Team 8
Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel

Project Name:

HawkQuest

Project Synopsis:

User-friendly digital mapping iOS app of KU's campus that finds the most efficient path between locations on campus.
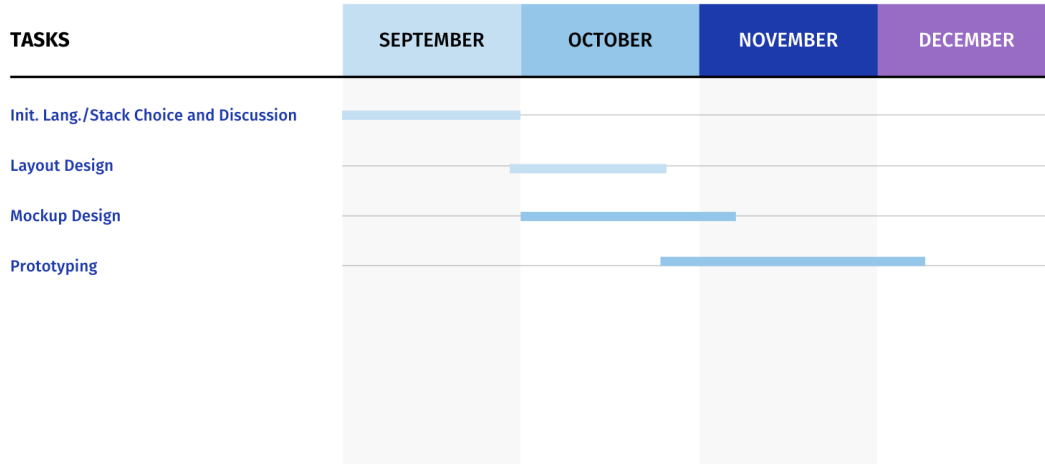
Project Description:

Our project is being undertaken because of the lack of clarity when navigating KU's Lawrence campus on foot, especially for students and visitors who are new to KU's layout and need more guidance. There is a great opportunity to streamline an interface that allows students to navigate their way through campus with the utmost ease and efficiency, uniquely fit and shaped around the buildings in the university grounds. The results of this project will be an application specifically made for the Lawrence campus, which is free from the extra hassle and unneeded features the more general mapping apps, such as Google Maps or Waze, have to offer; these extra tools and options aren't needed when you want to simply navigate around campus on foot campus. The application will not necessarily need the user to have their location services on, which can drain battery and slow performance. To handle this it will allow users to input the starting location if they opt to not use location services, and where they would like to go.

Project Milestones:

- Initial Language/Stack Choice Discussion - September 2021
- Layout Brainstorming and Design - October 2021
- Mockup Design - October 2021
- Prototyping - November/December 2021
- Front-end and Back-end Development - January-March 2022
- Unit Testing - February-March 2022
- Functioning Connection of Front-end and Back-end - March 2022
- Testing with mock data - March 2022
- Actual Campus Data Entry - April 2022
- Completed Node Map of Campus - April 2022
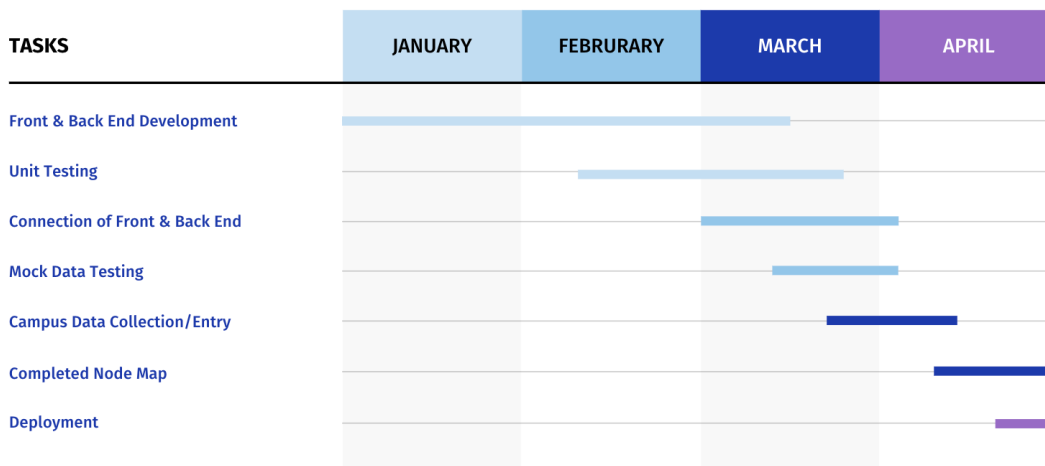- Deployment to mobile market (iOS) - April/May 2022

Team 8
Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel

EECS 581 Group 8

HAWKQUEST GANTT CHART
FALL 2021 SEMESTER

TASKS	SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER
Init. Lang./Stack Choice and Discussion	▬			
Layout Design		▬		
Mockup Design		▬		
Prototyping			▬	

EECS 581 Group 8

HAWKQUEST GANTT CHART
SPRING 2022 SEMESTER

TASKS	JANUARY	FEBRURARY	MARCH	APRIL
Front & Back End Development	▬			
Unit Testing		▬		
Connection of Front & Back End			▬	
Mock Data Testing			▬	
Campus Data Collection/Entry			▬	
Completed Node Map				▬
Deployment				▬

Team 8

Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel

Project Budget:

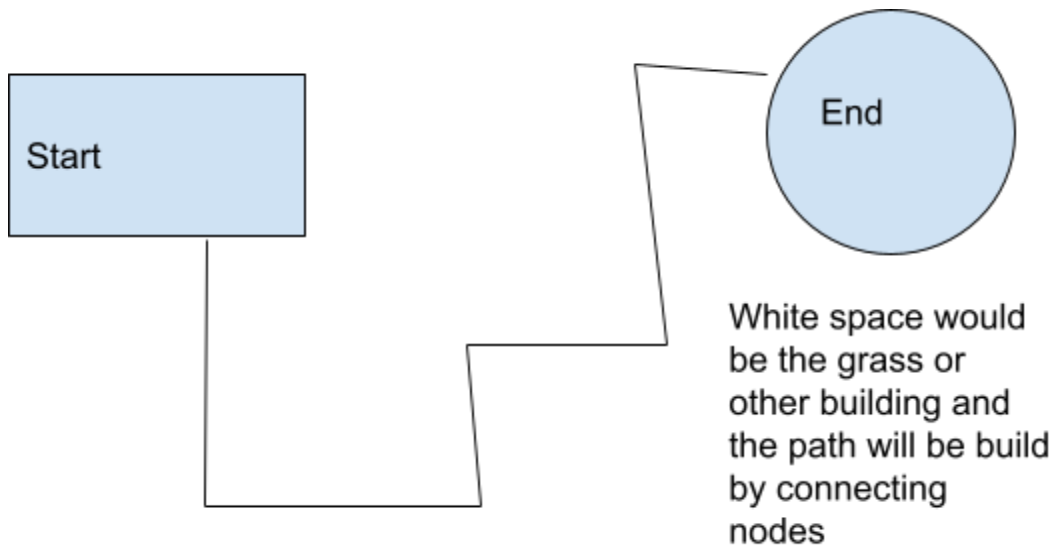- iOS development license: **$99** - Required any time before publishing to Apple App Store

Preliminary Project Design:
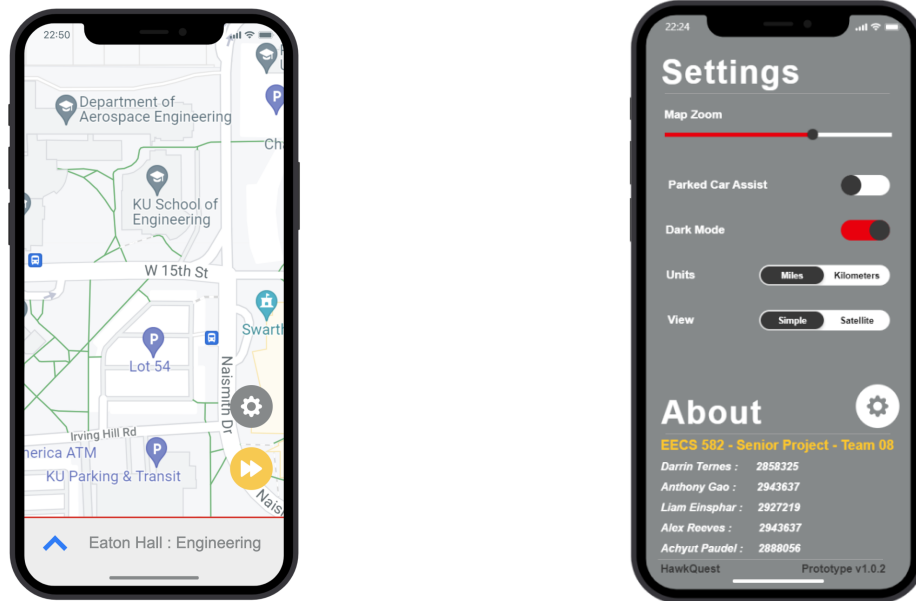
The software for HawkQuest is designed in a way that it is usable through iOS. The UI is designed through the usage of Swift and instead of the drag and drop feature we will be using the SWIFT UI which gives a cleaner overview of the product. We will be using the MVC pattern. This is the model where The model updates the view and the view shows what the user sees. The user then can use the view to control and manipulate the model. As shown below in the diagram. By using Apple's programming language, we increase the reach of our product, but constraint ourselves to the features and attributes of Swift and or XCODE. We can reach a large number of mobile users by choosing iOS as our development platform, however, we consequently miss out on the users on desktops, laptops, Androids, and other devices. In addition, this is our first time coding in Swift and using XCode. Our lack of familiarity limits the potential features of the application. Using Python or JavaScript would be much easier for our group to code. Not only would that save time, but it would allow us to push the complexity of the product. Each teammate's ability to learn and utilize Swift will be a significant indicator of how future stages of development may be. If the whole team works to understand the programming language and development environment, then collaborating as well as designing becomes much simpler at a group level. In addition, it would be helpful to the team to browse through the common Swift libraries which could be helpful later in the coding process.

Team 8
Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel
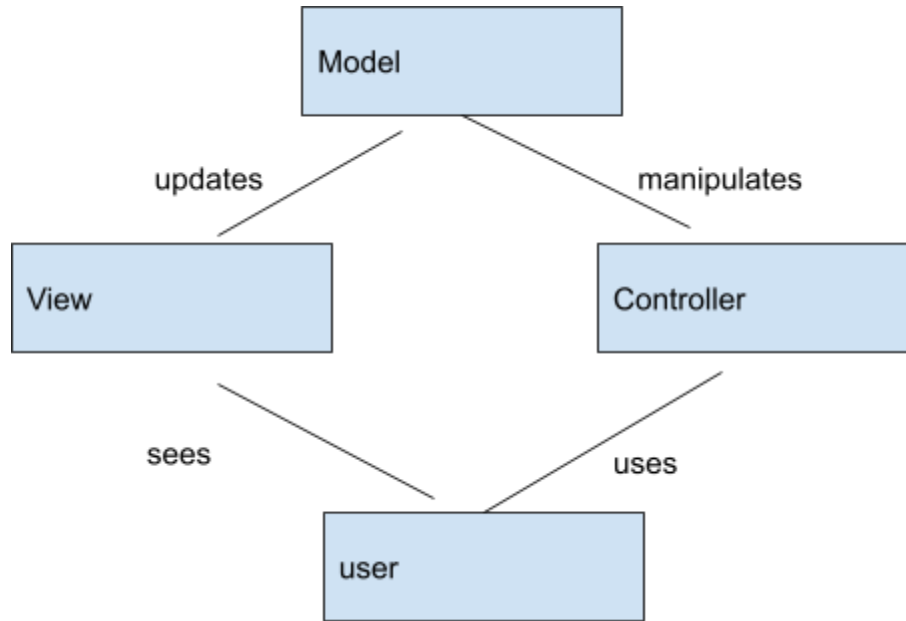
The backend of the whole software will be using Swift dictionaries as we won't need too much data storage. At most we will store a couple images around campus to fulfill our current goals. This allows our backend to be clean and we can call on the certain image to create nodes in order to navigate around campus by connecting the node. The nodes are going to be a grid on top of each individual image. Then we could connect the nodes in each image based on the png or image store in each array in Swift. The UI will reflect that in the iOS app. Another thing we would use is backtracking and using recursion to trace through the nodes from the start to the end. Otherwise, manually tracing from building to building will cause a lot of issues as we might make mistakes. This will greatly increase the efficiency and accuracy of the iOS app. This also makes it scalable. In addition, we added in Dark Mode and Unit conversions for a cleaner look and caters toward users internationally.

White space would be the grass or other building and the path will be build by connecting nodes

Team 8
Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel

One of the major facets of this project that is a limiter is time. Since we only allot approximately 9 months to plan, design, write, and present the project, we are only able to realistically map the Lawrence campus of the University of Kansas; instead of including the greater Lawrence area or even Edward's campus in Overland Park. We have accounted for this constraint in our preliminary designs and are going to make a more detailed layout of the Lawrence campus, including footpaths and sidewalks to truly find the most optimal routes, making it especially catered to KU students attending the Lawrence campus. Time also keeps us from going into a more in-depth analysis and mapping of each of the buildings located at the Lawrence campus which would yield the most detailed, clear, and efficient path. This also limits how much detail we can go into the XCODE IDE design alongside our Swift code; instead, we have to use a templated format for our XCODE design. We also had to reformulate and revise our goals for the project given our limited time and decided that we would not be able to do a separate Android app and Web app without sacrificing some of the details and features we are planning to put into the iOS application. One other idea we had was about physically mapping some of the more convoluted footpaths around campus for better precision but there is just not enough time given our other responsibilities and obligations.

In order to implement a map we have considered utilizing Google Maps API. Using Google's API will cost the user per call to the API. This can add up and become large if we are not careful to closely monitor the frequency of location updates. This means there is the potential of being limited by our available resources. However, closer analysis of the pricing for the API revealed that the cost is more than fair. That is to say that it shouldn't cause us any problems in our development process. In addition, the features of the map itself is limited to the features currently available within Google Maps' API. Google Maps' API also gives the programmer the ability to draw and mark up the map. We are constrained in the data we are able to access. This can be more useful when we implement our idea of adding more detailed pathing. We are able to manually update this through API calls. We are not completely sure of the disadvantages of this API, but they will present themselves as we get closer to the coding part of the development cycle.

Along with the business side of the operations, we must consider how we will distribute the application after completion of development. For distribution on Apple's App Store, developers must sign up for a development account. This comes with a price tag of $99. This is one of the few financial stresses. Without having a platform in which people can download the app, this project becomes nothing more than busy work in the classroom. Selling our app on Apple's store means we must adhere to the terms and conditions of their app store. This can include technical and ethical measures to keep developer's in check.

Ethical Issues:

There are few possible ethical issues with HawkQuest since it is only providing users with objective map data and path information and does not require a user account or any storage of user data. Unlike mapping apps that cover large areas such as countries/continents we will not have to worry about political implications of locations and borders, rapidly changing routes and layouts, and focus/avoidance of any areas on the map. The area being mapped is relatively small-scale and the applications sole purpose is to calculate and display the shortest path from one location to another using the footpaths around the KU campus.

- **The use of a users GPS location data**

- ○ As stated above this application will not require the use of user accounts or the storage of usage data, taking this into consideration the use will be anonymous and will not require users to input personal information or consent to collection of usage and location data relating to their trips. If this is ever to change, in terms of collection of user data for application improvement or commercial use, it will be necessary to notify all users and work on legal agreements and ensure the safety of any sensitive information that may be collected through the use of the app. At this point there are no plans to store any information other than the data used to calculate and display paths.
- **Accessibility of the application**
  - ○ Since the focus of this project/application is the ability to find and display the shortest footpath to a desired location on campus the final product may not be fully accessible to all users such as users with disabilities that would not allow them to use the app as originally designed or users that are not able to read English. Implementation of accessibility features, for example, descriptive text compatible with screen readers for the visually impaired and global translation text that would allow information to be displayed in languages other than English, could be added in at a later date but there are not plans for features such as these to be implemented at this time.

Intellectual Property Issues:

We are mostly concerned about three potential intellectual property issues:

- ***The use of the name: "HawkQuest"***

We highly doubt that this would create any conflict, but regardless needs to be addressed as potential conflict. The University of Kansas is already using the name "HawkQuest" at their career center website: https://career.ku.edu/hawkquest. Their functionality of HawkQuest is a quiz/survey that students can use to determine a career path they might be interested in. We dont believe that any conflict will result from using the name, as the app we are building directly benefits the university and the students attending. The name also is not being used for any *major* function at the university currently.

HawkQuest

The Career and Major Exploration Quiz for Jayhawks

HawkQuest is a brief, informal quiz specifically for KU students (or

---

- ● ***We will be using a map not created by us***

In order for our app to work, we will need a map of the university's roads and paths. The task of creating one from scratch that is accurate to gps points is far outside the scope of our abilities. Therefore we will need to use one that has already been made. At this time it is unclear where we will be obtaining the final map that we use in the app, but it is possible that we may need to get permission or even a license to use it. To avoid this, we could use an open-source map that wouldn't require any of these permissions. Many of these already exist, such as: OpenStreetMap, that already map out many of the campus features.

Team 8

Liam Einspahr, Alex Reeves, Anthony Gao, Darrin Ternes, Achyut Paudel

---
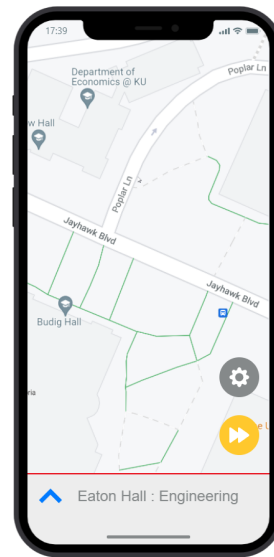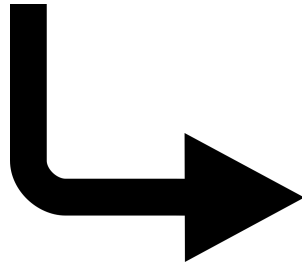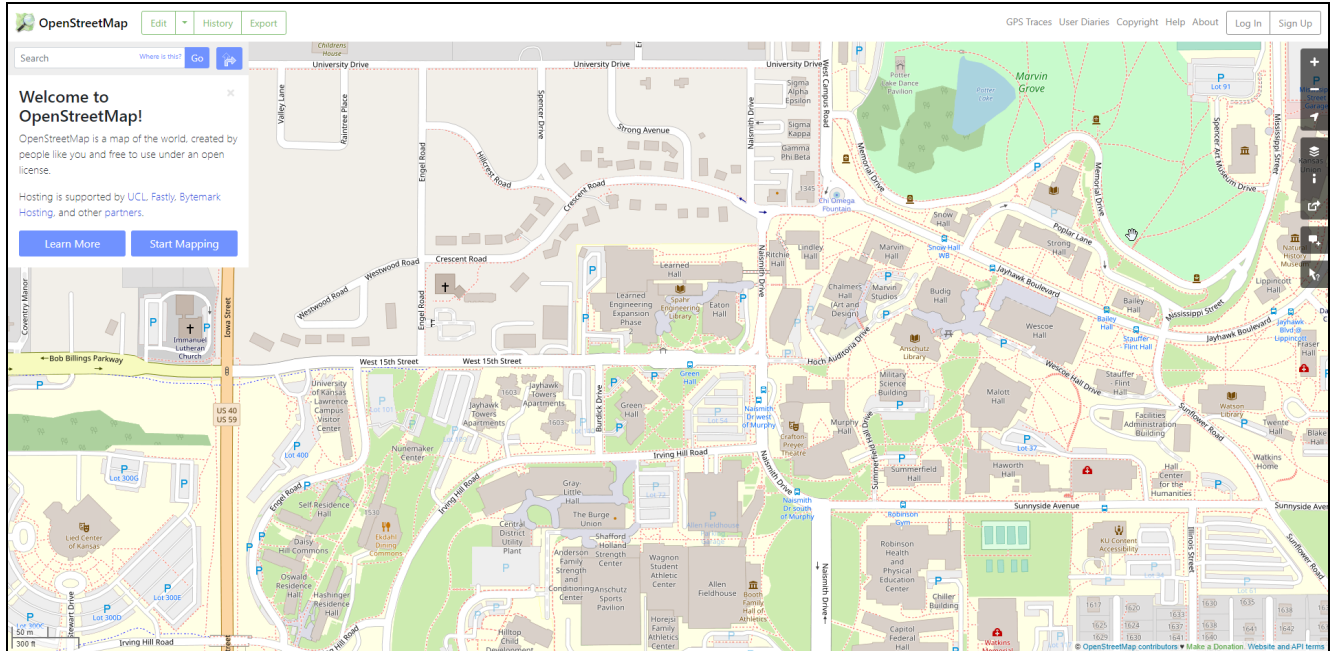
- ### *Similarities in User Interface layout*

There are many apps that already exist with a similar design layout for the user interface that we have prototyped. There are only so many ways to make an app functional and unique, but some companies have been known to sue over things such as layouts and aesthetics (eg: Apple). The current prototyped user interface for HawkQuest looks somewhat similar to the user interface for "Google Maps". In particular the two buttons in the bottom right hand corner of the screen. It may be a good idea to change our current design with the buttons
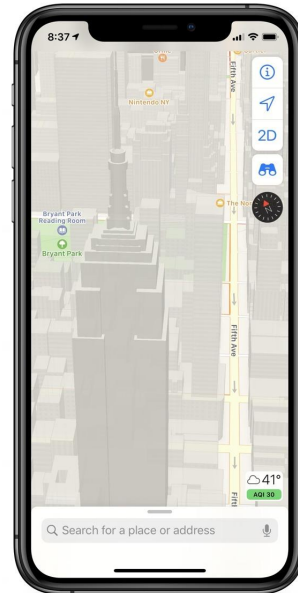
spaced more apart or on the other side of the screen to avoid any complications with the layout of our app.
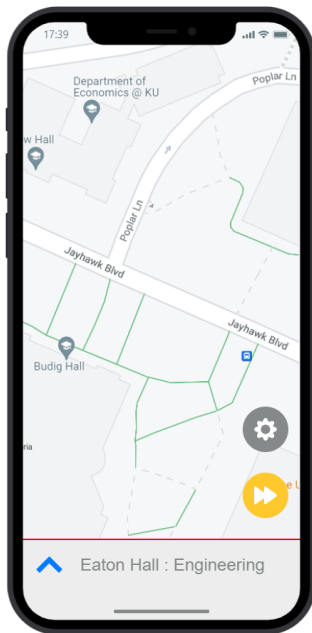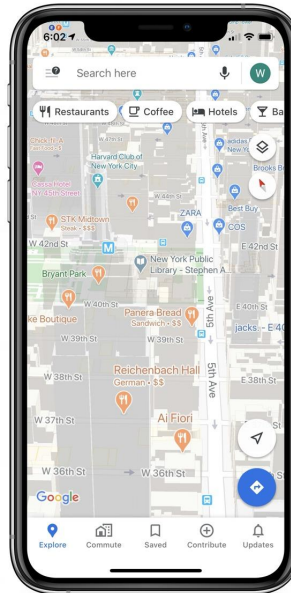
HawkQuest | *Google Maps* | *Apple Maps*

Change Log:

- **Decided to deploy exclusively on iOS**
  - After having group discussions and doing research on deployment of an application on multiple platforms we decided to design the application for iOS specifically and have been learning about mobile development with Swift
- **Removed the need for server/cloud time**
  - Since our application will not need a large amount of data to run and find directions accurately we decided that we will not need remote hosting for databases and since there is no longer the need for hosting for a website this cost and extra development consideration can be removed from our plans.