Sprint 1 Design Document

CSCI 492 - AR Tour Guide
Ethan Blight, Cole Goodnight, Nick Vyvyan, Danny Inga

Administrative Web Panel / Web Scraper | Ethan Blight & Cole Goodnight

Context/Overview:
We are constructing the basic implementation of an administrative web panel for the app, which will be used to add buildings, and later landmarks.

Primary functionality includes:

- Scraping building web pages for information on said building.
- Drawing closed shapes on a map to signify the building surface area.
    - GPS coordinates of interest (e.g. centerpoint) are derived from the outline.
- Data is returned in the admin panel, which allows the cleaning up of data, and admins to make manual modifications to the scraped data before submitting.
- Once submitted, the building data is pushed to the database.
- This approach can also be used to edit/update existing buildings in the app.

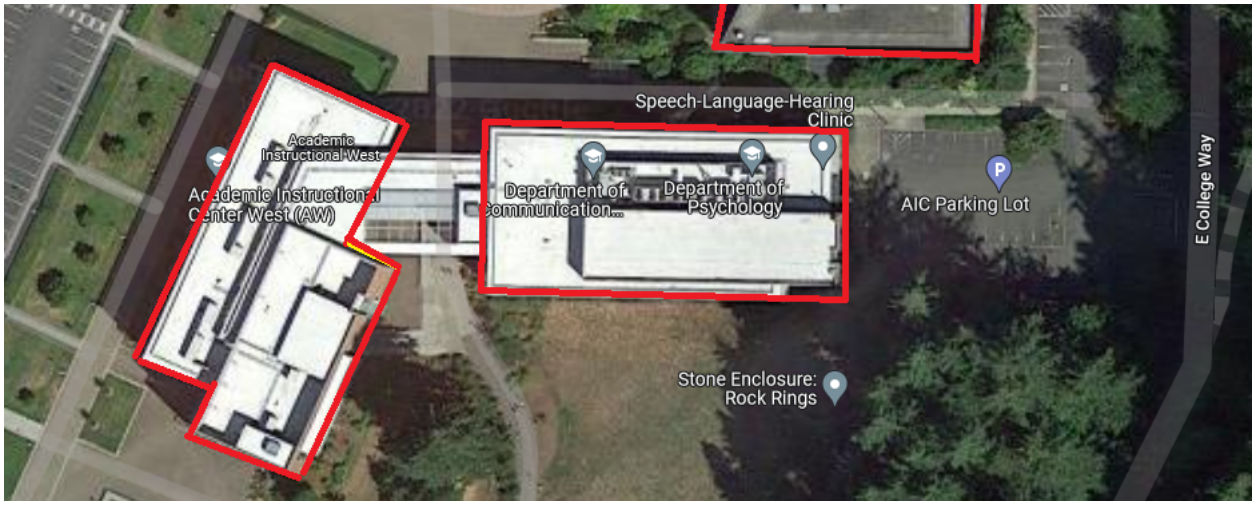
Stretch goals/future functionality:

- Uploading audio/additional data on given buildings for accessibility purposes and interaction.

Libraries/APIs:
The primary website design will be built using the React framework, and will use the Google Maps JavaScript API to draw on an embedded map and outline the given campus building. For testing purposes, a local database will be set up, but this will eventually move to a cloud-based solution or university-hosted server.

Web Scraping:
As many of the university building pages share similar elements, HTML pages will be downloaded and a JavaScript scraper will parse information directly from the page. The object produced will then auto-populate a form on the React site, so that manual modifications can be made to the scraped data by the admin.

Map Drawing:



The Google Maps JavaScript API should allow the user to draw onto the map in a similar fashion to the above mockup. From there, points can be derived that describe geometry, a centerpoint for the building can be calculated, and all of these points can be stored in the database.

Existing Buildings:
Buildings will be listed with the ability to create, delete, and/or update information for each respective building.

GPS / Compass Data | Nick Vyvyan & Danny Inga

Context/Overview:
Using the FusedLocationProvider API in order to get the phone's current location, and the phone sensor manager to get the compass heading, this data will be used to determine which buildings are nearby and currently being looked at.

Primary Functionality:
To get accurate location and orientation data from the phone's built-in sensors. Simple display of current location/sensor data to verify functionality.

Libraries/APIs:

- **FusedLocationProvider**
  - Uses a combination of WiFi, cell towers, and GPS location data to get the last known location of a device. This combination can be configured to be more precise, (at the cost of more power usage), or less precise, but more power efficient.
- **SensorEventListener Interface and SensorManager**
  - Uses device hardware sensors to provide updates of current readings, which is used to get the current heading for the compass.