H1B Visa Analysis

By: Bug Brawlers

BUG BRAWLERS
(Team composition)

- Akhilesh Goel: Non - Tech
- Sparsh Aggarwal: Non - Tech
- Vaibhav Gupta: Tech
- Divik Satija: Tech
- Kashyap Raina: Tech

Why We Choose Problem 1??

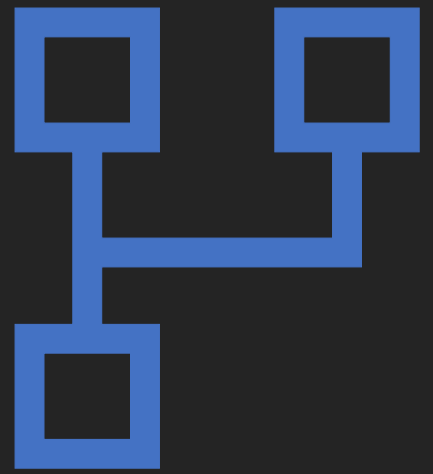
- Bug Brawlers comprises of 3 Tech and 2 Non-Tech members. Our team's particular proficiency resides in Data Analysis, with a focus on SQL, led by Vaibhav Gupta.
- Divik Satija spearheads the Backend operations, specializing in Django, an area that significantly intersects with Problem-1.
- Complementing our Data Analysis results, Kashyap Raina ensures the presence of an essential Frontend, while Akhilesh Goel and Sparsh Aggarwal takes charge of Version Control and Management.
- As for the PPT, it falls under our collective responsibility. Therefore, our team was ideally suited for Problem-1.

Problem Statement

- Download the data files for the years 2016 to 2019 in a format like CSV.
- Write Python code to read the downloaded CSV files and insert the data into the PostgreSQL database.
- Define models in Django to represent the data in your PostgreSQL database.
- Create API endpoints for certain functionalities.
- Write Python code within your Django app to perform the required data analyses

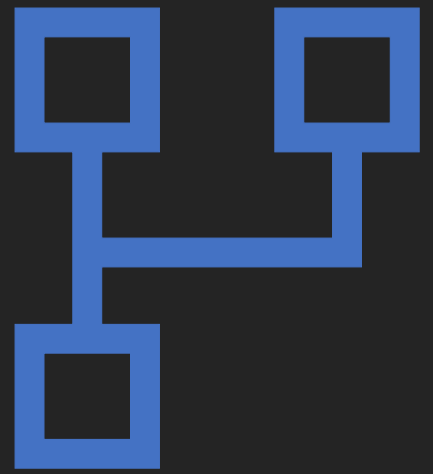
Ideology / Brainstorming

- Initially, we thought of pushing the files of type .csv in the PostgreSQL database using Python.
- But then we starting using django.
- We first psycopy2 library to connect to the server on pgadmin4.
- After that we developed classes in model which created tables.
- Now to push the data on the server, we defined a function in views.py file and we defined a function that will handle a specific url route.

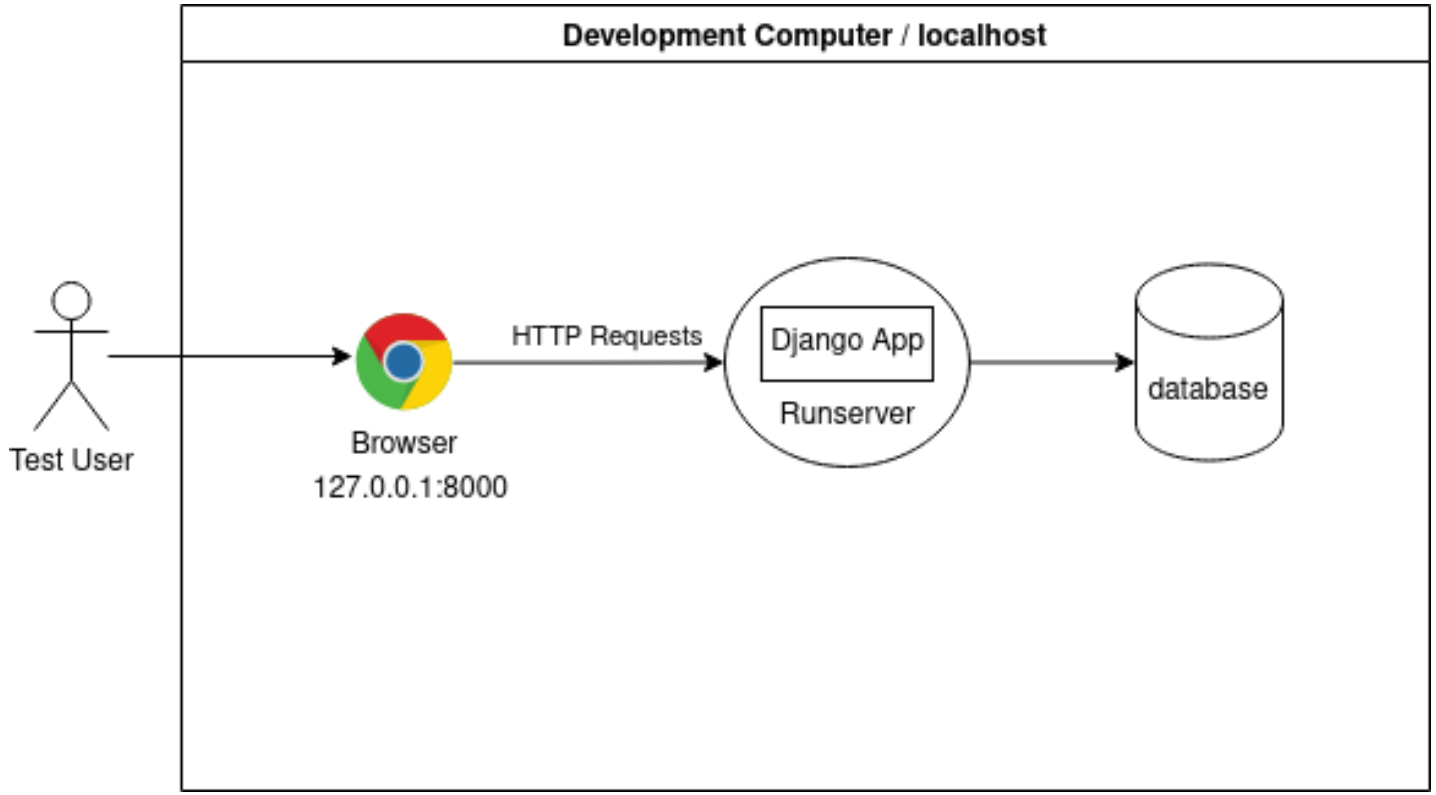


Ideology / Brainstorming

- Then we imported csv library instead of pandas to read the csv file.
- We defined a function which imported data from csv to our django model by iterating over the file.
- Now we created 5 APIs for the endpoints of the given functionalities.
- Now we analysed the data and made SQL queries for the functionalities.
- After this, we built the website to show the results of our model but due to lack of time, we were unable to connect the front-end with the back-end.



Setup


Development Computer / localhost
Test User
Browser
127.0.0.1:8000
HTTP Requests
Django App
Runserver
database

Data Modifications

- Columns which are more than 80 percent empty were removed.
- The date format was changed.
- As we have seen the salary per unit has multiple time ranges using which we were able to create the total annual salary.
- The functionalities were created by considering the unique IDs and total annual salary.


Thank You