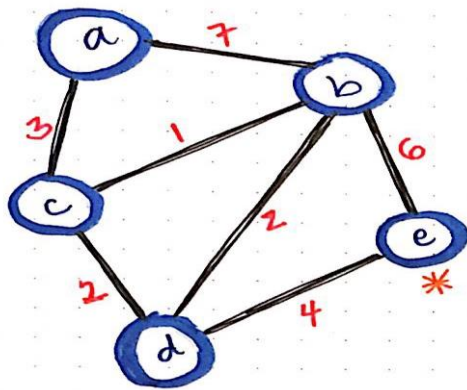
Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks. It was conceived by computer scientist Edsger W. Dijkstra in 1956 and published three years later.[1]

Dijkstra's algorithm makes use of the weights of the edges for finding the path that minimizes the total distance (weight) among the source node and all other nodes. This algorithm is also known as the single-source shortest path algorithm.[2] A drawback for this algorithm is that negative weights are not usable as they can affect the addition of weights. The source starts at weight 0 and the weights to other nodes are added on from there.

The algorithm starts at the node that you choose (the source node) and it analyses the graph to find the shortest path between that node and all the other nodes in the graph. The algorithm keeps track of the currently known shortest distance from each node to the source node and it updates these values if it finds a shorter path. Once the algorithm has found the shortest path between the source node and another node, that node is marked as "visited" and added to the path. The process continues until all the nodes in the graph have been added to the path. This way, we have a path that connects the source node to all other nodes following the shortest path possible to reach each node.[3]

Dijkstra's algorithm will find the shortest path between the start node and the target node every time. However, it will also find the shortest path between the start node and every other node in the graph. This leads me to the biggest drawback of this algorithm. It can be inefficient as it will waste time evaluating routes that could be ignored.

Example



Visited = [a, c, b, d]

Unvisited = [e]
 ↗ current vertex

VERTEX	SHORTEST DEST. FROM @	PREVIOUS VERTEX
a	0	
b	7 4	a c
c	3	a
d	5	b
e	1 9	b d

*The only node left to visit is (e). However, all of its neighbors have already been visited, so there's nothing for us to examine or update here!

Figure 7. Dijkstra's Algorithm Example. Source: pdfprof.com

A* Algorithm

A* (pronounced "A-star") is a graph traversal and path search algorithm, which is often used in many fields of computer science due to its completeness, optimality, and optimal efficiency. One major practical drawback is its space complexity, as it stores all generated nodes in memory. Thus, in practical travel-routing systems, it is generally outperformed by algorithms which can pre-process the graph to attain better performance, as well as memory-bounded approaches; however, A* is still the best solution in many cases.

Peter Hart, Nils Nilsson and Bertram Raphael of Stanford Research Institute (now SRI International) first published the algorithm in 1968. It can be seen as an extension of Dijkstra's algorithm. A* achieves better performance by using heuristics to guide its search.[4]

A* algorithm has 3 parameters:

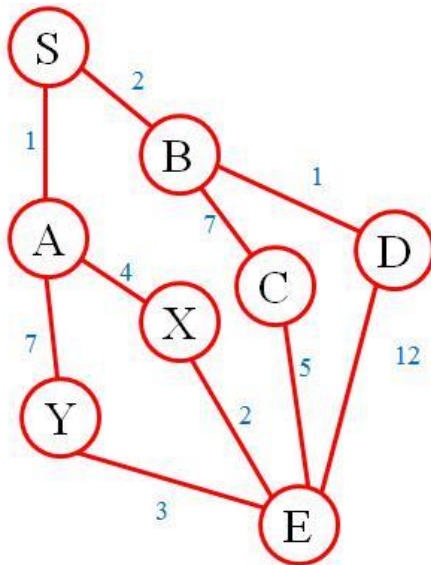
1. **g**: the cost of moving from the initial cell to the current cell. Basically, it is the sum of all the cells that have been visited since leaving the first cell.
2. **h**: also known as the heuristic value, it is the estimated cost of moving from the current cell to the final cell. The actual cost cannot be calculated until the final cell is reached. Hence, h is the estimated cost. We must make sure that there is never an over estimation of the cost.
3. **f**: it is the sum of g and h. So, $f = g + h$

The way that the algorithm makes its decisions is by taking the f-value into account. The algorithm selects the smallest f-valued cell and moves to that cell. This process continues until the algorithm reaches its goal cell.[5]

The biggest potential issue with selecting the A* algorithm is the need for a good heuristic function. The time it takes to provide the heuristic must not cancel out any time savings in the process of

pathfinding. In addition, the heuristic must not overestimate the cost of the path. If $h(n)$, where n is the cell, is always lower than (or equal to) the cost of moving from n to the target cell, then A* is guaranteed to find the shortest path.

Example



■ Values for h :

A:5, B:6, C:4, D:15, X:5, Y:8

Expand S

$\{S,A\} f=1+5=6$

$\{S,B\} f=2+6=8$

Expand A

$\{S,B\} f=2+6=8$

$\{S,A,X\} f=(1+4)+5=10$

$\{S,A,Y\} f=(1+7)+8=16$

Expand B

$\{S,A,X\} f=(1+4)+5=10$

$\{S,B,C\} f=(2+7)+4=13$

$\{S,A,Y\} f=(1+7)+8=16$

$\{S,B,D\} f=(2+1)+15=18$

Expand X

$\{S,A,X,E\}$ is the best path... (costing 7)

Figure 8. A* Algorithm Example. Source: Stack Overflow.com

Decision

The A* algorithm is more efficient than Dijkstra's algorithm. By using a heuristic function to provide an estimate of the cost of the path between each node and the target node, it can make better choices about the next path to take and will find the shortest path faster.

For these reasons I have decided to use the A* algorithm for my map application. If I take the time to use a good heuristic function and make good estimates for the weights between nodes, A* will give me a much more efficient application.

Similar Products

Penn State Map

The Penn State interactive map is the first campus map I have researched to get an idea for my own application. The Penn State map has a clean user interface. It allows users to look around a map of the campus and click on various buildings for more information. It also provides a list on the left side of the screen that allows users to choose different facilities to pin on the map.[6]

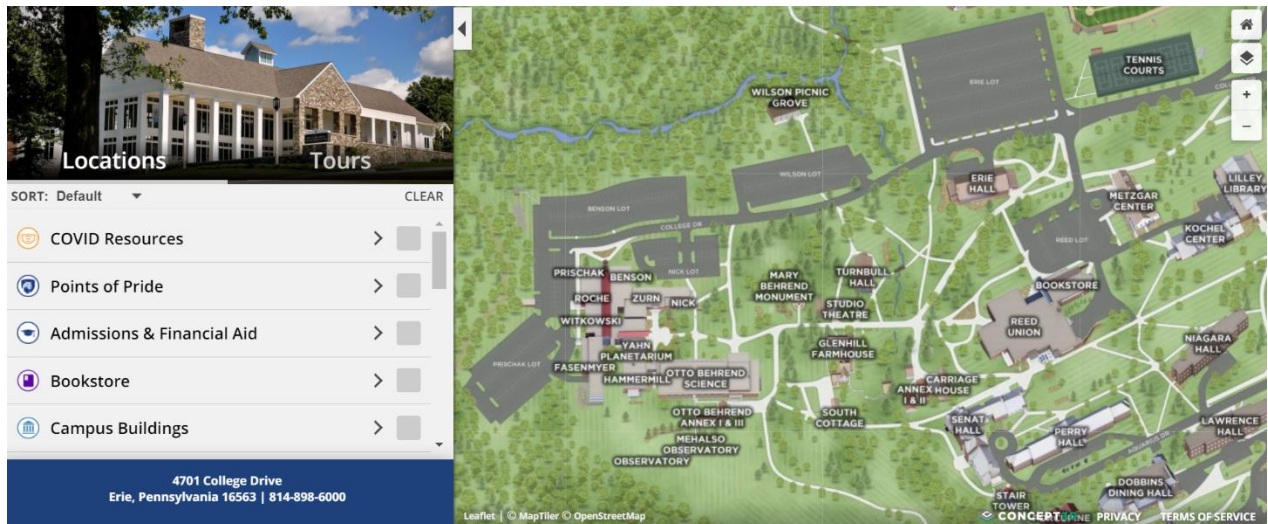


Figure 9. Penn State Interactive Map. Source: behrend.psu.edu

While this map is very nice, it doesn't satisfy the requirements of my project. It does not provide any pathfinding between classrooms, which is a primary feature for my application.

MazeMap

MazeMap is a company that builds maps for clients. They have made map applications for many buildings, including college campuses. They have a very full feature list, which includes:

- Pathfinding
- Indoor maps and positioning
- Multiple tailored map views
- Scalable maps for additional buildings and outdoor areas

The maps include indoor and outdoor mapping and positioning, making them ideal solutions.[7]

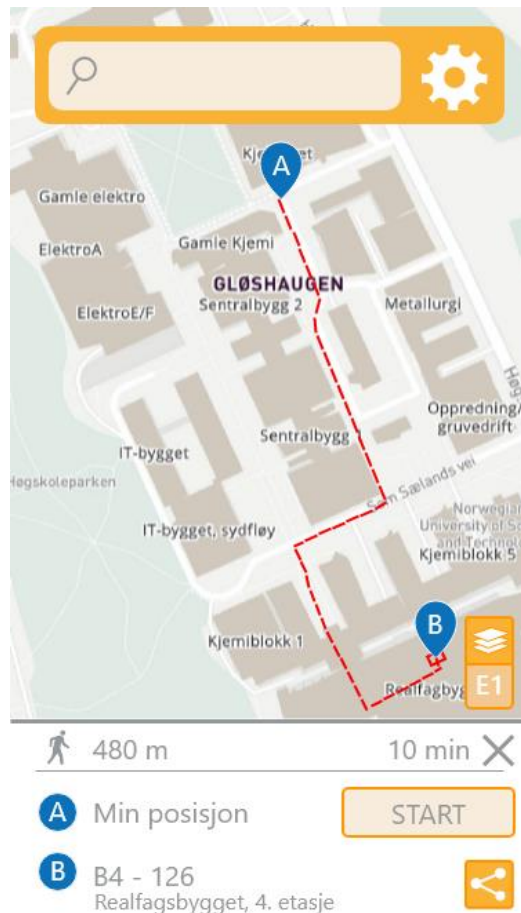


Figure 10. MazeMap User Interface. Source: behance.net

While this is how I would like my application to look upon completion, I cannot use MazeMap directly. The first problem is that they are a paid service. The second issue is that by using them, I would not have enough work to do myself. It is up to me to code and build this project, not MazeMap. I can however use them as a visual guide, as well as a goal for where I would like my map to be upon completion.

Conclusion

In conclusion, I have decided to develop my map application using Expo. This will allow me to build, run and test code on my phone.

I have also decided to write the code using React Native. It will allow me to maintain a single codebase which I can run on a plethora of different devices.

Finally, I will use A* as my pathfinding algorithm. I think it will allow for a quicker, more accurate route to be plotted than other pathfinding algorithms. I will make sure to take the time to use a good heuristic function and to make accurate estimates for the weights between nodes.

References

Table of Figures

Figure 1. JavaScript Code Example. Source: medium.com	2
Figure 2. Leaflet Map Example. Source: leafletjs.com	3
Figure 3. Python Code Example. Source: beginnersbook.com	4
Figure 4. Java Code Example. Source: researchgate.net	4
Figure 5. Android Studio User Interface. Source: developer.android.com	6
Figure 6. Visual Studios User Interface. Source: devblogs.microsoft.com	7
Figure 7. Dijkstra's Algorithm Example. Source: pdfprof.com	9
Figure 8. A* Algorithm Example. Source: Stack Overflow.com	10
Figure 9. Penn State Interactive Map. Source: behrend.psu.edu	11
Figure 10. MazeMap User Interface. Source: behance.net	12

Bibliography

- [1]"Dijkstra's algorithm - Wikipedia", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm. [Accessed: 22- Oct- 2021].
- [2]N. Tyagi, "What is Dijkstra's Algorithm? Examples and Applications of Dijkstra's Algorithm", Analyticssteps.com, 2020. [Online]. Available: <https://www.analyticssteps.com/blogs/dijkstras-algorithm-shortest-path-algorithm>. [Accessed: 22- Oct- 2021].
- [3]E. Navone, "Dijkstra's Shortest Path Algorithm - A Detailed and Visual Introduction", *freeCodeCamp.org*, 2020. [Online]. Available: <https://www.freecodecamp.org/news/dijkstras-shortest-path-algorithm-visual-introduction/>. [Accessed: 22- Oct- 2021].
- [4]"A* search algorithm - Wikipedia", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/A*_search_algorithm. [Accessed: 25- Oct- 2021].
- [5]"What is the A* algorithm?", Educative: Interactive Courses for Software Developers. [Online]. Available: <https://www.educative.io/edpresso/what-is-the-a-star-algorithm>. [Accessed: 25- Oct- 2021].
- [6]"Campus Map", Penn State Behrend. [Online]. Available: <https://behrend.psu.edu/map>. [Accessed: 28- Oct- 2021].
- [7]"Indoor Maps and Wayfinding MazeMap", Mazemap.com. [Online]. Available: <https://www.mazemap.com/>. [Accessed: 29- Oct- 2021].
- [8]"JavaScript - Wikipedia", En.wikipedia.org. [Online]. Available: <https://en.wikipedia.org/wiki/JavaScript>. [Accessed: 02- Nov- 2021].
- [9]"Leaflet — an open-source JavaScript library for interactive maps", Leafletjs.com. [Online]. Available: <https://leafletjs.com/>. [Accessed: 02- Nov- 2021].
- [10]"Overview | Maps JavaScript API | Google Developers", Google Developers. [Online]. Available: <https://developers.google.com/maps/documentation/javascript/overview>. [Accessed: 02- Nov- 2021].

- [11]"Python (programming language) - Wikipedia", En.wikipedia.org. [Online]. Available: [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language)). [Accessed: 04- Nov- 2021].
- [12]"Java (programming language) - Wikipedia", En.wikipedia.org. [Online]. Available: [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)). [Accessed: 08- Nov- 2021].
- [13]W. Rowe and J. Johnson, "Python vs Java: What's The Difference?", BMC Blogs, 2020. [Online]. Available: <https://www.bmc.com/blogs/python-vs-java/>. [Accessed: 11- Nov- 2021].
- [14]P. Paterska, "What is React Native and When to Use It For Your App? (Updated) - elpassion.com", *Elpassion.com*, 2021. [Online]. Available: <https://www.elpassion.com/blog/what-is-react-native-and-when-to-use-it>. [Accessed: 13- Nov- 2021].
- [15]"Flutter - Build apps for any screen", *Flutter.dev*. [Online]. Available: <https://flutter.dev/>. [Accessed: 13- Nov- 2021].
- [16]G. Thomas, "What is Flutter and Why You Should Learn it in 2020", freeCodeCamp.org, 2019. [Online]. Available: <https://www.freecodecamp.org/news/what-is-flutter-and-why-you-should-learn-it-in-2020/>. [Accessed: 13- Nov- 2021].
- [17]"Xamarin - Wikipedia", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/Xamarin#Xamarin_platform. [Accessed: 14- Nov- 2021].
- [18]"Android Studio - Wikipedia", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/Android_Studio. [Accessed: 15- Nov- 2021].
- [19]"Microsoft Visual Studio - Wikipedia", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/Microsoft_Visual_Studio. [Accessed: 16- Nov- 2021].
- [20]"Introduction to Expo - Expo Documentation", Expo Documentation. [Online]. Available: <https://docs.expo.dev/>. [Accessed: 16- Nov- 2021].

Appendix

Plagiarism Form

Projects & Plagiarism

All projects are individual projects. The project is expected to be all your own work. Under certain circumstances, you may be able to incorporate material from other sources. This "incorporation" needs to be discussed with, and agreed upon by, lab supervisor and all such material must be referenced appropriately.

- Plagiarism is defined as " ... *presenting someone else's work as if it were your own, whether you mean to or not. 'Someone else's work' means anything that is not your own idea, even if it is presented in your own style. It includes material from books, journals or any other printed source, the work of other students or staff, information from the Internet, software programs and other electronic material, designs and ideas. It also includes the organization or structuring of any such material ...* " [taken from: Victoria University of Wellington, New Zealand, <http://www.victoria.ac.nz/home/glossary.aspx#p>].
- What this means is that it is wrong to copy another's work or download material for incorporation into your project from the Internet (without the express permission of your supervisor). Even if you take someone else's algorithm/code and rewrite it, this must be documented in your program to state where you got the original algorithm from. Failure to do so is considered plagiarism. If you are in any doubt, always ask your lab supervisor.
- Plagiarism is a **serious academic offence** (equivalent to cheating on an exam) and the consequences of being found guilty of plagiarism are severe.
- Students can use the internet to research information on their projects, they can ask lab supervisors for help with their project and they can also discuss problems with their fellow class mates.
- Students should be able to explain fully their code and may be asked to do so by one of your lab supervisors.

Work submitted for assessment which does not include this declaration will not be assessed.

DECLARATION

*I declare that all material in this submission e.g. thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.

*I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.

*I have provided a complete bibliography of all works and sources used in the preparation of this submission.

*I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offence.

Student Name: (Printed) ADAM STRAHAN

Student Number(s): COO216499

Signature(s): Adam Strahan

Date: 26/10/21

Please note:

The Institute regulations on plagiarism are set out in Section 10 of Examination and Assessment Regulations published each year in the Student Handbook.