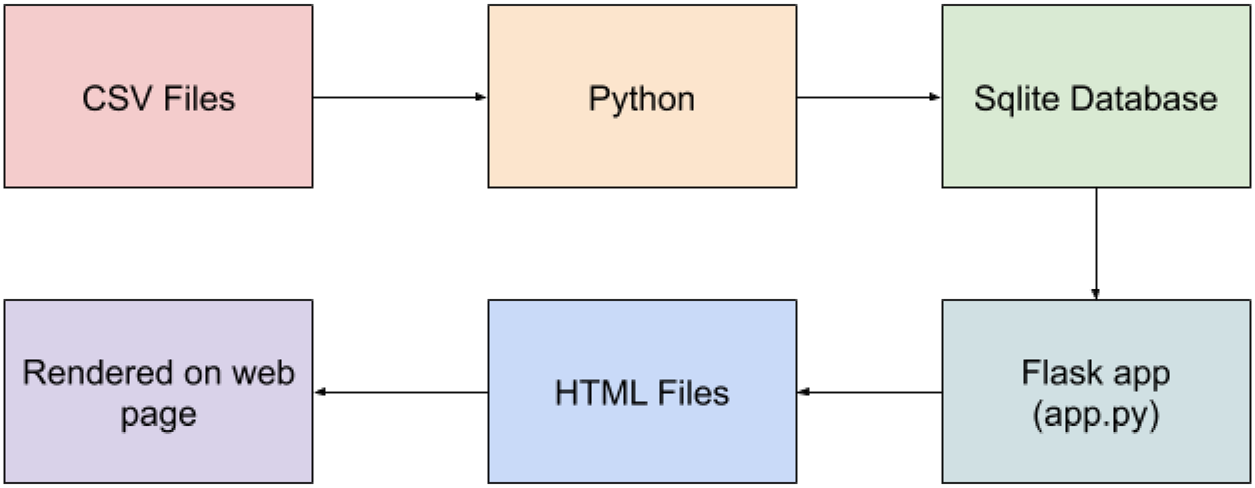
Team Lach On (Maddy Andersen, Dean Carey, Kelly Huang, Erin Lee)
SoftDev — Rona Ed.
P0 — Quick! Quick! DESIGN DOCUMENT
Due 2020-12-18

Program Components and Role of Each

- **Sqlite Database**: holds our data and information so we can access it to display on the site
- **CSV Files**: holds information for our databases to be inserted into the sqlite tables
- **Flask**: the framework for our web-based app, connecting and managing the routes and functions
- **Python**: converts our CSV to the sqlite, works with Flask, directs our functions and renders templates
- **HTML/CSS**: foundation and styling for our front-end pages

COMPONENT MAP



Databases

- NYC Stats (table)
  - Headings
    - Population (integer)
    - pic_URL (text)
    - Map (text)
    - Description (long text)
    - Fun facts (text)
- Boroughs (table)
  - Headings
    - Name (text)
    - pic_URL (text)
    - Population (integer)
    - Area (integer)

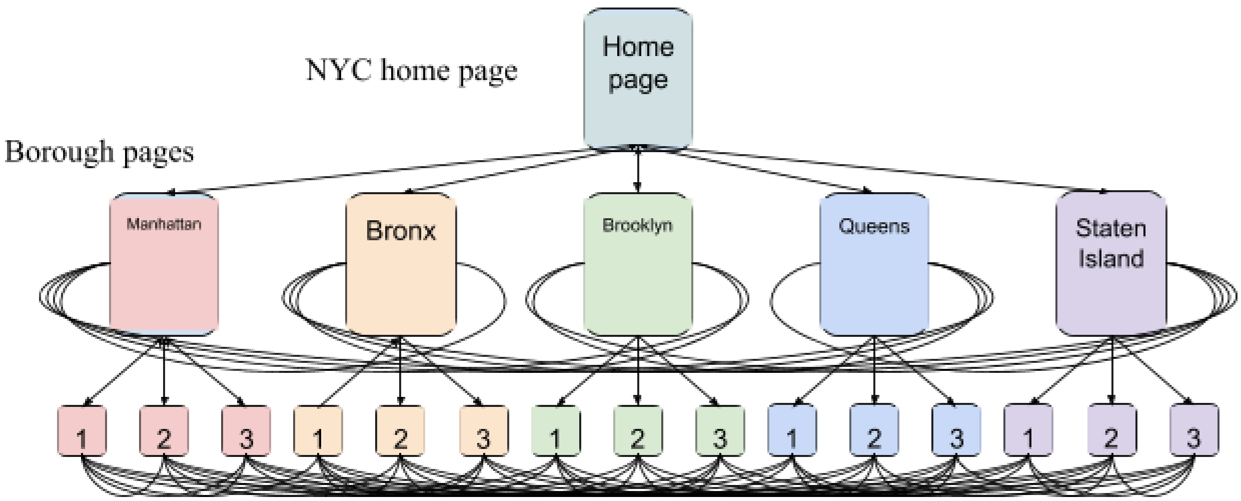
        - Neighborhoods (text)
        - Subway lines (text)
        - Quote from student (text)
- Restaurants (~7 per borough) (table)
    - Headings
        - Name (text)
        - Borough (text)
        - Type (text)
        - Price range (text) [in $ signs, $ - $$$]
        - URL (text)
- Parks (~5 per borough) (table)
    - Headings
        - Name (text)
        - Borough (text)
        - Area (integer)
        - One sentence summary (text)
        - Cool fact (text)
- Landmarks (~5 per borough) (table)
    - Name (text)
    - Borough (text)
    - Description (text)
    - Price range (text) [in $ signs, free - $$$]
    - URL (text)

Task Breakdown

- Create CSV files (5) with information added manually (*everyone*)
    - All group members work on files and take on a different CSV file/type of information
    - NYC stats/boroughs (Dean)
    - Restaurants (Kelly)
    - Parks (Maddy)
    - Landmarks (Erin)
- Set up sqlite database, tables with Python (refer to k16) (*Kelly, Dean*)
    - Only complete once CSV files are finished
    - Connect the database to the python
        - Figure out how to access/print tables onto the Flask app (can only view in terminal)
- Set up Flask app (*Maddy*)
    - Make app.py
    - set up folder structure
    - set up routing
- Create templates/HTML files (*Erin, Maddy*)
    - Connect to flask routing
    - POST/GET for retrieving information (linking pages)
    - Rendering different HTML pages, possibly adding CSS (if everything works)

    - Home page
    - Borough page (1 template, 5 variations)
    - Restaurants page (1 template, 5 variations)
    - Parks page (1 template, 5 variations)
    - Landmarks page (1 template, 5 variations)
- If we have time (*TBD*)
    - Adding users (cookies, sessions), creating a database of passwords and usernames

SITE MAP



Borough sub-pages (as of now):

1. Restaurants/food
2. Parks
3. Landmarks

Pathways:

1. Homepage connects to all borough pages
2. Each borough page connects to their respective subpages (1, 2, 3), as well as the other borough pages and also back to the homepage
3. Each subpage connects back to the borough page and the other subpages within the borough, as well as to the same subpage type of other boroughs

Ex. Manhattan 1 connects to Bronx 1, Brooklyn 1, Queens 1, SI 1, Manhattan 2 & 3, and Manhattan page