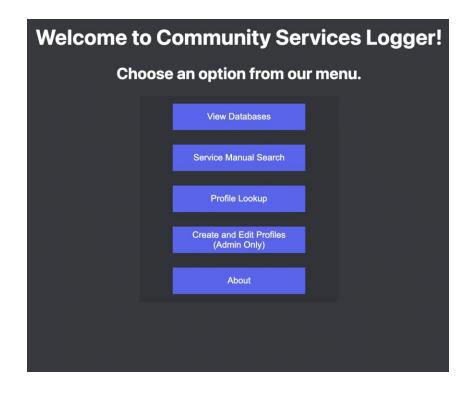
# Community Services App

Alejandro Sanchez

Aditi Kisara



## System Description: Main Menu

Our project utilizes a database that is filled with resources and is catered for people in underserved communities. The following are the services our app offers: Food Pantries, Clothing Banks, Soup Kitchens, Medical Assistance Services, Daycare Services, and Homeless Shelters

Users can find services based on zip code, city and county.

Each service has the following information stored in our database:

- Service name
- Address
- State
- City
- County
- Zip code
- Phone Number
- Days/Times available
- Website url

Users can also look up the **managers** working at each service and the **customers** that are subscribed to that service.

'Create and Edit Profiles (Admin Only)' is an **admin only** option, which means the admin will be asked to login with admin credentials.

#### **Welcome Admin**

**Edit User Profiles** 

**Edit Manager Profiles** 

Create User Profile

Create Manager Profile

Delete User Profile

Delete Manager Profile

Logout

#### System Description: Admin

- Admins can search up any profile in the managers and subscribers database and make changes.
- The role of admin is simply to authorize the edit, create and delete function, which gives them the opportunity to modify the managers and subscribers table.
- Currently, there are only 3 people who can make changes, including us, who can login and use the admin features.
- The reason that 'Delete User Profile' and 'Delete Manager Profile' are red is because if any user or manager is deleted, they are permanently deleted and their information cannot be retrieved after being deleted.

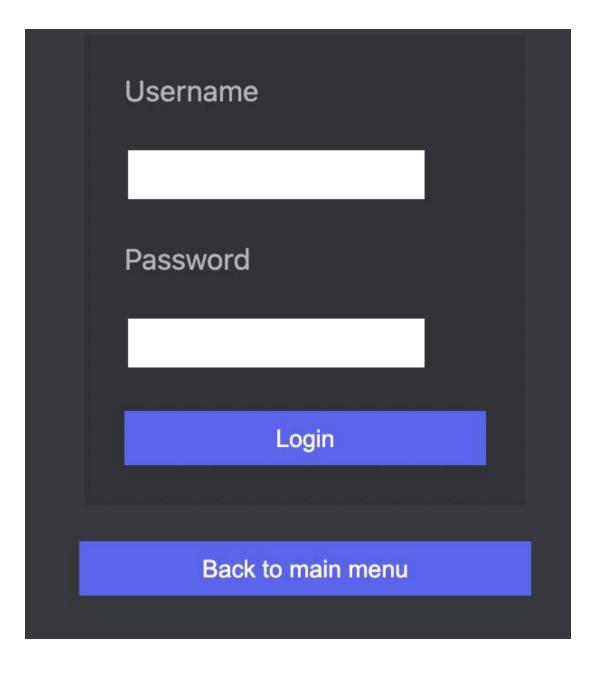
#### Login Credentials for Professor and TA

- For demonstration purposes, our team developed a login username and password for the professor and TA so that they may utilize the admin feature.
- LOGIN:

•Username: Teacher

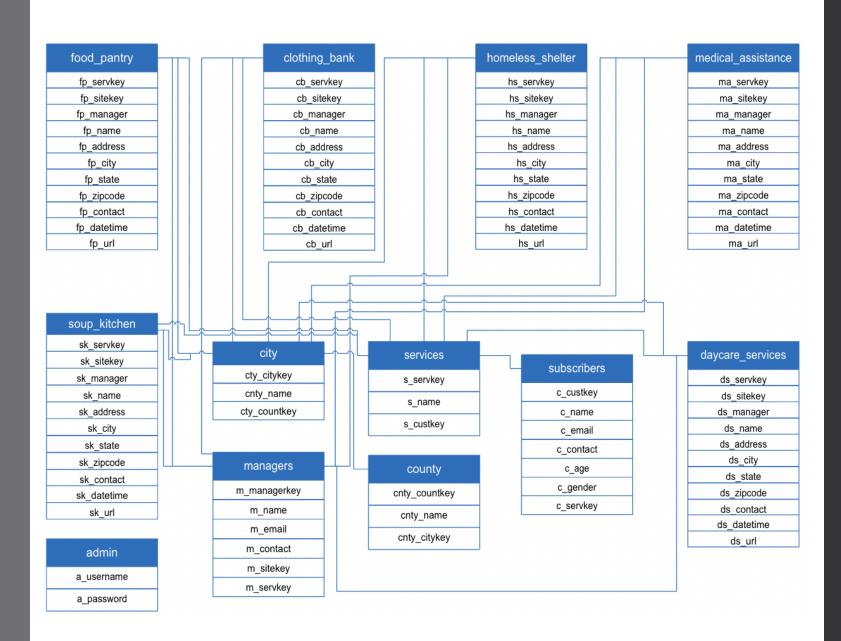
•Password: CSE111Database2021

- •These login credentials are not meant to be shared with anyone outside of the teaching staff.
- Visit webapp at
  - •https://nightraven.pythonanywhere.com/



#### View Main Menu << extends >> << extends >> << extends >> receives service info--receives profile data-User Search for View all Search for services databases profiles manually mannually << extends >> << extends >> << extends >> << extends >> Admin View Main Menu << extends >> -<<red to>>-Edit, Create, and Delete Records Edit or Delete << extends >> **Existing Record** <<extends>> Edit, Create, and <-<extends>>-▶ Delete Menu <<extends>> Create a Profile <<redirected to>>-

## Use-Case Diagram



## Relational Schema

#### cb\_address fp\_name fp\_address cb\_name hs\_name hs\_address cb\_sitekey hs\_sitekey fp\_sitekey fp\_url fp\_zipcode cb\_url cb\_zipcode hs\_url hs\_zipcode hs\_state cb\_state fp\_state fp\_contact cb contact hs contact hs\_city cb\_city clothing\_bank food pantry homeless\_shelter fp\_city cb\_datetime hs\_datetime fp\_datetime < is in fp\_servekey has hs\_manager fp\_manager cb\_servekey cb\_manager hs\_servekey < is in is in services include services include services include has has s\_custkey c\_custkey cty\_countkey c\_name a username s\_servkey share a\_password c\_servkey 444 city admin cty\_name cty\_citykey s\_name subscribers c\_email c\_gender contains c\_contact m\_managerkey c\_age cnty\_countkey c\_name cnty\_citykey managers county cnty\_name m\_servkey c email m\_sitekey c contact has is in has services include services include services include has is in < is in ma\_servekey ma servekey ma\_manager ma\_city ma\_sitekey ma\_manager ma\_sitekey ma\_manager ma\_servekey ma\_sitekey ma zipcode ma\_zipcode ma\_state ma\_state ma\_state ma\_zipcode soup\_kitchen ma\_contact medical assistance ma\_contact daycare\_services ma\_name ma\_name ma\_name ma\_contact ma\_datetime ma\_datetime ma city ma\_url ma url ma address ma datetime ma\_city ma\_url ma\_address ma\_address

# ER-Diagram







#### Implementation Details

- IDE: Visual Studio Code.
  - Useful for collaborative production of code and scripts.
  - 'Live Share' feature was useful because it lets multiple users modify the same document in real time on separate systems.
- Database Handler: SQLite.
  - Databases are created, modified, and managed with SQLite commands and queries.
- Executed on a webpage using Python, HTML, CSS.
- Python handles queries using the sqlite3 library. Flask handles the presentation of data.
  - Python was useful because data can be retrieved in many ways needed by the user using SQLite commands.
  - Flask makes data presentable possible. The 'render\_template' library gave us the flexibility to use HTML and CSS with Python, meaning we could create tables, menus, and produce meaningful redirects based on user input.
- Github repository:
  - https://github.com/alejandrohsanchez/Community-Shelter-and-Food-Pantry-Data-Logger

# Demo