

Cole Ding, Justin Farnsworth, Quinn Wasko, Jake Dingley, Anthony Kim
2/27/19

Project Specifications - Beating Houses to Death

1. Problem Statement.

People of lower income are having an increasingly difficult time finding affordable housing inside cities.

2. Objective of the module.

A way for people of lower income to find affordable housing based on specified constraints including income, location and desired standard of living.

3. Description of the desired end product, and the part you will develop for this class.

The overall goal for the end product is to create a comprehensive database that allows users to quickly access housing information inside the five boroughs of New York City. This database will provide queries that will allow users to enter general information about themselves and return affordable housing options that will meet their specifications. For example, a user may wish to find an apartment for rent with a monthly expense of \$2,000 to \$3,000 dollars in a well regarded school district with low per capita crime rate. The queries will return all the housing options in the NYC area that meet these requirements allowing the user to begin their affordable housing search. Additionally, it is our goal for the database to be able to handle various different user criteria including desired rent expense, desired borough, crime rate, and school district rating. In this class we will develop the database with all of its tables and relations in addition to building in queries using SQL.

4. Description of the importance and need for the module, and how it addresses the problem.

Low income housing and the ability to find that kind of housing in certain areas can be very difficult using normal resources. This specifically allows people to find this data in an easy and convenient manner by just defining the constraints that they specifically have. Though this is not necessarily limited to those of low income to benefit from, it does make there search for housing significantly less painful and time consuming.

5. Plan for how you will research the problem domain and obtain the data needed.

We are planning to find datasets that have to do with house prices in NYC by looking at houses sold or houses available for sale. We are also looking for datasets that tell us the value of land such as points of interest and sales of properties in general.

6. A high-level description of the data source(s) and links if applicable.

Historical data of property sales in NYC (2016-2017):

<https://www.kaggle.com/new-york-city/nyc-property-sales>

Bureau of Labor Statistics: https://www.bls.gov/oes/2017/may/oes_ny.htm#otherlinks

Zillow: <https://www.zillow.com/new-york-ny/>

7. Other similar systems / approaches that exist, and how your module is different or will add to the existing system.

Zillow does a good job describing the house and how the property's price will scale over time based on the neighborhood. We want to differentiate our project by focusing more on the income level of users and how they can best find affordable housing options. In the future, we would like to add features that give a more comprehensive view of living in a city and can calculate the feasibility of living in the area based on the user income. For example, the system could calculate the average total expense of a geographic area given the calculated rent, food expense, transit expense, and other general information. Additionally, we would like to add what kind of industries exist in the area for the users to find desired work in the area. The Bureau of Labor also provides data regarding salaries, which can give us an idea on how much income may be needed to afford certain houses and also what a person with an arbitrary job can afford.

8. Possible other applications of the system (how it could be modified and repurposed).

It could be applied to other cities or even to a larger area such as counties and states. It also might be possible to add a more holistic sense to the database where it incorporates average cost for living (transit, food, utilities, school, etc.) and based off of income determine whether outside of the cost of housing is it reasonable to live in this location.

9. Performance – specify how and to what extent you will address this.

In general, we want to improve performance by keeping separated tables for non-unique attributes of house entities. We are aiming to limit the number of properties we need to look at through query search by separating the tables.

10. Security – specify how and to what extent you will provide security features.

The main security threat to our system would be from injection attacks. User inputs will be properly sanitized before being fed to our DBMS to prevent unauthorized access and modification to our system. The system will have minimal personal info attached to it, so that even if data were to be leaked it would not be a significant loss. Logs may be used to verify if data was accessed or modified by a malicious user, and backups may be required to restore the data to an accurate state if necessary.

11. Backup and recovery – specify how and to what extent you will implement this.

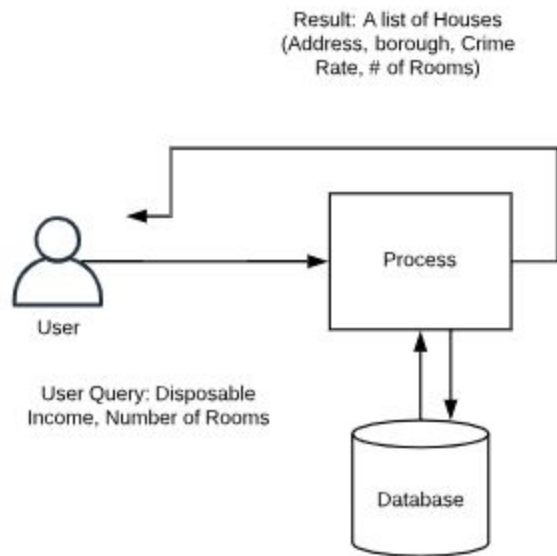
Possible implementation of multiple server redundancy to insure if one server with the data malfunctions, then there is another server with the data stored on it somewhere, and just general use of redundancy within the database to ensure that data doesn't only have one copy.

12. Technologies and database concepts the team will need to learn, and a plan for learning these.

- PostgreSQL
- SQL
- Normalization
- Database structure
- Data Integrity and Consistency

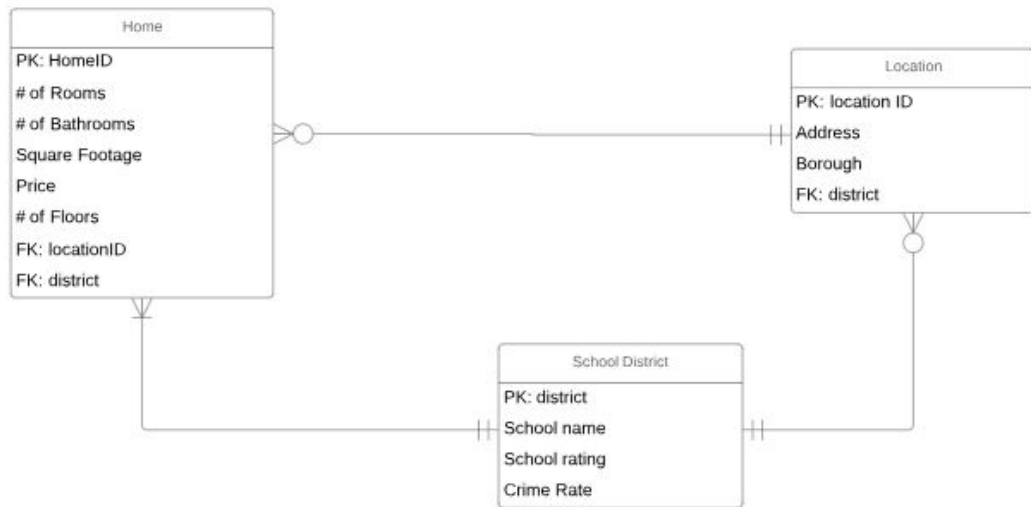
The plan is for us to either together or separately use different resources such as textbooks or the internet to gather information about these topics.

13. A diagrammatic representation of the system boundary, i.e. specify what data you will model and which queries you will implement.



database ER

Anthony Dohyung Kim | February 27, 2019



Private Repository: <https://github.com/kimd24/Beating-Houses-toDeath>