

## Transcript for Project proposal

2. The basic idea for our idea can be summarized in one sentence: A simple route generator for runners and cyclists to use so they have variety in their workouts outside. If you want to get outside and do more cardio or you want to see a neighborhood more then this is the app for you. Having a random route will help keep things interesting while also giving motivation to run or cycle.

3. Because the Upper East Side is rather simple to map with a nice and even grid structure and uniform block size. Due to just the sheer amount of streets we'll have though, we're going to reduce the actual size of the data we store for the purpose of the app. We first want to understand that the concept can work and then once we establish that we can add more and tweak the algorithm to maintain the level of efficiency we had with the small sample size. Hopefully, at least. We're doing this on the android platform because Gary has some experience with android app development and some experience with the server we would connect to using android so it is the best choice for us.

4. Now, onto the features for the app: we want users to be able to search using only three things: Distance, terrain type, and starting location. The first is so we know how long to generate the tracks, the second is so we know what types of streets to include in the generation and the third is just so the user has a choice of whether to start their run from home or from some other point. Originally we weren't going to have an option to make a profile seeing as there's only random routes and no real reason to login for that. Until we thought of putting in ratings and then we needed to have some way of saving them. Ratings will allow users to dislike certain parts of a route if they don't like it or like it if they do. They'll also have a history option that saves routes they've picked before incase they'd like to use them again. The last user option is to pick a starting location: users can start from their current location (home) or generate any route no matter the starting location (as long as it's within the neighborhood).

5. Now onto the data we'll be using in our database. We'll be using Google's elevation API to get the elevation of the intersections of streets. We'll get distance data from Google maps and we'll also get latitude and longitudes from Google maps (that way we can avoid scanning the middle of a building).

6. This is a sample map of the elevations around Hunter College. As you can see from walking around Hunter North most of the streets are flat with relatively no incline but notice the slight incline going from Madison to Lexington. The map may be outdated as it includes Radioshack but that's not a problem for us.

7. Because it is an android application, the language is obviously android. We'll use JSON to communicate with server. JSON is more structural information in the document, thus it is easier for us to read and write data by ourselves, and it is more readable. And JSON is easy to handle with Android, which already has many fixed function dealing with JSON. The Server we are going to use is OpenWS REST API. I used this server before in another class. It is a private server and totally free. In addition, the server is

built aiming for developers to deal with JSON object which it is what we need. And no server-side knowledge requirement, supporting POST,GET,DELETE,PUT methods.

8. This is the data format we'll use. Because we can't use data from Google Map directly and we have to store data by hand, it is important to think about what format we should use which is easier for us to implement them. Thus for the street data, We select each corner as each piece of data including ID , latitude, longitude, elevation. It is hard to store each corner name so we want to just create some number as ID for each piece of data. The elevation is for us to get the terrain type of streets. Because we should know which corner connects to which corner, we will create a graph, too. It is an undirected graph. It includes ID1 ,ID2, distance. The two ID are two connected corner, and distance is just the length between two corner. According to these two type of data, it is easier for us to implement algorithm on them.

9. Now onto the project structure we'll build. First, we will do a log in screen to store user information, and we can implement features such as history, preference through knowing which user it is from this step. Second, it is a main page of the project which will include some buttons for different features. Once user click one of them, user will go to the screen corresponding to the feature. Third, it is the screen for users to type requirement for generating routes. Finally, we'll see a displaying screen which it is the result we want.

10. For the distribution, Jerry will built the Android application interface, and Gary will deal with data and algorithm. Because Gary worked on JSON and Android before, it is suitable for him to do data part. And if each of us meets any problem in any part, we will work together to figure it out.