NavSB Draft Project
Team KIWI

team members / email

Ben Ye:   z_ye@ucsb.edu
Joon Lee:  jlee11@ucsb.edu
Dawit Aboye:  dawit@ucsb.edu
Kalyn Klimek:  kalynklimek@ucsb.edu
Alex Rasla:  alexrasla@ucsb.edu

Github:  https://github.com/jlee122/KIWI_CS48
Slack:  https://kiwics48.slack.com/messages/CHN0R5T6H/details/
Trello:  https://trello.com/b/nxdetmJq/cs48-project
Travis-CI:  https://travis-ci.org/jlee122/KIWI_CS48
What is project about?

The problem and how we want to solve it:

Students in college are overwhelmed and confused about a variety of things. One main barrier that students have to face is getting to know their new home - both the school campus and the city. We feel this is an opportunity to address some shortcomings presented by the simplicity of other UCSB campus resources. For example, many online UCSB maps are a simple digital view, without any notable interactive features that ease the navigation process or help discern/give information pertaining to certain buildings. In addition, an AR feature with pins and arrows would make navigating and routing easy, eliminating the difficulty in judging your direction or relativity from one location A to location B. We want to build these features to make a user friendly iOS app that helps guide students through their time in college and minimizes these barriers.

Why the problem is important:

This problem is important because as students at UCSB, we know how frustrating and difficult it can be to find the fastest route to get to class or to navigate through Isla Vista and campus. We want to take this opportunity to make students' lives easier by creating an app that addresses these annoying barriers.

How the problem is solved today:

There are many apps that currently exist including GoGaucho and the official UCSB mobile app that try to address and facilitate easy navigation for their users through interactive maps. We are expanding on this by adding more features and information to the interactive 2D map and including the AR feature that allows a more functional interactive user tool and experience for students.

Identify the outcome of the project:

Users including both students and visitors will utilize our app and its features to help navigate and facilitate their experience at UCSB.

How do you plan to articulate, design, and implement a solution?

We plan on implementing our solution by utilizing Swift language, and building with Xcode IDE. Swift is an object oriented language so we will be applying various OOP features, as well as incorporating C++/Objective-C subfeatures/sub-components. For our AR feature, we will be incorporating AR kit and APIs that work alongside our iOS app.
For 2D interactive map we will be using the “mapbox” Software Development Kit. It has a lot of features built in that will allow us to add more interesting features with more freedom and functionality. For our AR features, we will be using ARCL Framework that incorporates Cocoa’s ARKit and CoreLocation Frameworks.

Sprint Planning

Sprint 1 (April 15th - April 26th)

- Basic app that we can open with icon and at least a working 2D map
  - Where we are on the map
  - Swift has its own mapkit? Maybe we could use that also
  - Familiarize ourselves with google map API
  - Find out if we can use google map in our app
- Everyone on the team should get comfortable with using the XCode IDE and the Swift programming language - everyone make their own small app using Swift and XCode
- Research AR kit and how to utilize/analyze feed from iOS camera
- Assign specific features and tasks to people/groups of people
- Set up any possible tools and technologies we will need throughout the quarter
- Design and implement the user interface of our application, get it ready to demo
- Figure out/research routes to be used for AR
- Finish frontend / main screen for demo

Conclusion/Milestone: We finished every task in Sprint 1 except for finding routes to be used in AR and familiarizing ourselves with google maps API (highlighted in red above). Our project has changed a lot since the beginning in terms of which API's we will be using.

Sprint 2 (April 29th - May 10th)

- Use mapbox for 2d interactive map and have it ready for demo
  - Have interactive features working but add more later
  - Predefined pins as locations
● Implement basic AR features
  ○ Be able to incorporate pins and arrows with random pathing to make sure AR works -- working for sprint 2 demo!!
  ○ Make sure it can detect your coordinates + direction and assign random path from there
● Be able to move back and forth between view controllers
● Learn UI testing
  ○ Make tests for main screen
● Figure out how to use iOS location services

**Conclusion/Milestone:** We have completed almost all our Sprint 2 tasks, except for what is highlighted in green (in progress). We have now discovered and decided to use MapBox for our 2D interactive map and ARCL for our AR feature. We are all going to continue to dive into learning AR and getting it to work. For now, we were able to incorporate one pinpoint and we are working on more.

**Sprint 3 (May 13th - May 31st)**
● Finish up interactive map
  ○ Add search bar
  ○ Connect AR to 2D interactive map (add button)
● Add ‘Detailed View’ feature on building
  ○ Able to view exactly where rooms are located in buildings
● Finish implementing paths for AR
● Optimize user interface, make it more appealing to navigate and use
Requirements: Use Cases

1. Use Case: Navigate 2D map

**Actors:** Students, visitors

**Pre Condition:** User has opened the app and chose the option to view the 2D map

**Basic Path**
- App uses location services to figure out exactly where you are on the map and show it using some kind of indicator
- Top of the view will prompt for a desired destination when the destination is entered, a list of possible locations on campus appear. When the ‘Go’ button is pressed on one of them, the app will automatically direct the user to Google maps where the path will be shown

**Alternate Path**
- If no possible locations pop up when destination is entered, then that means that location does not exist on campus

**Post Condition**
- Path is shown on Google Maps
- User can return to app and continue use

[https://trello.com/c/oxaJt68j/2-use-case-1-navigate-map-2d](https://trello.com/c/oxaJt68j/2-use-case-1-navigate-map-2d)

2. Use Case: Interactive (Detailed) Building Information

**Actors:** Students, visitors

**Precondition:** App is open, the 2D map is chosen

**Basic Path**
- When 2D map is being viewed, the user can zoom in and click on individual building to have the option to display more detailed information
- One of the choices will be to show an interior floor plans with all the room numbers laid out. The user can choose the specific floor he wants to view

**Alternate Path**
- If no detailed information is available to certain small buildings, the user will not be able to click on them.

**Post Condition**
- The floor plan, or any other building information is displayed. The user can easily return back to the map, or home screen to use a different feature.

3. **Indicators showing path [AR]**

**Actors:** Students, visitors

**Precondition:** App is open, the AR feature is chosen, location enabled, camera access allowed

<table>
<thead>
<tr>
<th>Basic Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>● When AR feature is being viewed, the user can look around while the indicators are fixated to certain coordinates on screen</td>
</tr>
<tr>
<td>● The indicators take into account building proximity, and won't overlap on a building.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alternate Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>● If camera access is not allowed or location is disabled, the user will not be able to see indicators.</td>
</tr>
<tr>
<td>● If a path is being calculated/unable to be determined, the indicator will not show a path.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Post Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>● The indicators are displayed, allowing the user to see what direction to take.</td>
</tr>
</tbody>
</table>

[https://trello.com/c/yEPz6Mav/24-use-case-3-indicators-showing-path-ar](https://trello.com/c/yEPz6Mav/24-use-case-3-indicators-showing-path-ar)

4. **Pop Up pins in certain areas on campus [AR]**

**Actors:** Students, visitors

**Precondition:** App is open, the AR feature is chosen, location enabled, camera access allowed

<table>
<thead>
<tr>
<th>Basic Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>● When AR feature is being viewed, the user can look around while the pin is fixated to the coordinate destination on camera</td>
</tr>
<tr>
<td>● The indicators take into account building proximity, and won't overlap on a building.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alternate Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>● If camera access is not allowed or location is disabled, the user will not be able to see indicators.</td>
</tr>
<tr>
<td>● If a path is being calculated/unable to be determined, the indicator will not show a path.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Post Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>● The indicators are displayed, allowing the user to see what direction to take.</td>
</tr>
</tbody>
</table>


5. **Different types of ways to get to campus (ex: scenic, fastest) [2D]**

**Actors:** Students, visitors

**Precondition:** App is open, the 2D Map is chosen, location enabled

<table>
<thead>
<tr>
<th>Basic Path</th>
</tr>
</thead>
</table>


### 6. Different types of routes [AR]

**Actors:** Students, visitors  

**Precondition:** App is open, the AR feature is chosen, location enabled, camera access allowed

**Basic Path**
- The user can select from several options after inputting destination.  
- There will be provided scenic, fastest, and avoid bike traffic route.

**Alternate Path**
- If location disabled, user will not be able to select or see any routes.  
- If a certain route doesn't exist, then user will have fewer options. The route that doesn't exist will be opaque.

**Post Condition**
- The user is able to select a route from several options after entering destination location.  
- After selection, the AR feature will prompt the respective indicators and pins to proceed towards destination.


### 7. Main Screen for NavSB App

**Actors:** Students, visitors

**Pre Condition:** User has opened the app and chose the option to view the 2D map

**Basic Path**
- Multiple buttons to navigate functionality of the app  
- About Us: Team kiwi  
- Help & Assistance  
- Interactive Map: Opens 2D map with search bar and ability to find route to location input
8. AR Button on interactive map

Ae IRAS: Students, visitors

Precondition: App is open, the interactive map feature is chosen, location enabled, directions/route to entered location in progress

Basic Path

- When interactive map is in use and directions are given for certain location entered by the user, there will be an option to switch to AR map
- Once this button is pressed, directions are set up in AR map setting with indicators showing

Alternate Path

- If camera access is not allowed or location is disabled, the user will not be able to see directions in AR setting
- If a path is being calculated/unable to be determined, show an error message and option to re-enter address/location

Post Condition

- AR Map is opened and camera access is allowed
- The indicators are displayed, allowing the user to see what direction to take.

https://trello.com/c/XMGCB0fk/33-use-case-8-ar-button-on-interactive-map
Our system architecture consists of UI presentation layer, Logic process layer, Data Access Layer, and Information Layer. UI presentation layer has all the user interfaces, such as Interactive Map, Navigation in 2D/AR, Query address, about us, and user history. Logic process layer deals with all the logics from user interfaces like getting current location of the user, finding routes to the destination, etc. Data Access Layer deals with accessing the data from the database from Information layer. Our data will mainly be the coordinates and descriptions of the buildings on our campus and possibly the user information if we decide to implement accounts for our app.
Working with the interactive 2D map:

User has the option to choose 2D map option on the main screen of our navSB app. The user can search for their desired destination, and get a route on the 2D map. When the user presses a location pinpoint on the map, they can get the route on the 2D map or they can click the AR button to switch to the AR map. Additionally, when the user presses a location pinpoint, they can click the info button on that location to get a more interactive experience and more info on that location (ex: Phelps Hall, info button will show more information and a picture of Phelps Hall). This is represented by the show info message on the interactive lifeline. To go back to the 2D map from the interactive part, you click “Go Back” and to go back to the main screen from the 2D map, you click “Go Back.”

Mockup for Main screen and 2D interactive map below:
Working with the AR Feature:

User also has the option to choose AR map option on the main screen of our navSB app. The user can search for a destination in the search bar and get a route, or use a predefined location from the 2D map to get the route. Once the route is calculated, the AR feature will show arrows in the direction of the specified destination. AR will also show indicators such as pinpoints when passing known landmarks on the UCSB campus. This is shown on the interactive lifeline as well.

Above is our initial AR node/indicator
Appendix:

Location Services:
- Use of Core Location to deal with real-world locations, coordinates, and current location

Mapbox Library:
- Use of Mapbox third party library to implement the interactive 2D map
  - Implements navigational and directional features
  - Implements 2D map interface and significant locations

AR Feature:
- Use of ARKit library to implement basic AR features
- Use of ARCL library to implement AR features related to location