Project Name: WALK CHALK

Team number 17
Team members: Matthew Eagle, Zach Harris, Doug Newman, Xianye Zhou, William Graham

Synopsis

Create an easy to use KU campus navigation system that shows the fastest path on a real 3D/360° photo.

Project Description

The goal is to create a better navigation system using 3D/360° photos. For those unfamiliar with an area, use Walk Chalk instead of straining to find road signs to follow a map route. Typical gps navigation systems still force the user to read an actual map. They just highlight a specific road to walk along. Walk Chalk's goal is to improve the user experience by enhancing the traditional map apps with our outdoor and indoor navigation functions. Simply choose your location and the location you want to go; the key words can be as precise as specific rooms. It handles knowing exactly where the mysterious class rooms on campus are and the short cut paths unknown by new students or visitors. The targeted scope of the project is specifically for navigation around the University of Kansas campus. The audience most impacted by the creation of this software is new KU students with no knowledge of the layout of the campus. After the formation of a KU navigation database, the project could fairly easily be extended for use anywhere.

Project Milestones/Completion Dates

First semester:

- Project Scope Defined / October 11
- Technological Survey and Requirements Decided / October 25
- Initial code design / November 10
- Programming Interfaces and Technologies Specified / November 8
- Project Management and Responsibilities Delegated / November 29
- Initial UI design / Dec 1

Second semester:

- Retrieval of images from server to Unity / Feb 15
- Complete alpha version, start server testing / March 1

- Complete navigation algorithm / March 28
- Complete beta version, start testing / Apr 1
- Finish testing and deploy to store / May 1

Budget

3D camera $599

Camera stand backpack $149

We will develop in Unity (personal version) and deploy to the Google Play Store and Apple App Store.

Work Plan

- Navigation Algorithm
  - Doug Newman
- Virtual Space Building - UI Overlay
  - Matthew Eagle
- Map Grid Building
  - William Graham
  - Xianye Zhou
- User Input Handling
  - Zach Harris
- UI Design
  - Doug Newman
- Taking pictures
  - Everyone

Project Design

The application aims to provide visual information in the form of 3D/360° photos to the user to assist in the navigation of the KU campus. There will be four main components to the application:

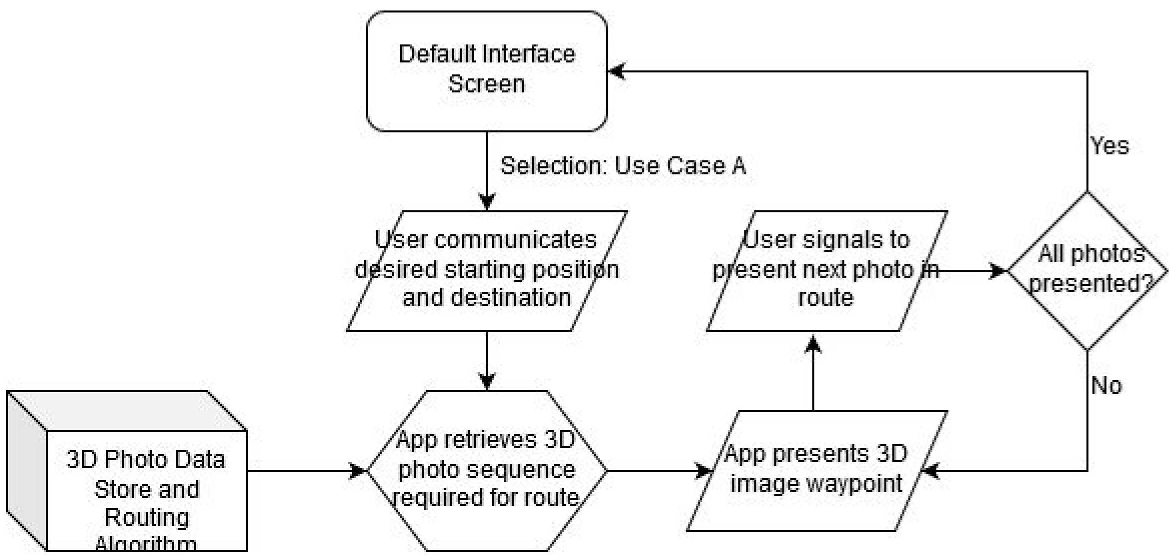
- User Interface
- 3D/360° Photo Data Structure

- Routing Algorithm
- Search function

We discuss each of these in turn. But first, we discuss the two primary use cases for the application.

Use Case 1: “Point A to Point B”

The user will be able to enter two locations (a “Point A” and “Point B”), and the application will present a series of 3D/360° photos that guide the user from the initial location to the desired destination. The photos will be presented in a particular sequence decided by the routing algorithm and taken from the 3D/360° photo structure. The user will signal to the application via touch input on arrows displayed on screen that it is ready to be presented with the next photo when they have reached that photo’s *waypoint* (see below). Once all of the photos in the sequence have been presented and the user progresses past the last photo, the application will return to its default interface screen.

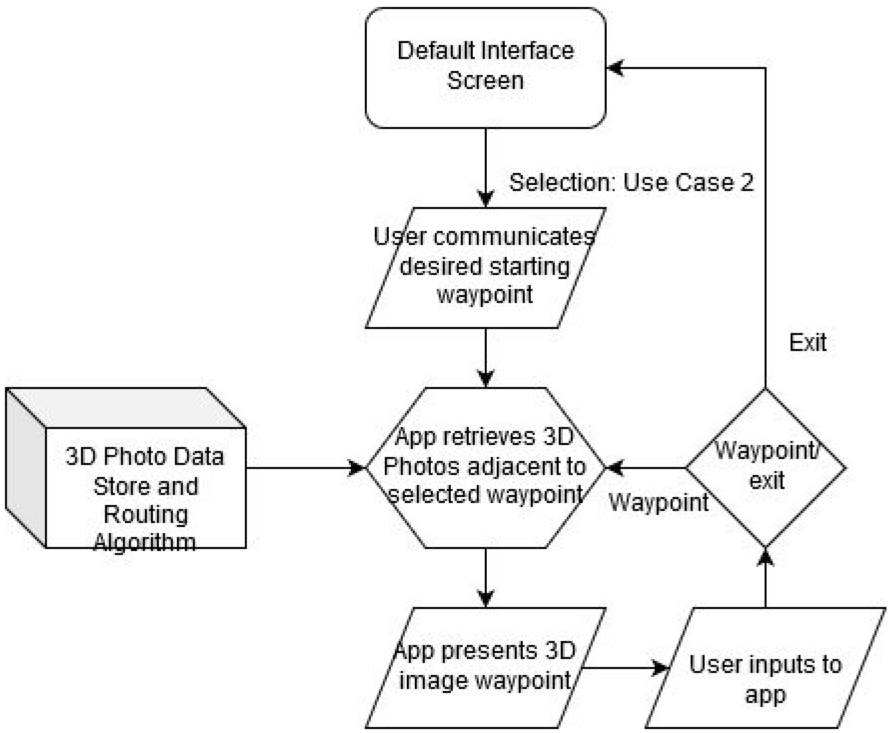


Use Case 1: “Point A to Point B”

Rough Idea of typical use case 1. The red icon in the top left is a back button that returns the user to the search screen. The END button in the bottom left corner takes the user to the Map View. The button in the bottom right re-centers the center of view to the direction of route. The red arrow in the middle demonstrates the optimal route.

Use Case 2: "Free Roam"

Alternatively, the user will be able to select any particular photo waypoint and navigate in any direction to any other waypoint that they wish, again via touch input on arrows displayed on the screen. In this case, the user may navigate around for an arbitrary amount of time, and the session will end when the user decides that they are done navigating, via some close button on the screen. When the user signifies that they no longer wish to navigate, they will be returned to the default interface screen, as in the first use case.



Use Case 2: "Free Roam"

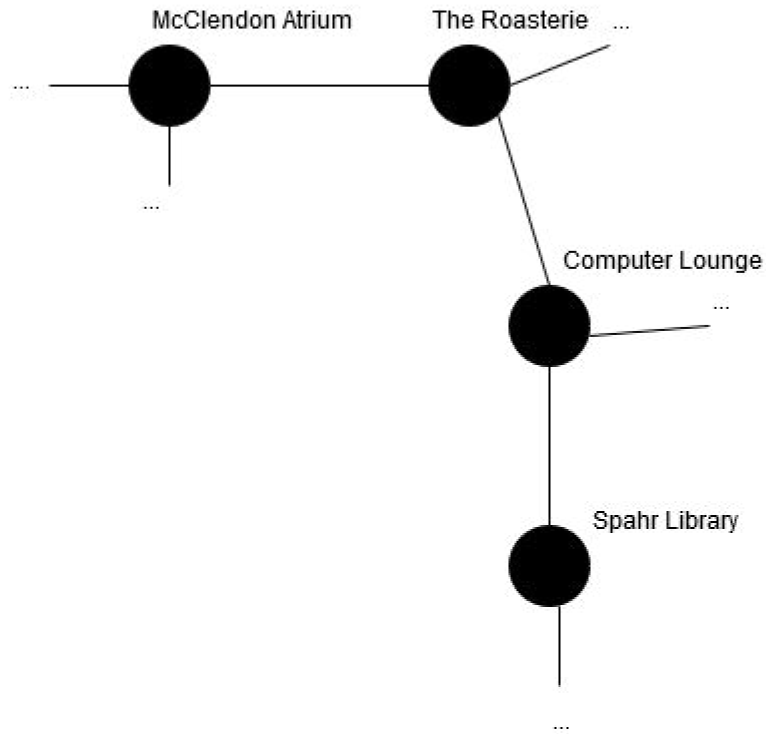
We now go into more detail about the constituent components of the application.

Component: User Interface

Upon opening the app, the user will be presented with the Default Interface Screen, which will consist of a Search Bar at the top of the screen and a map of campus filling up the rest of the screen. The search bar functions as the primary means of user use case selection (i.e., Free Roam and Point A to Point B). When the user touches the search bar, they will be taken to a screen where they can choose either to navigate in Free Roam mode or Point A to Point B mode, and will be prompted for appropriate location information for each mode. From here, the application will function primarily as outlined in the use case diagrams, though we do plan to add more functionality as we make more progress on the project.

When the user is navigating, they are viewing 3D/360° photos. Given a particular 3D/360° photo, it will be adjacent (see 3D/360° photo structure) to other 3D/360° photos of a nearby location. The user traverses from one photo to the next via arrows that will be placed in the 3D/360° photo at positions relative to the adjacency of the photo that arrow leads to. Once the user touches an arrow, the photo that the arrow leads to is displayed on the screen, upon which the user can click another photo to navigate to another nearby location. In this way, the user may navigate to any point on campus for which there is a 3D/360° photo of that location.

Component: 3D/360° Photo Data Structure
The data structure storing the photos lies at the heart of the application. We plan on using a linked graph structure to store our 3D/360° photos. Each node in the graph will contain, among other things, some geolocation information, a 3D/360° photo at the corresponding location, and a timestamp for when the photo was taken. Adjacent nodes in the graph correspond to locations which are nearby each other, and in each use case where the user is navigating through the photos, users can only navigate to photos which are adjacent to the current photo in the abstract graph structure. We plan on taking a good number of photos to increase the granularity of the structure, and storing the photos and associated data on a server to be downloaded by the application for its needs. We have not yet decided how the server and the user application will communicate to provide and request data, but we will most likely develop our own application-layer protocol to regulate and facilitate such communication. Additionally, we have yet to decide on the formatting of each piece of node data, but this is the basic idea of how data will be structured.



Example snippet from anticipated graph structure; one 3D photo per vertex

Component: Routing Algorithm
The routing algorithm is central to the Point A to Point B use case. It is used to select a *path* of nodes from the graph structure which, for now, will constitute the *shortest path* from the starting location to the desired destination. Such algorithms (such as Dijkstra's Algorithm) are already known, and so we plan on employing one of the already-existing algorithms for this component.

At some point along the development, we plan on making the two use cases mendable, in that they may transition from one to the other seamlessly while navigating. Additionally, we would like to implement some "rerouting" algorithm which will decide on a different path to be used if the shortest path is infeasible for whatever reason, as well as some additional criteria for "optimal" paths (such as accessibility, etc.).

Component: Search Function
The search function is a convenience (although necessary) feature for newcomers to campus. Part of the implementation of the search functionality will be in "tags" that will be part of associated node data. Additionally, we will utilize an already-existing string searching algorithm suited to the task of matching with the tags in the nodes.

Supported Operating Systems; Languages

In order to be most useful to new students and visitors, we would like our app to be useable on all popular phone types. This means we need compatibility with the largest mobile phone operating systems, including iOS, Android, and Windows. Luckily, all three of these operating systems are supported by Unity, which we plan on using for our project. Because we're going to be using Unity, the programming language of choice will be C#.

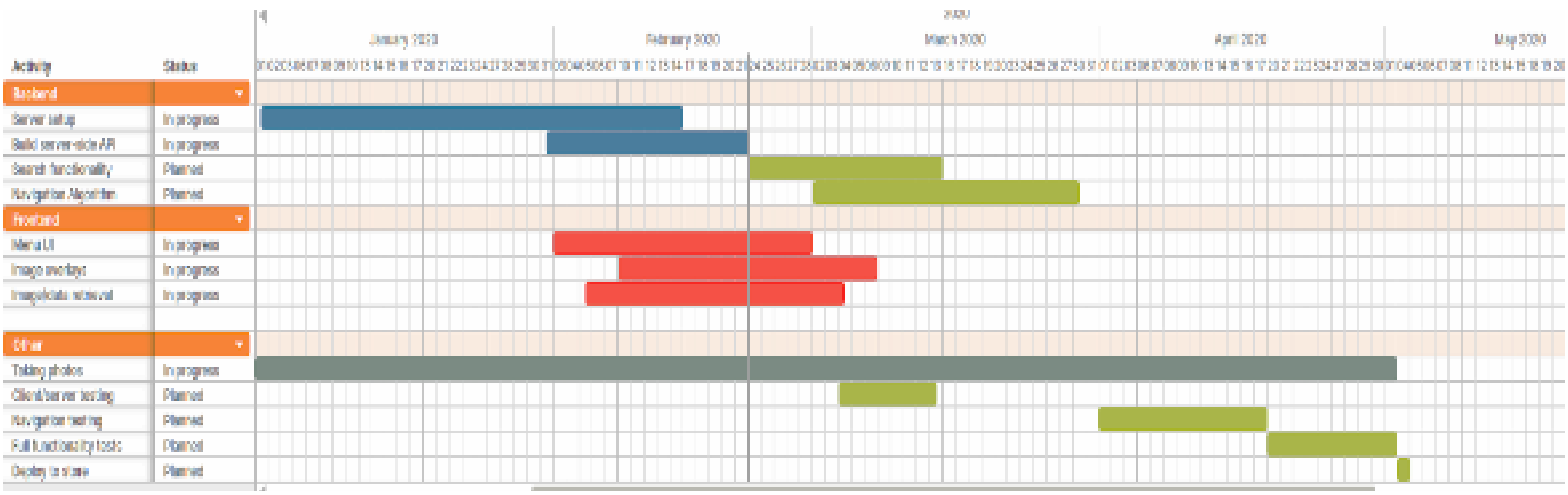
Ethical Issues

One ethical issue relating to our project has to do with the gathering of data. Our plan is to collect copious 3D/360° pictures to create a seamless virtual reality environment. The ethical complication revolves around the privacy of individuals located on campus at the time of capturing the photo. Since we're going to be photographically mapping a popular public location, there is a high likelihood that other people will end up in our pictures and therefore our 3D/360° environment. This is unethical for several reasons. First, the presence of photographic equipment in a public space may put people enjoying that space in a state of unease. It's not hard to imagine a situation where someone sees us taking photos and wonders where the photos are going to end up. Second, people simply might not want to have pictures taken of them. Perhaps they're having a bad hair day and might prefer that no photographic evidence of it existed. However, there are ways we can mitigate some of the ethical concerns. It's feasible to take pictures at times that are less busy. In addition, if we provide information to nearby people about who we are and what we're doing, and if we ask for consent of individuals in an area before taking pictures, we can greatly lessen public discomfort.

Another possible ethical issue might revolve around the future business of KU. For example, if our quality of camera is too low, the pictures of campus might end up looking unappealing. This could negatively impact future students' interest in coming to the University. Furthermore, it's possible that we accidentally photograph a cockroach, rat, or other worrying objects in one of the buildings that disgusts the users of the app. Possible ways to mitigate these issues include buying a better camera and manually double checking all 3D/360° photos for potentially disturbing details. In addition, we can clean up any trash in areas before we photograph them in order to make the campus look as appealing as possible.

Intellectual Property Issues

There are few IP issues to be addressed, one of them is copyrights. We are going to use some existing map instead drawing by our selves, the copyright of the map data could be an issue if we are using a commercial map. Some brand logos, or even the Jayhawk, may appear in our 3D/360° photos, we can hide those logos or statues using picture editing tools, but sometimes those objects are really helping to pinpoint the user’s location so those geo-symbol is important for users while using the app. There is a probability we need to get licences for those copyrights.



Change log:

- Changed project name to Walk Chalk
- Budget price updated
- Updated 2nd semester project milestones to better reflect our current plan
- Updated gantt chart to reflect current and expected progress