Neighborhood Informant - Sprint #3 Progress Report

Group 2: Aiwan Hazari, Deven Patel, Dhrumil Patel, and Jay Patel

---

Project Summary

Neighborhood Informant is a desktop application that provides ease to Chicagoland residents by providing a single application of various Chicagoland data. This application will gather real and accurate data straight from the City of Chicago data portal website. There will be an easy-to-use user interface that will allow a user to pick a location and neighborhood and receive relevant feedback about it, such as crime, schools, average income per capita, homes for sale, tax increment financing, etc. Neighborhood Informant will be a one-stop for a multitude of accurate Chicagoland information.

Sprint 3 Expected Goals

What we initially expected to accomplish for Sprint 3 was to clean up the database of irrelevant information, and to make a connection between the database and program's graphical user interface. The program would also ideally be able to plot some data points on the map by the end of this sprint. Adhering to Extreme Programming protocols of continuous testing, we also expect to test the program for bugs and possibly fix any occurrences within this sprint. One last goal was to initiate a backup MSSQL database file, in case an internet connection error occurs while a user is using Neighborhood Informant.

Sprint 3 Finished Tasks

Since the last release we settled on what data to use. We decided that we want to use all the data points that are between Harrison, Racine, State St, and Cermak Rd (Figure 1). The simple reason for choosing these intersections was because these are the neighborhoods that encompass UIC and many of its students. Furthermore, by limiting the data to a small subsection of Chicago (as suggested by Tanima during the release), we would be able to fully incorporate many features, without the overhead of worrying about speed because of having too much data.

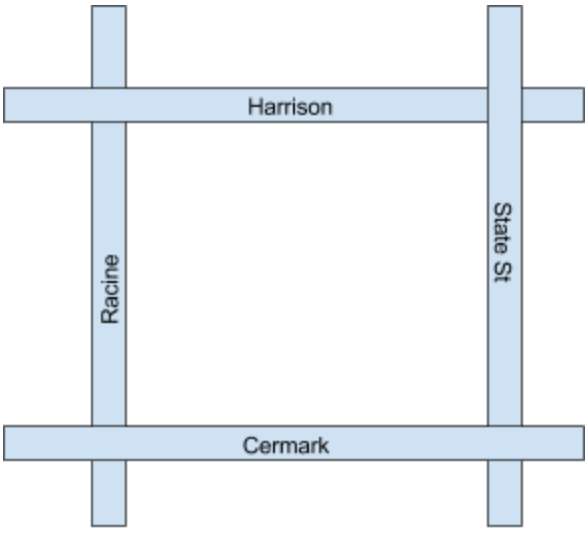

Figure 1

Currently data is successfully being transferred from our machine to Firebase online. We have also started

researching how to instantiate the backup MSSQL database file. Currently, the application's main codebase (user interface, data operations, etc) is in Java. However, since the Google Maps API was not readily available for Java, we are also using HTML, Javascript, and JXBrowser to display the Google Maps data.

Additionally, we used in-class techniques and performed "code smell tests" on our code. There was a lot of code that was being used multiple times. So to make our code more efficient, we refactored our entire code base to share this repeated code. We than spent rest of the time testing the new code that we created after refactoring.

Next Steps

The next steps to take for the fourth sprint are to further enhance the database and program integration. This involves only sending data (to the database) that we are going to display on the application's current screen, rather than always displaying it all at once. We also want to map all the coordinates to their respective street addresses on the map. Another feature we will incorporate is to create a way where a developer can just feed a file with data of the entire city, and the program would only keep the data that falls into our parameters. Then, after successfully uploading the file, the data will load into the Firebase database for later use. We will also continue testing our code and further refractor any "code smells" within our code.