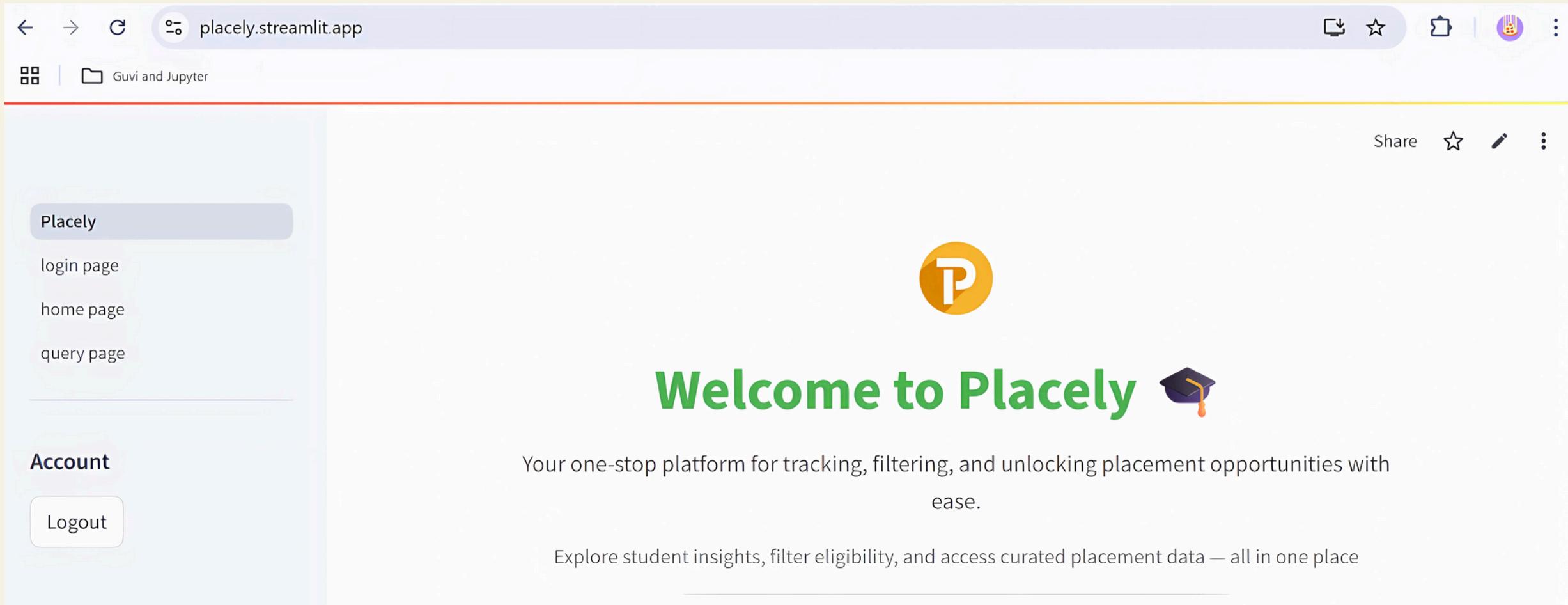


# About

## Placely - a placement eligibility app



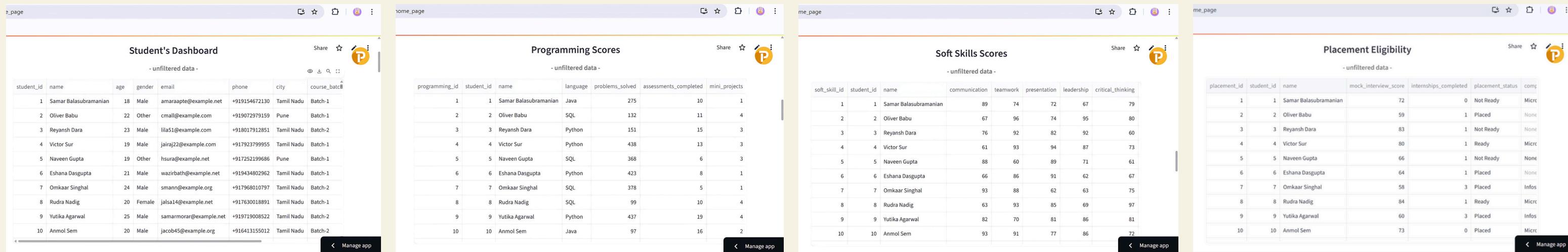
The screenshot shows a web browser window with the URL `placely.streamlit.app` in the address bar. The page title is "Welcome to Placely". On the left, there's a sidebar with a "Placely" tab and links for "login page", "home page", and "query page". Below that is an "Account" section with a "Logout" button. The main content area features the "Placely" logo (a yellow circle with a white 'P') and the text "Welcome to Placely" followed by a graduation cap emoji. It describes the app as "Your one-stop platform for tracking, filtering, and unlocking placement opportunities with ease." A call-to-action button says "Explore student insights, filter eligibility, and access curated placement data — all in one place". The top of the browser window has standard navigation and sharing icons.



Placely uses a thoughtful color scheme: yellow to highlight bright, eligible students; green to symbolize growth and successful placements; and light blue to visually separate filtered data from unfiltered data, keeping the interface clean and intuitive.

Placely is a web app built with Jupyter Notebook ( Python language and libraries ) MySQL, VS Code, Railway, Github and Streamlit that makes student placement data easy to explore. It helps users filter records, run queries, and download insights through a simple, interactive interface. Designed for clarity and speed, Placely streamlines the process of checking placement eligibility and analyzing student performance.

# Data Generation

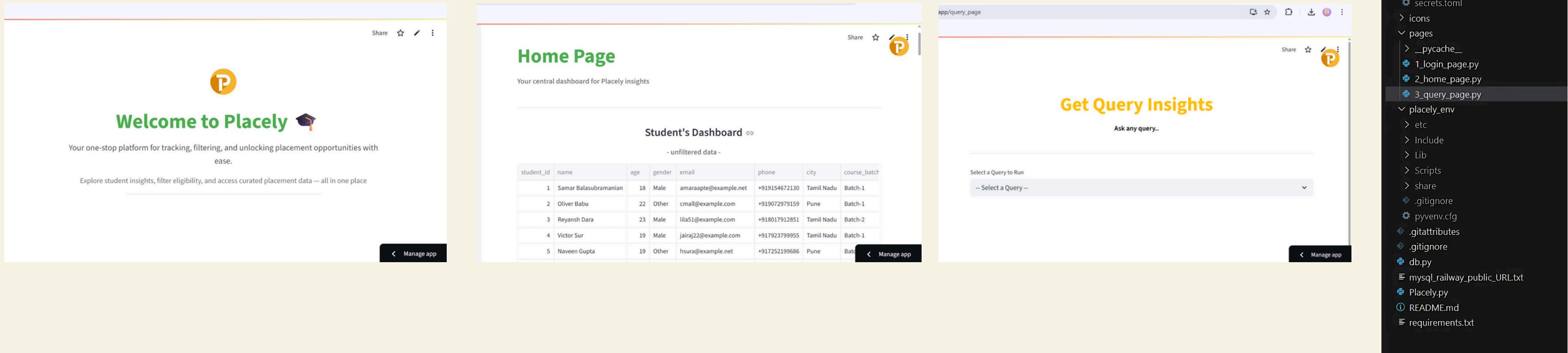


The screenshots show four pages of a web application:

- Student's Dashboard:** Displays a table of student records with columns: student\_id, name, age, gender, email, phone, city, course\_batch. The data is unfiltered.
- Programming Scores:** Displays a table of programming scores with columns: programming\_id, student\_id, name, language, problems\_solved, assessments\_completed, mini\_projects. The data is unfiltered.
- Soft Skills Scores:** Displays a table of soft skills scores with columns: soft\_skill\_id, student\_id, name, communication, teamwork, presentation, leadership, critical\_thinking. The data is unfiltered.
- Placement Eligibility:** Displays a table of placement eligibility with columns: placement\_id, student\_id, name, mock\_interview\_score, internships\_completed, placement\_status, comp. The data is unfiltered.

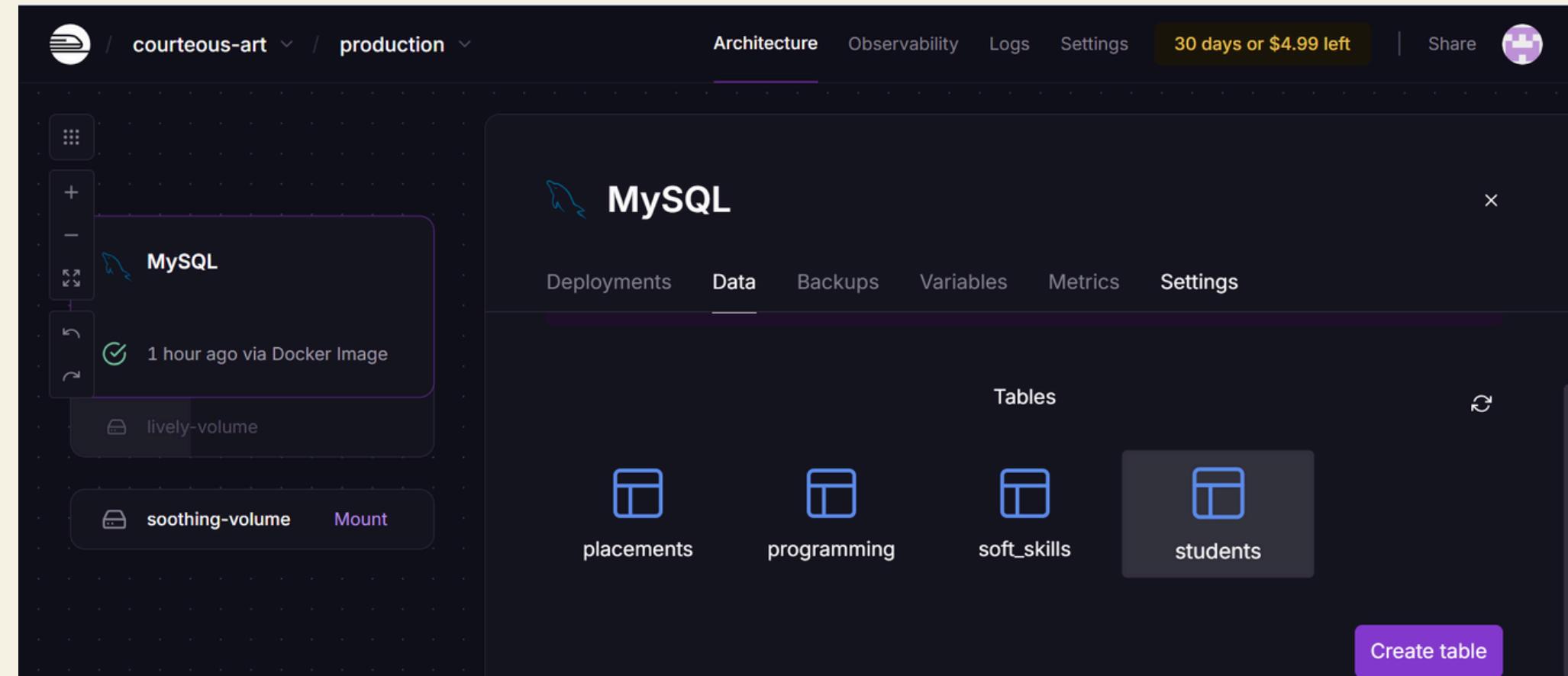
Synthetic placement data was generated in Jupyter Notebook using Python. Libraries such as Faker, pandas, and mysql.connector were used to build custom Python classes corresponding to the Students, Programming, SoftSkills, and Placements tables. Within these classes, a generate\_data method was implemented to create approximately 2,000 realistic fake records. After generation, the data was organized into pandas dataframes and inserted into the respective MySQL tables via mysql.connector. This approach ensured that the Placely app had clean, structured, and consistent data ready for efficient querying and visualization.

# App Setup & Coding



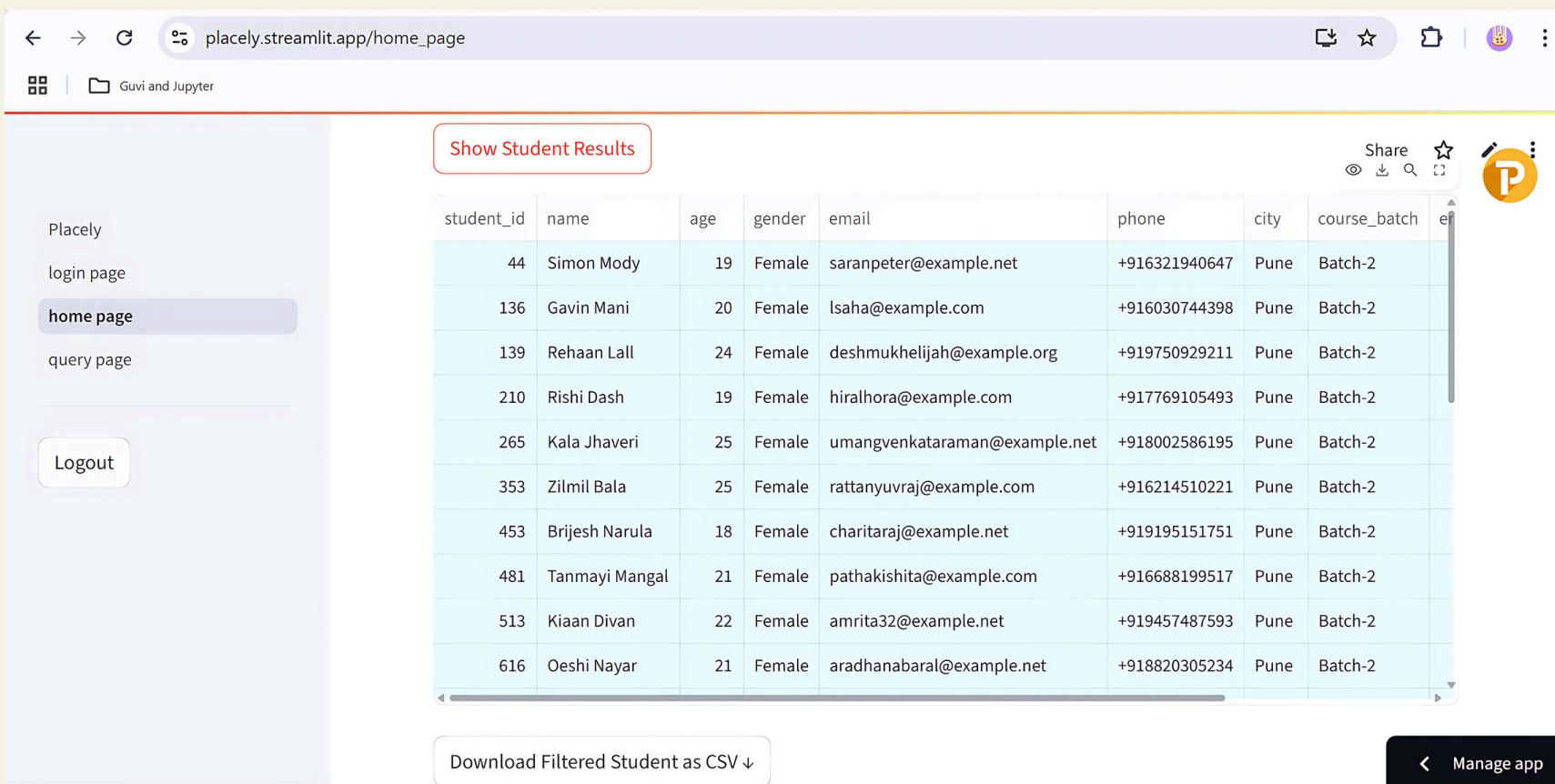
The Placely app was built using Streamlit and organized into separate Python files to keep the project modular and clean. The main file, `Placely.py`, handles page routing and caching for faster performance. The app includes a welcome page with a project introduction, a home page for filtering where users can explore data, apply multiple filters and a query page that runs predefined SQL queries based on user selection. Throughout, use of pandas was made to manage and display dataframes and to add download buttons to export filtered data as CSV. This setup ensures the app remains interactive, fast, and easy to navigate.

# Deployment



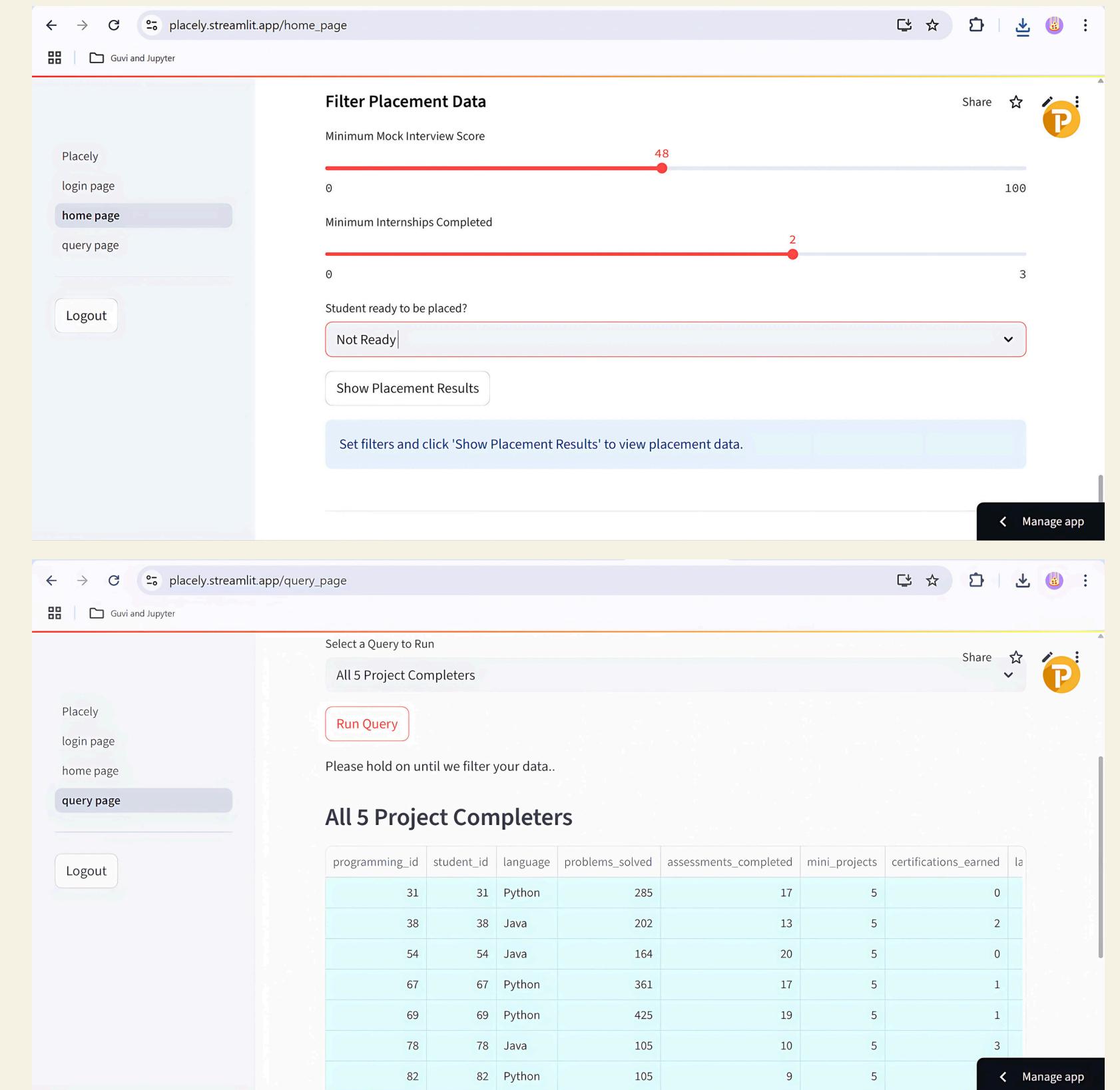
For deployment, the project was migrated from a localhost MySQL setup to an online database hosted on Railway, ensuring reliable cloud access. Connection settings were configured to securely link the Railway database with the Streamlit application, allowing real-time data retrieval and updates. The complete project, including all Python scripts and supporting files, was then pushed to GitHub, with commit and an updated README.md documenting the setup and usage instructions. Finally, the Streamlit app was deployed online, making Placely accessible to users through a simple web interface.

# Deployment



The screenshot shows the Streamlit home page for the 'Placely' application. On the left, there's a sidebar with links for 'Placely', 'login page', 'home page' (which is selected), and 'query page'. Below these are 'Logout' and 'Manage app' buttons. The main area features a table titled 'Show Student Results' with columns for student\_id, name, age, gender, email, phone, city, course\_batch, and experience. A red-bordered button labeled 'Show Student Results' is positioned above the table. At the bottom, there's a 'Download Filtered Student as CSV' button and another 'Manage app' button.

The app starts with a login page, after successful login, users are automatically redirected to the home page. On the home page, users can choose options from dropdown menus to narrow down student data. A prompt guides them to "Set filter to get data" and clicking the "Show Result" button instantly displays the filtered results. Additionally, a download button makes it easy to export this data as a CSV file for offline use.



The screenshot shows the Streamlit query page for the 'Placely' application. On the left, there's a sidebar with links for 'Placely', 'login page', 'home page', and 'query page' (which is selected). Below these are 'Logout' and 'Manage app' buttons. The main area has a section titled 'Filter Placement Data' with two sliders: 'Minimum Mock Interview Score' set to 48 and 'Minimum Internships Completed' set to 2. It also includes a dropdown for 'Student ready to be placed?' with the value 'Not Ready', a 'Show Placement Results' button, and a note: 'Set filters and click 'Show Placement Results' to view placement data.' Below this is another section titled 'Select a Query to Run' with the option 'All 5 Project Completers' and a 'Run Query' button. A progress message 'Please hold on until we filter your data..' is displayed. At the bottom, there's a table titled 'All 5 Project Completers' with columns for programming\_id, student\_id, language, problems\_solved, assessments\_completed, mini\_projects, certifications\_earned, and last\_updated. The table contains 8 rows of data.



[Click Here to get redirected to the app](#)

[Github Repo Link](#)

Thank You!

Created by Kalpita Pawase