A Project Report

on

SKILL GAP ANALYSER

A Capstone Project for the #30DaysOfPython Challenge

Contributors:

KATARU SHREYA

SURYA PRAKASH R

SUCHITHA KAMARAPU

Organised by: INDIAN DATA CLUB

ABSTRACT

In today's competitive job market, many students and freshers apply to roles without a clear understanding of what skills are required. This often results in repeated rejections and self-doubt.

The Skill Gap Analyzer project addresses this by offering a personalized, visual, and interactive solution. It allows users to select an aspiring career, self-assess their current skills, and instantly visualize the gap. This project was built as part of a 3-member capstone project and focuses on empowering students to approach job readiness strategically.

INDEX

1. INTRODUCTION

The Skill Gap Analyzer is an application built using Python and Kivy, aiming to bridge the gap between academic knowledge and industry expectations. The app enables users to assess their current proficiency levels against the skills required for their desired roles. It provides a clear visualization of these gaps using bar charts and delivers a personalized 3-month learning roadmap in downloadable PDF format.

The tool is intuitive, visual, and practical, empowering users to take actionable steps toward their career goals.

2. PROBLEM STATEMENT

In today's fast-evolving job market, many individuals aspire to roles such as Data Analyst, Backend Developer, or UI/UX Designer but lack clarity on what skills are truly expected. Job descriptions can be vague or overwhelming, and self-assessment is often inaccurate or based on guesswork. This mismatch between self-perception and actual expectations leads to:

- Increased rejection in job applications,
- Wasted time pursuing irrelevant learning resources,
- Low confidence and discouragement.

This project helps anyone get career-ready by making it easy to spot their skill gaps and create a clear plan to improve.

3. OBJECTIVES AND SCOPE OF THE PROJECT

3.1 OBJECTIVES

- To enable users to select a desired career path.
- To help users identify skills needed for their dream role.
- To enable self-assessment of skills with a rating mechanism.
- To display required industry-standard skills for the selected role.
- To visualize the difference between current skill level and required skill level.
- To generate a personalized 3-month learning plan based on gaps.
- To export this information into a clean and readable PDF.

3.2 SCOPE OF THE PROJECT

- Covers role selection, skill input, and personalized gap visualization.
- Currently a Kivy based application; can be scaled to web or mobile.
- Can be used by students, freshers, bootcamp participants, and career switchers.
- Can be integrated with web technologies, backend APIs, and external databases in the future.

4. HARDWARE AND SOFTWARE REQUIREMENTS

4.1 HARDWARE REQUIREMENTS

- Processor: Any modern CPU (Intel Core i3/i5/i7, AMD Ryzen 3/5/7)
- RAM: 4GB minimum (8GB recommended for faster processing)
- Storage: At least 5GB of free space (for libraries and temporary data)

4.2 SOFTWARE REQUIREMENTS

- Operating System: Windows / Linux / macOS
- Python: 3.7 or higher
- Required Libraries: kivy, matplotlib, fpdf, json, os
- IDE: Any code editor (VS Code, PyCharm, etc.)

5. LIBRARIES USED

The below are the libraries used in this project:

Library	Purpose	Details
kivy	GUI creation	Used for building the entire interface including layout elements (BoxLayout, GridLayout, ScrollView), input fields (TextInput), dropdowns (Spinner), and buttons. Enables cross-platform desktop UI.
json	Data storage and retrieval	Handles reading from roles.json, target_levels.json, and roadmaps.json, and writing user input data to users.json. JSON ensures easy scalability and human-readable data format.
os	File handling	Used to check if certain files exist before attempting to load or write data, preventing crashes due to missing files.
matplotlib.pyplot	Data visualization	Generates a bar chart comparing the user's self-ratings against expected industry standards. The chart is saved as an image and embedded in the PDF.
fpdf	PDF generation	Creates a multi-page downloadable report that includes: the chart, a skill gap analysis table, and a custom roadmap based on the chosen role. Text formatting and layout are handled via this module.

6. IMPLEMENTATION OVERVIEW

1: Application Layout (GUI Design using Kivy)

- The main layout is created using a vertical BoxLayout to stack all components.
- User inputs include:
    - Text fields for name and current profession.
    - A dropdown to select an aspiring job role.
    - A scrollable grid layout to display skills based on the selected role.
    - Each skill row contains:
        - A label with the skill name and description.
        - A numeric rating input.
        - Increment (+) and decrement (-) buttons to adjust ratings.

2: Dynamic Skill Loading and Rating

- When a role is selected from the dropdown:
    - Skills and descriptions are loaded from roles.json.
    - Each skill is displayed along with an input for the user to rate their proficiency (from 0 to 10).
    - Ratings are collected into a dictionary for processing.

3: Saving User Input

- On clicking the "Submit" button:
    - The app validates that all fields are filled.
    - A dictionary containing username, status, role, and skill ratings is created.
    - This data is appended to users.json.

4: Skill Gap Chart Creation (Matplotlib)

- The app compares the user's skill ratings with predefined target ratings.
- A bar chart is generated:
    - Blue bars represent user ratings.
    - Pink bars represent target levels.
- The chart is saved as skill_gap_chart.png.

5: Roadmap & PDF Generation (FPDF)

- After submission, a “Download PDF” button becomes active.
- When clicked:
    - A PDF file is generated using fpdf.
    - Page 1 contains the skill gap chart.
    - Page 2 contains a table comparing each skill’s self-rating, target rating, gap, and a short suggestion.
    - Page 3 contains a 3-month personalized learning roadmap from roadmaps.json.
    - The final report is saved as Skill_Roadmap.pdf.

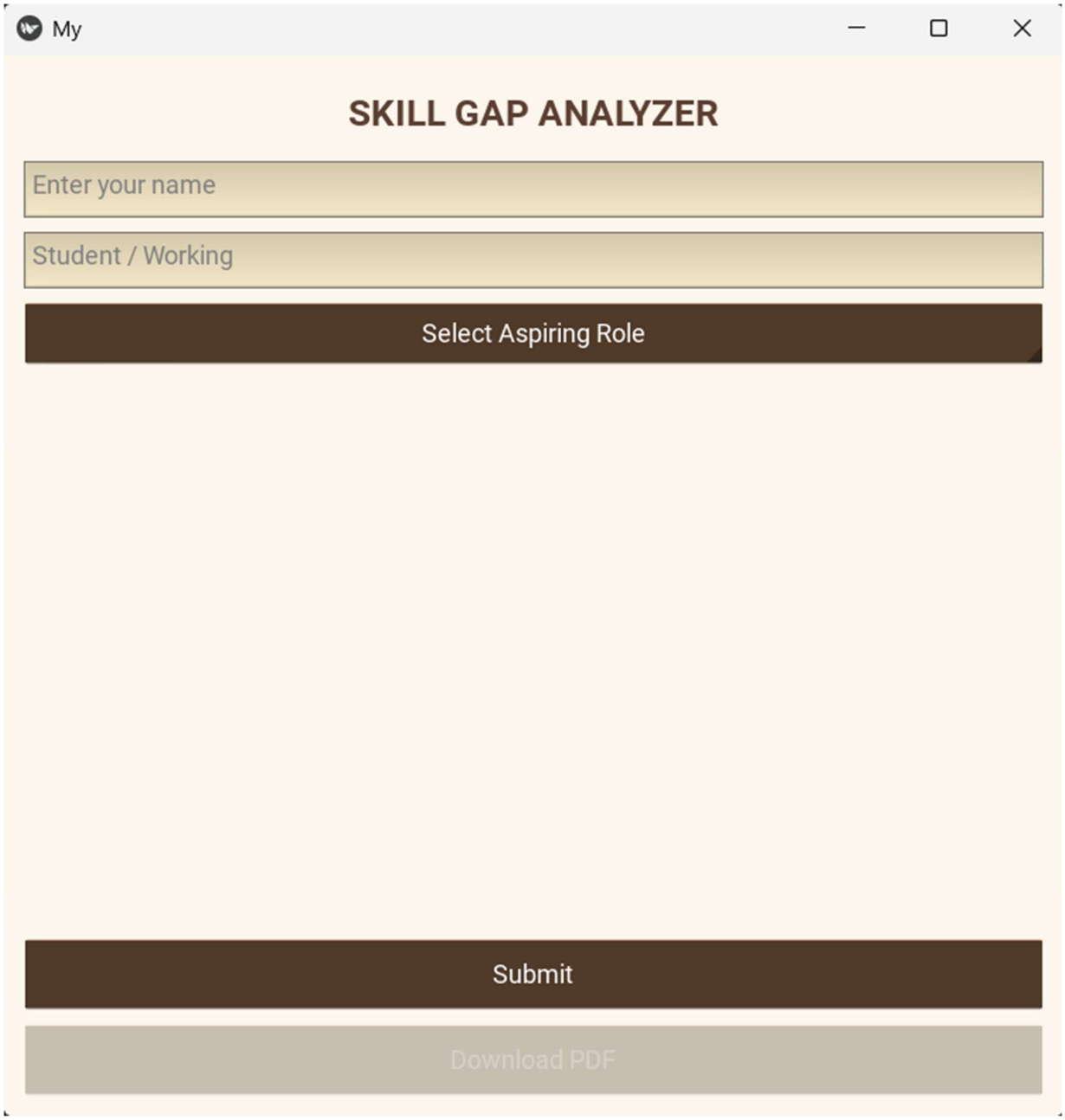
6: Popups and Feedback

- Popups notify the user about successful submissions, errors, or download completion.
- These are created using Popup and dismissed using a button or automatically with Clock.

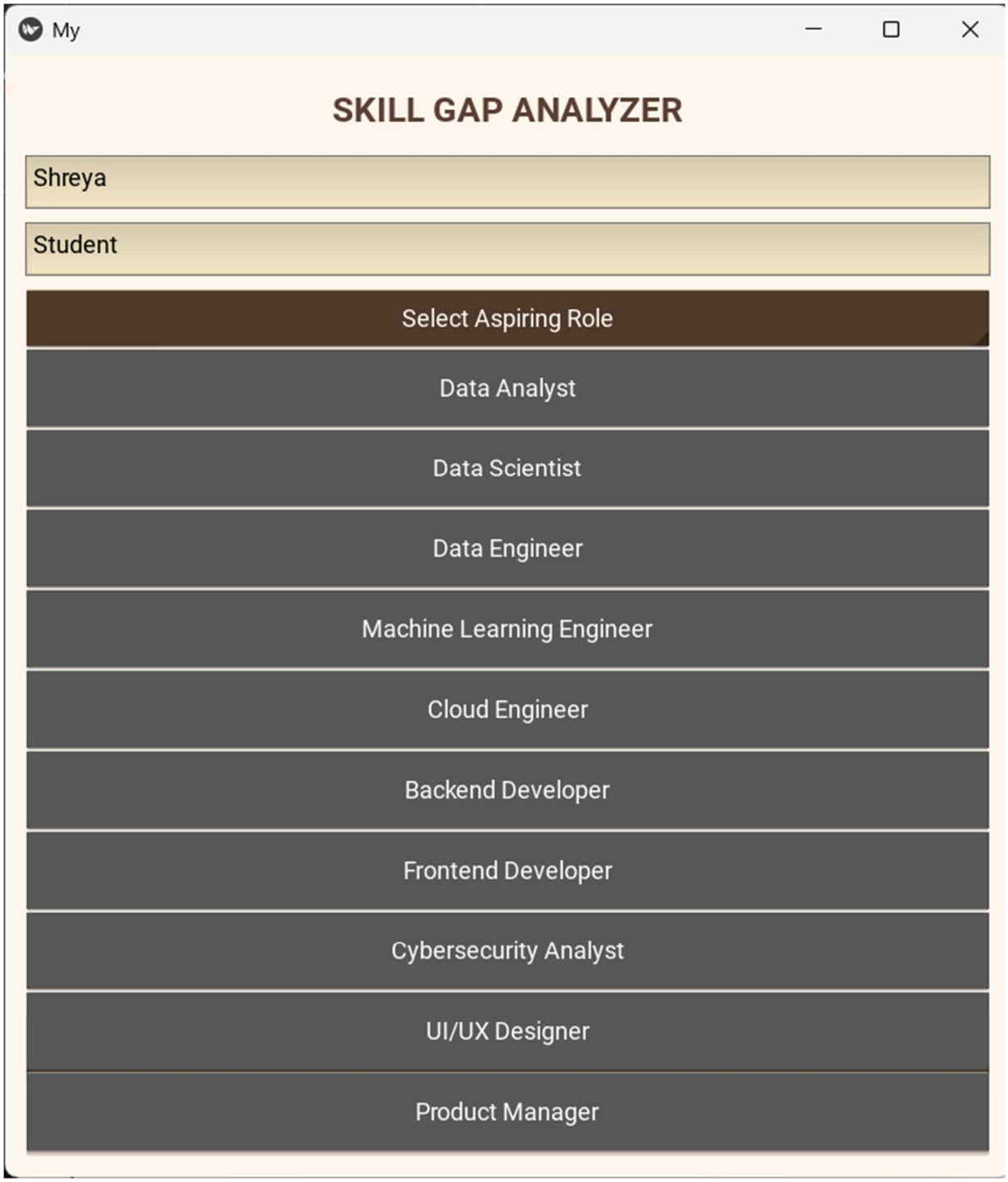
7: Data Files Used

- roles.json: Defines job roles and required skills with descriptions.
- roadmaps.json: Stores 3-month learning plans for each role.
- users.json: Collects and stores data entered by users for future reference or analysis.

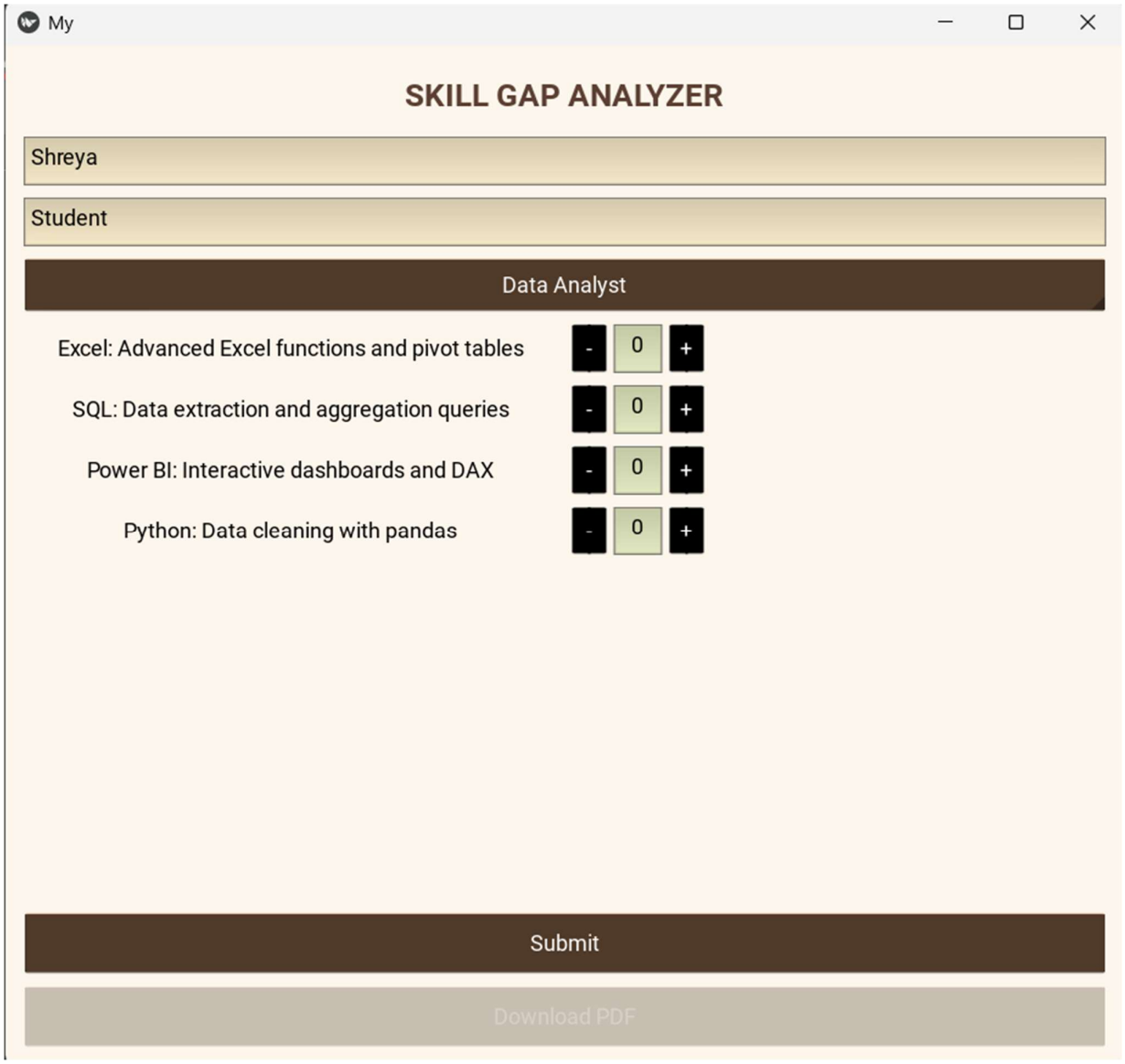
7. OUTPUT SCREENS



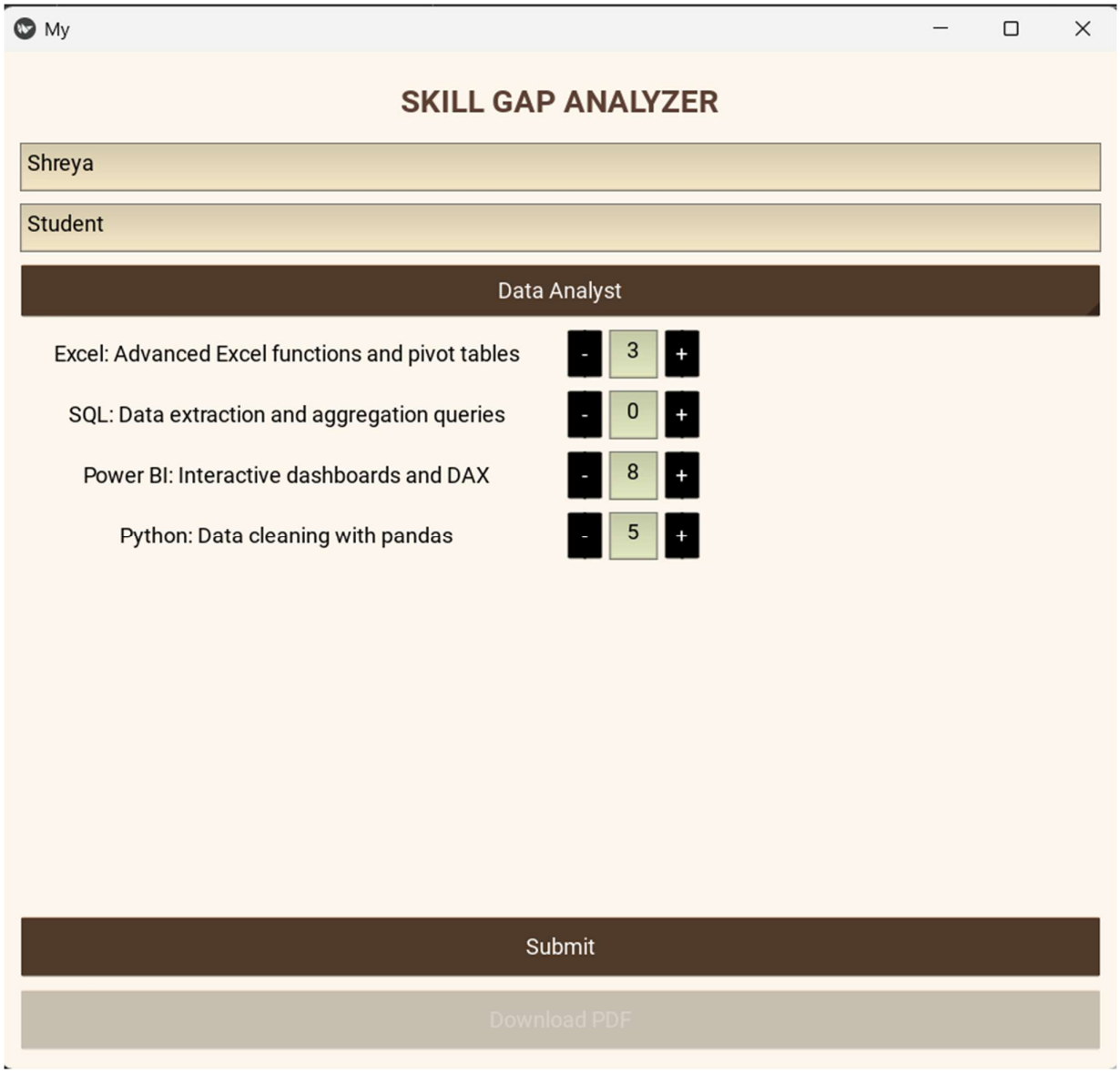
(Fig 1: Homepage)



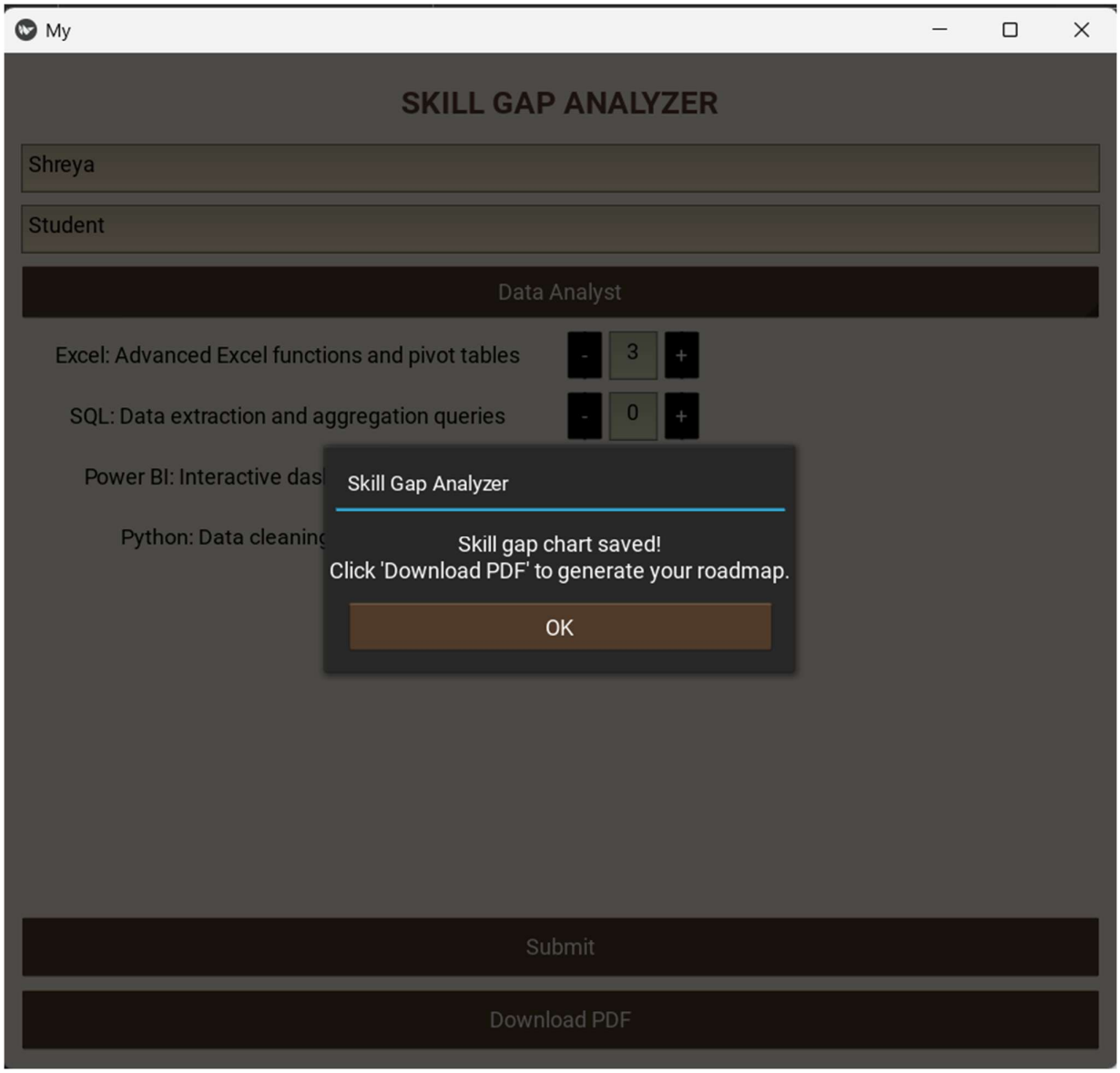
(Fig 2: Entered name, profession and clicked "Select aspiring role" option. A list of roles is displayed.)



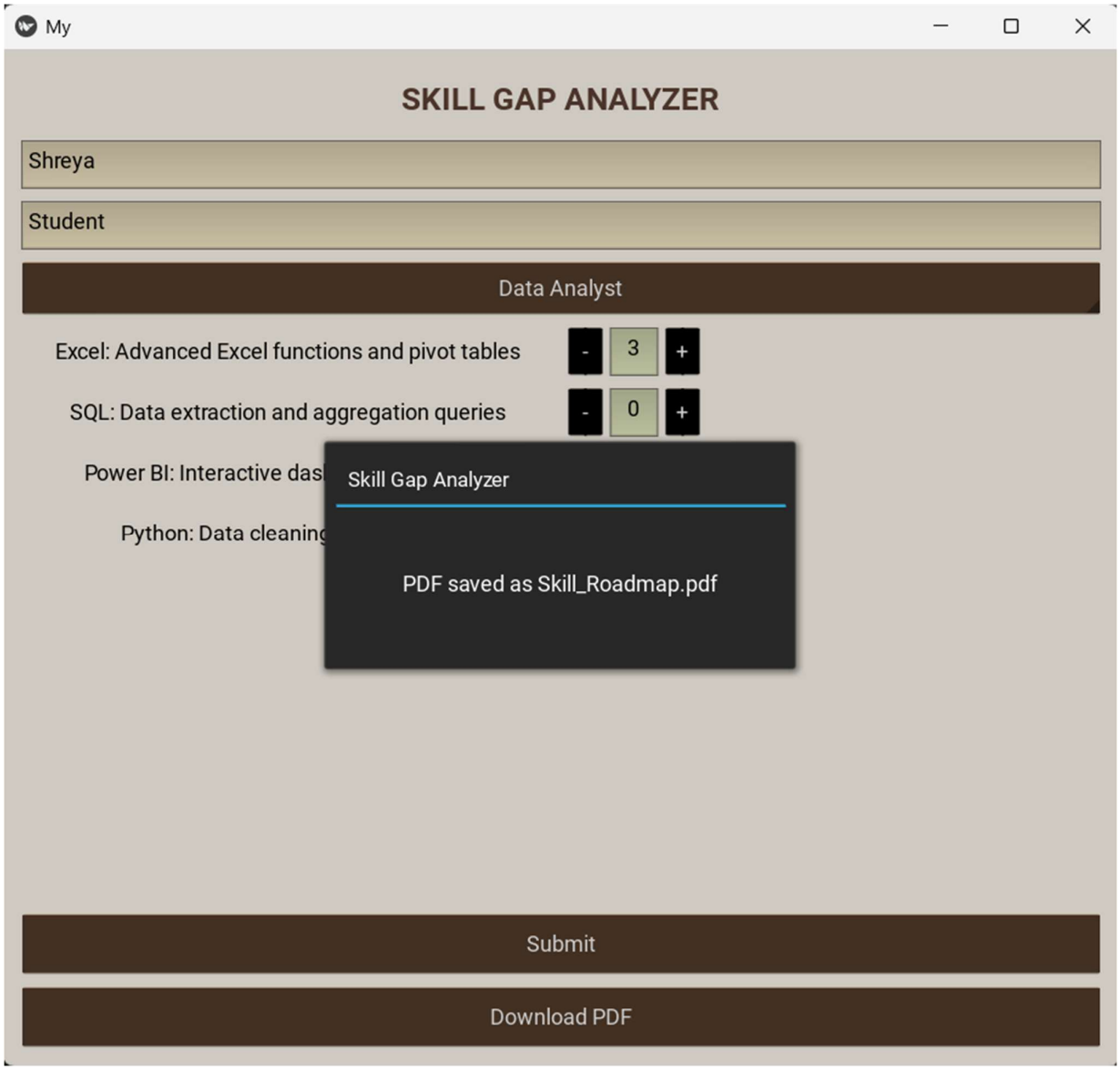
(Fig 3: Selected Data Analyst role and the essential skills necessary to become a data analyst are displayed. User can rate their current proficiency in those skills ranging from 0-10.)



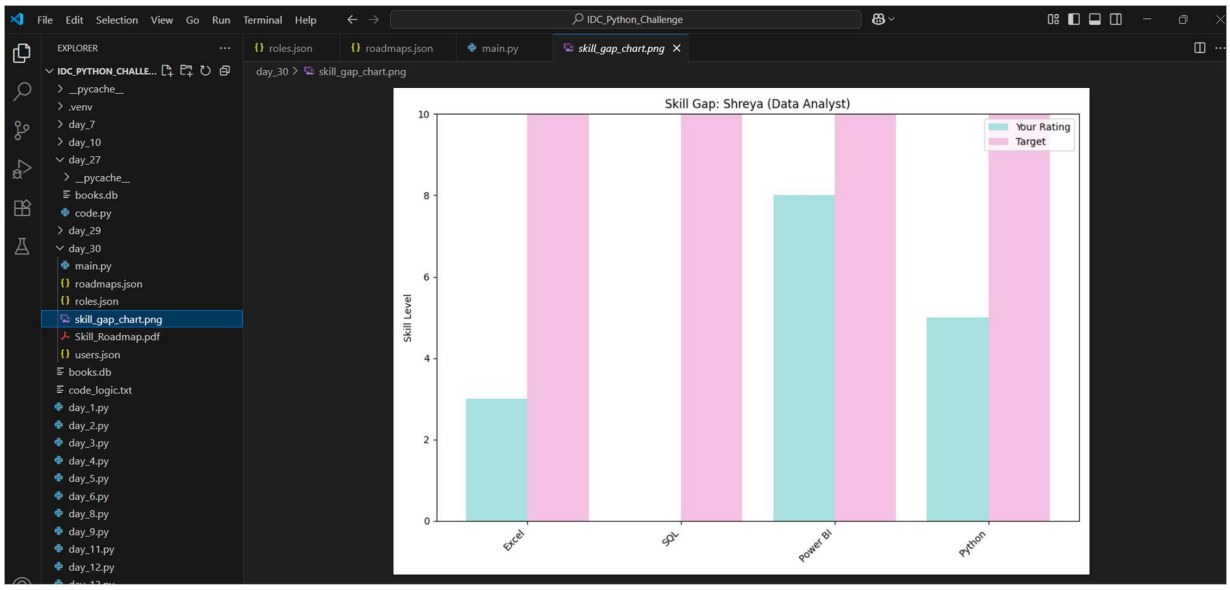
(Fig 4: User entered his/her level of proficiency in the specified skills.)



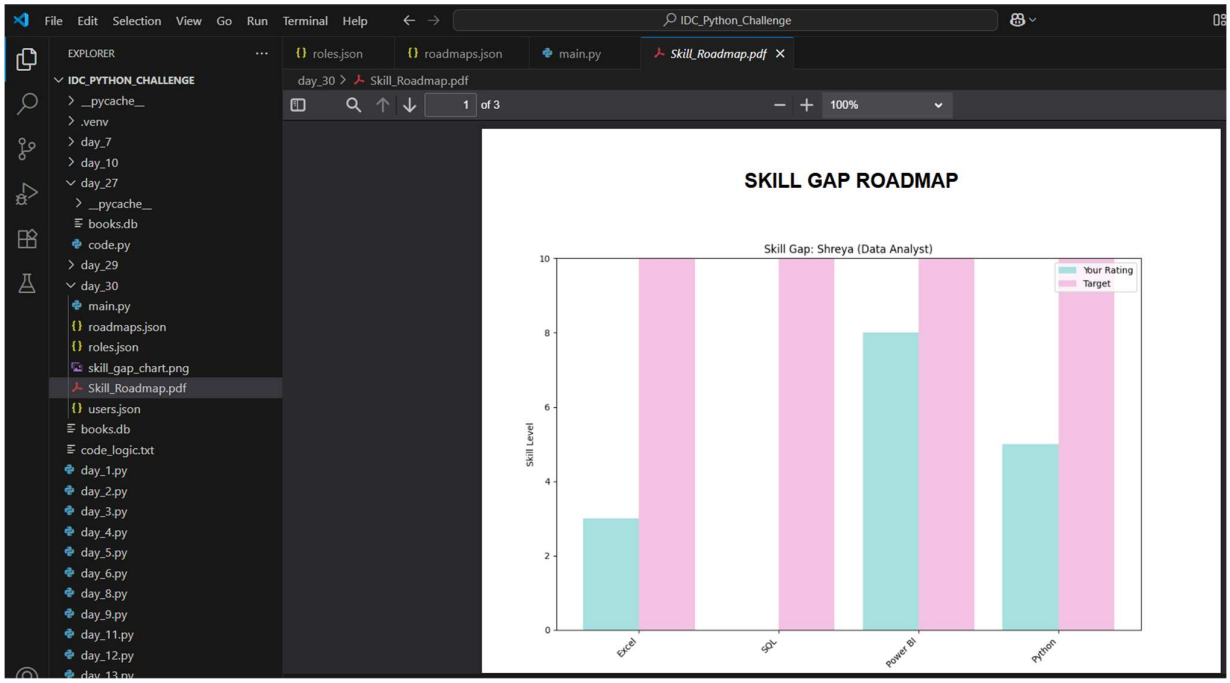
(Fig 5: User clicked on submit button and a popup is displayed)



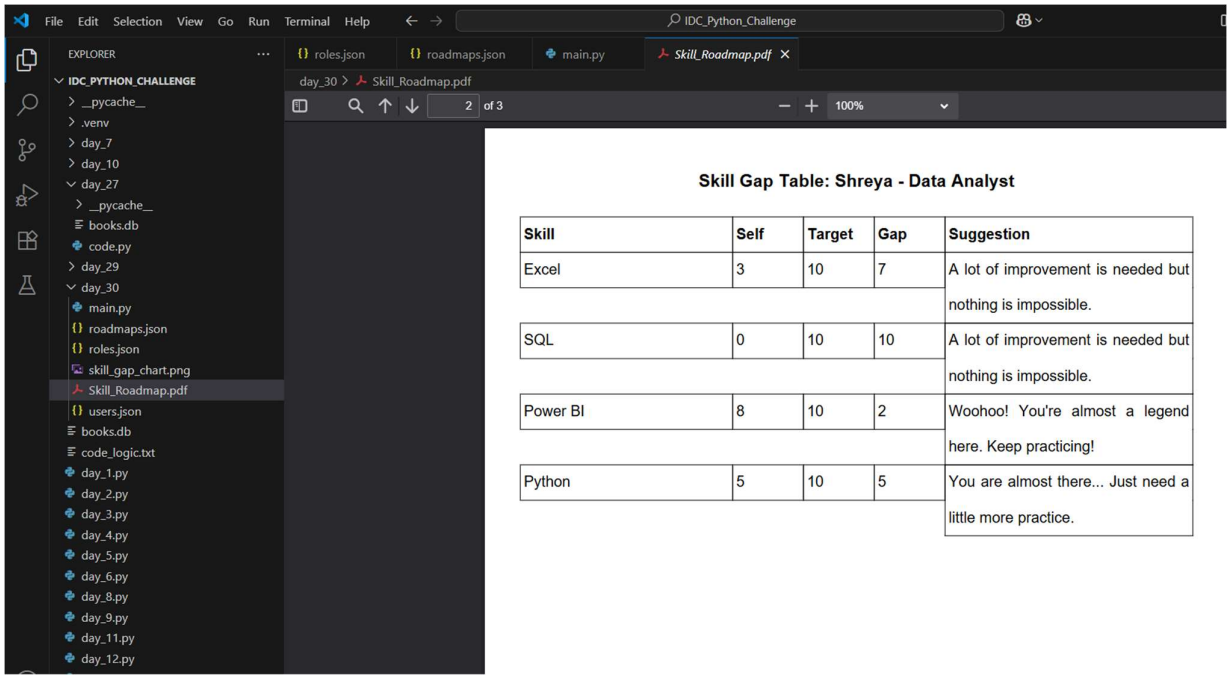
(Fig 6: User clicked ok in the previous popup window and clicked on download PDF button)



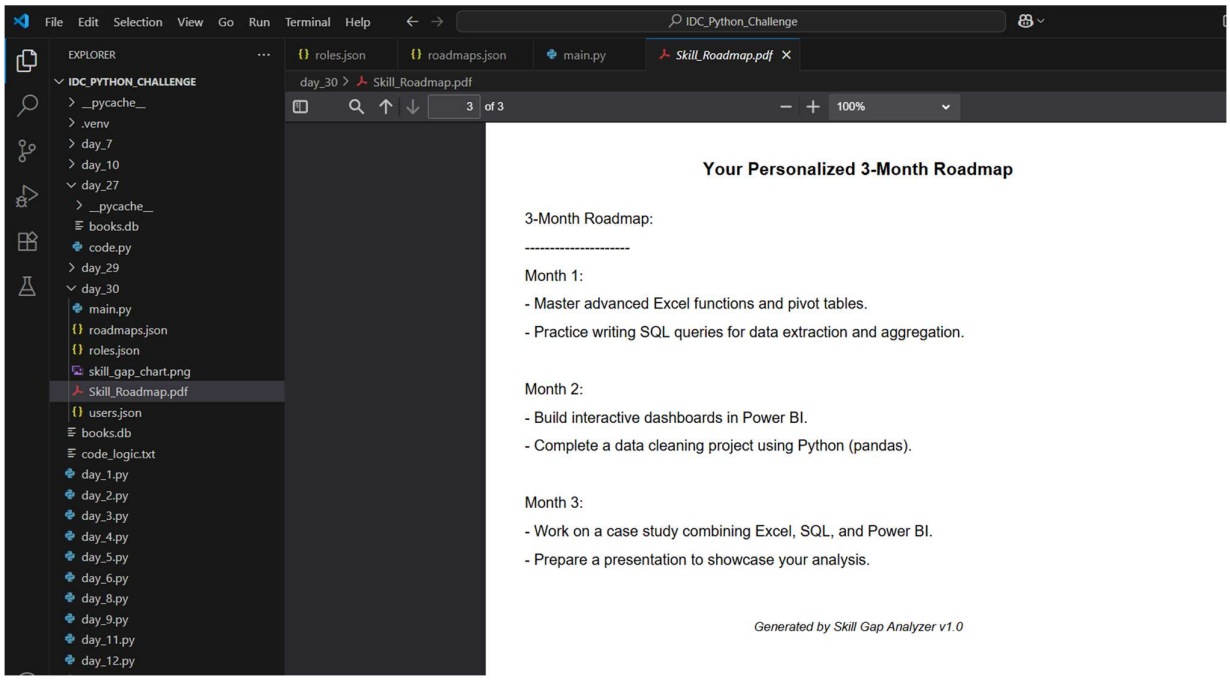
(Fig 7: skill_gap_chart of the user)



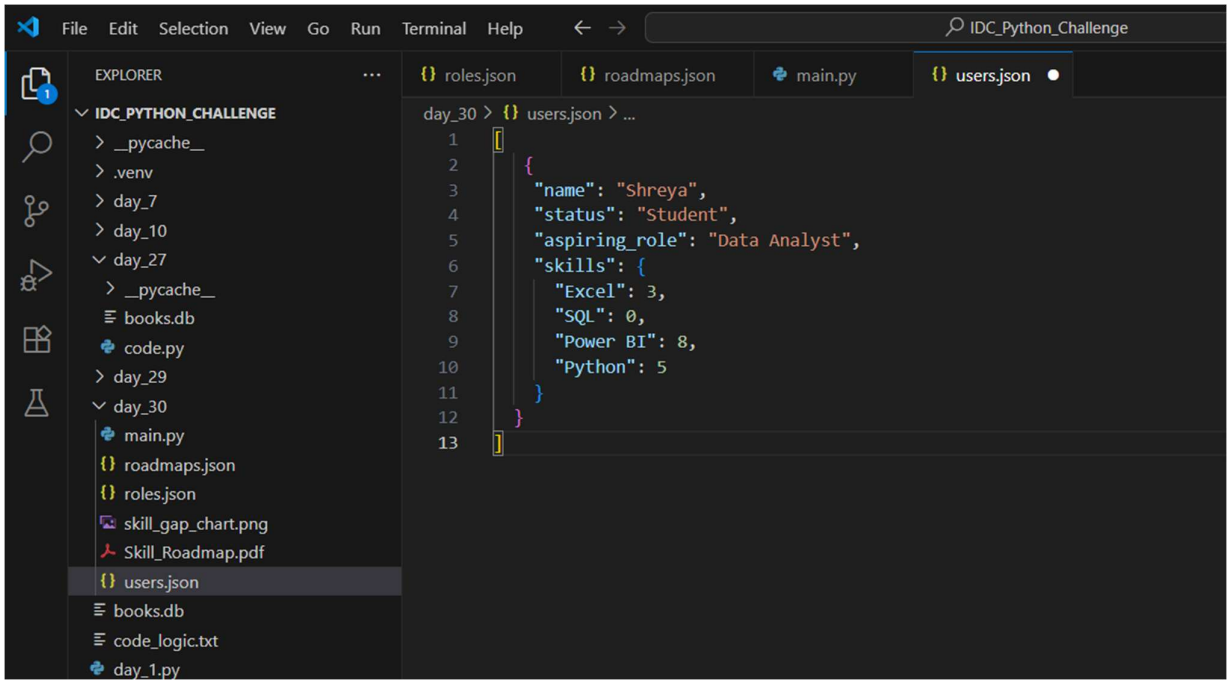
(Fig 8: First page of the user's personalised roadmap showcasing the data as a visual)



(Fig 9: Second page of the user's personalised roadmap showcasing a summary table of the user's skills level, target level, gap between the current level and the target level and suggestions)



(Fig 10: Third page of the user's personalised roadmap displaying the 3-month personalised roadmap for the user to get closer to achieving his dream role)



(Fig 11: users.json storing the user's data)

8. CONCLUSION

The Skill Gap Analyzer simplifies the process of career planning for beginners. It guides users to reflect on their current capabilities and take informed steps toward improvement. The project successfully integrates design, logic, data handling, and visualization, forming a foundation that can evolve into a complete career planning tool in the future. It helps individuals make informed learning decisions and boosts their chances of landing the right job.

9. FUTURE SCOPES

- **Web App Integration**: Build the same functionality as a web app using FastAPI or Flask.
- **Database Support**: Use SQLite or MongoDB for persistent user progress tracking.
- **AI Recommendations**: Suggest courses or learning paths using ML models based on skill gaps.
- **Job Matching**: Integrate with job APIs to suggest roles based on skill ratings.
- **User Login System**: Let users create accounts and return later to continue learning.