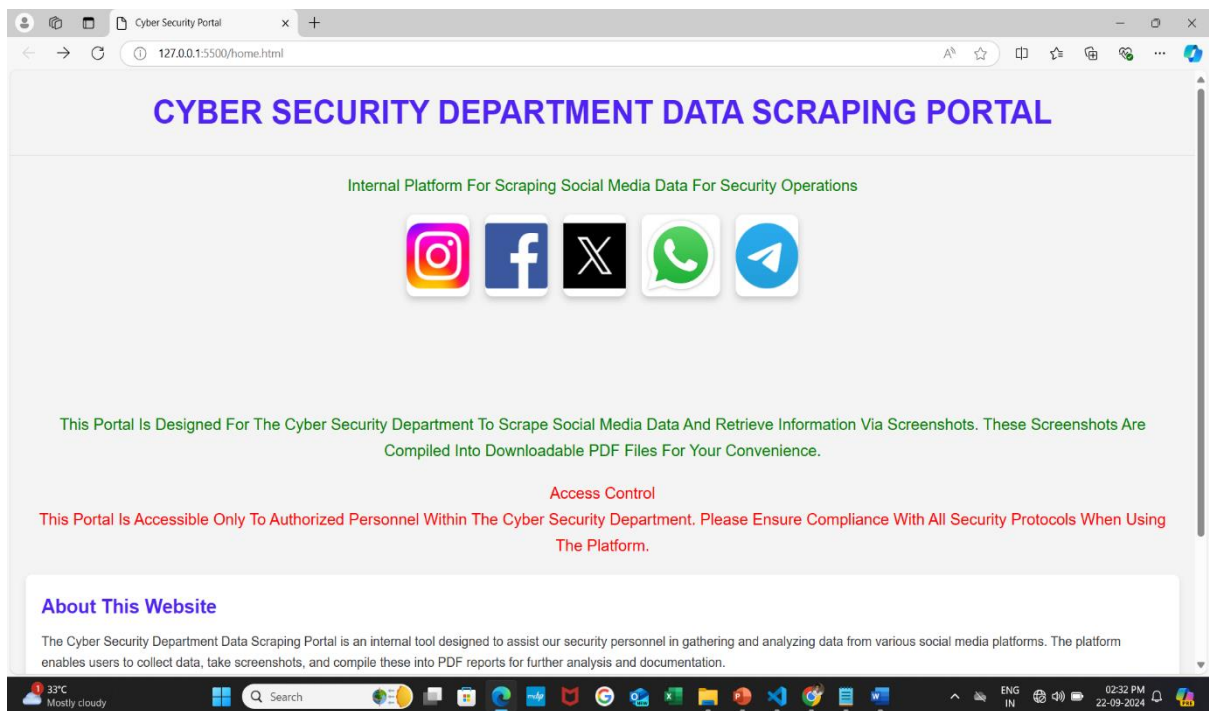


Detailed Working Procedure of the Project

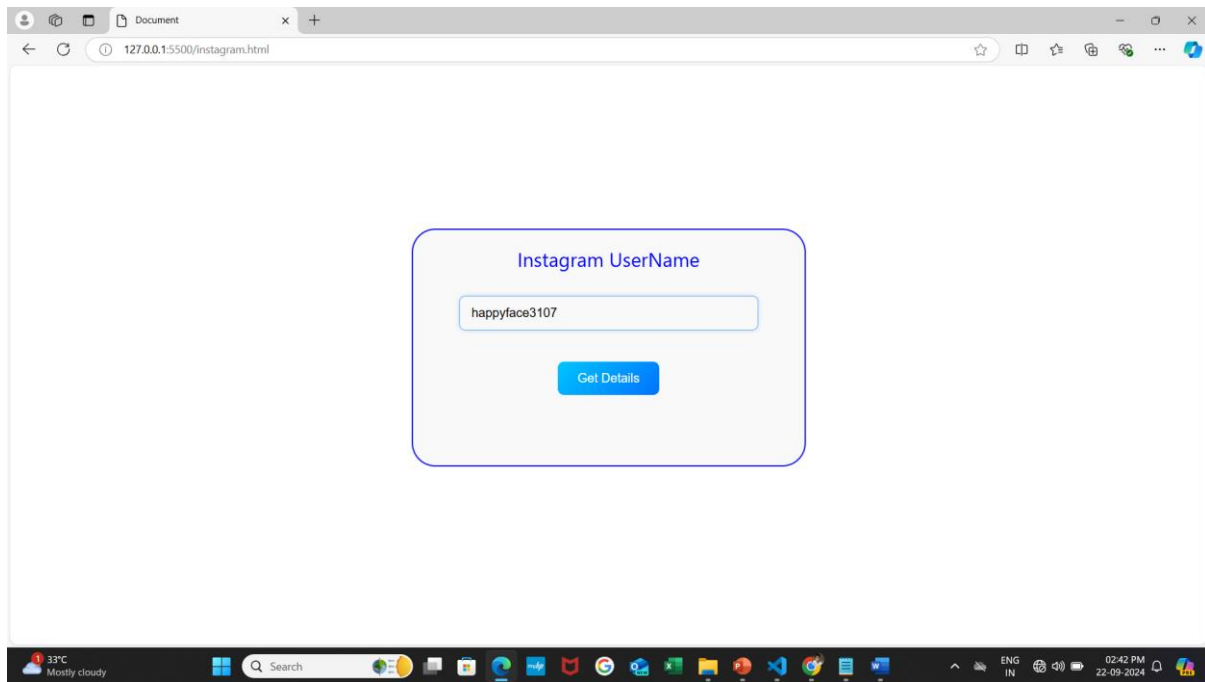
1. Platform Selection (Frontend Interface)

Technology: HTML/CSS/JavaScript The user starts by visiting the website. On the homepage, they are presented with a choice of five social media platforms: Instagram, Facebook, Twitter, WhatsApp, or Telegram. The website interface dynamically adapts based on the user's selection. For example: If the user selects Instagram, they are asked to enter a username or profile URL. If they select WhatsApp or Telegram, they are prompted to input a mobile number. This selection triggers a form submission, sending the chosen platform and user input to the backend.



2. User Input (Username, Profile URL, or Mobile Number)

Technology: HTML/CSS/JavaScript Once the platform is selected, the user enters the relevant data: For Instagram, Facebook, or Twitter: They enter a username or profile URL of the account they wish to scrape. For WhatsApp or Telegram: They input a mobile number. The form submission is validated on the client side using JavaScript to ensure the input meets the expected format (e.g., URL format or a valid phone number). After validation, the input data is sent to the backend server for further processing via an API call.

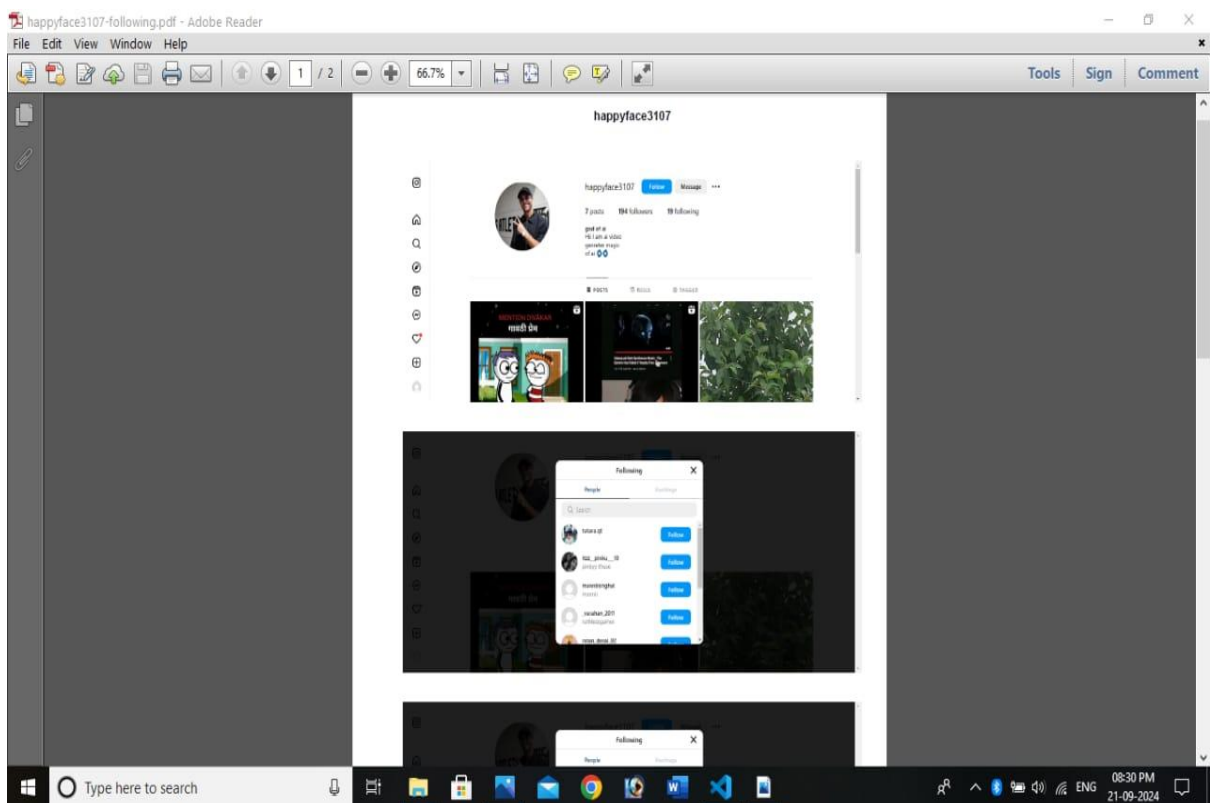
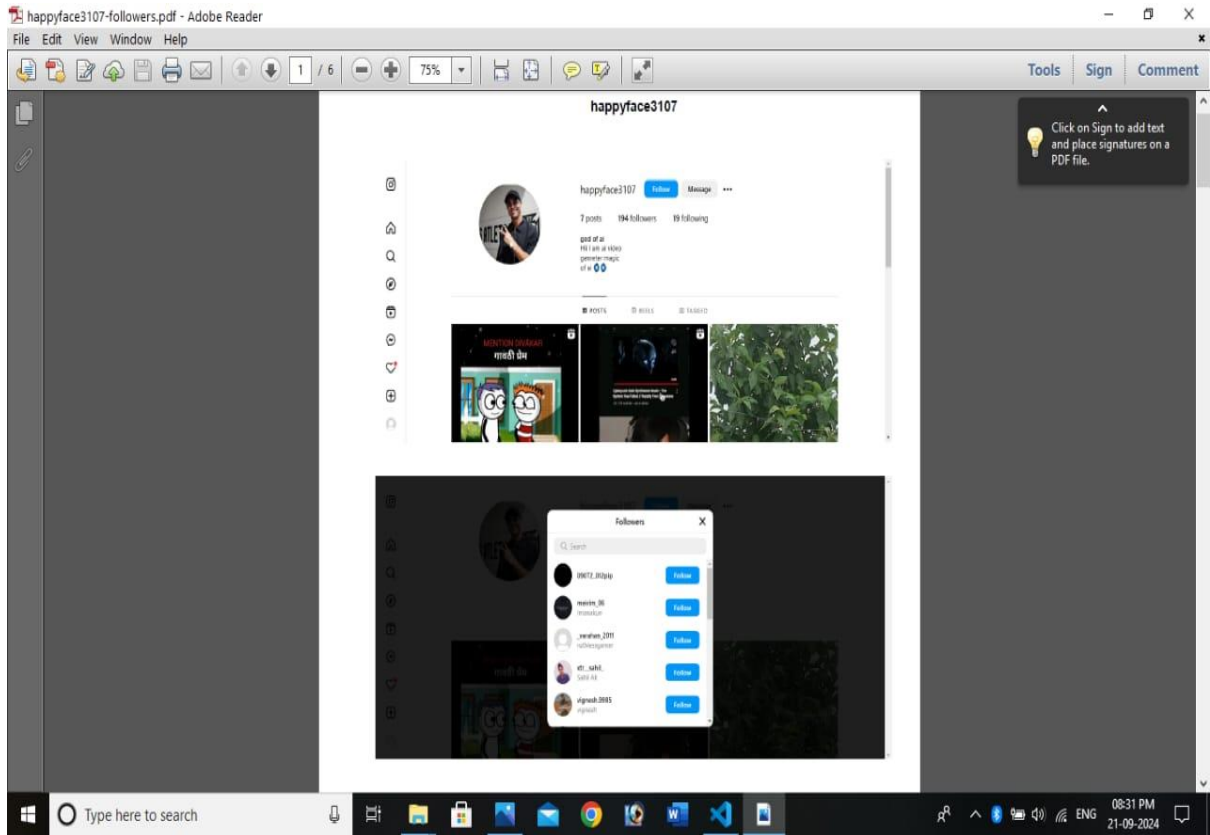


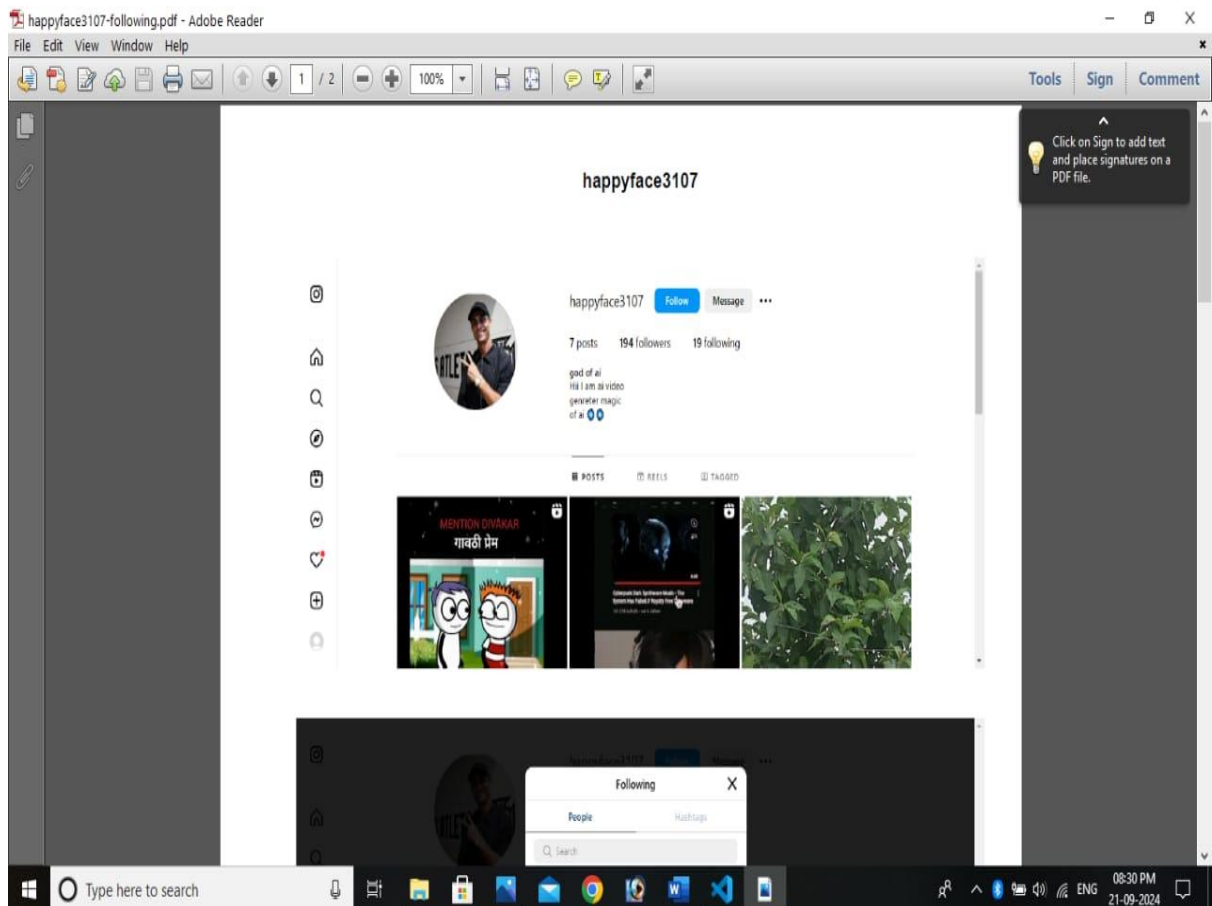
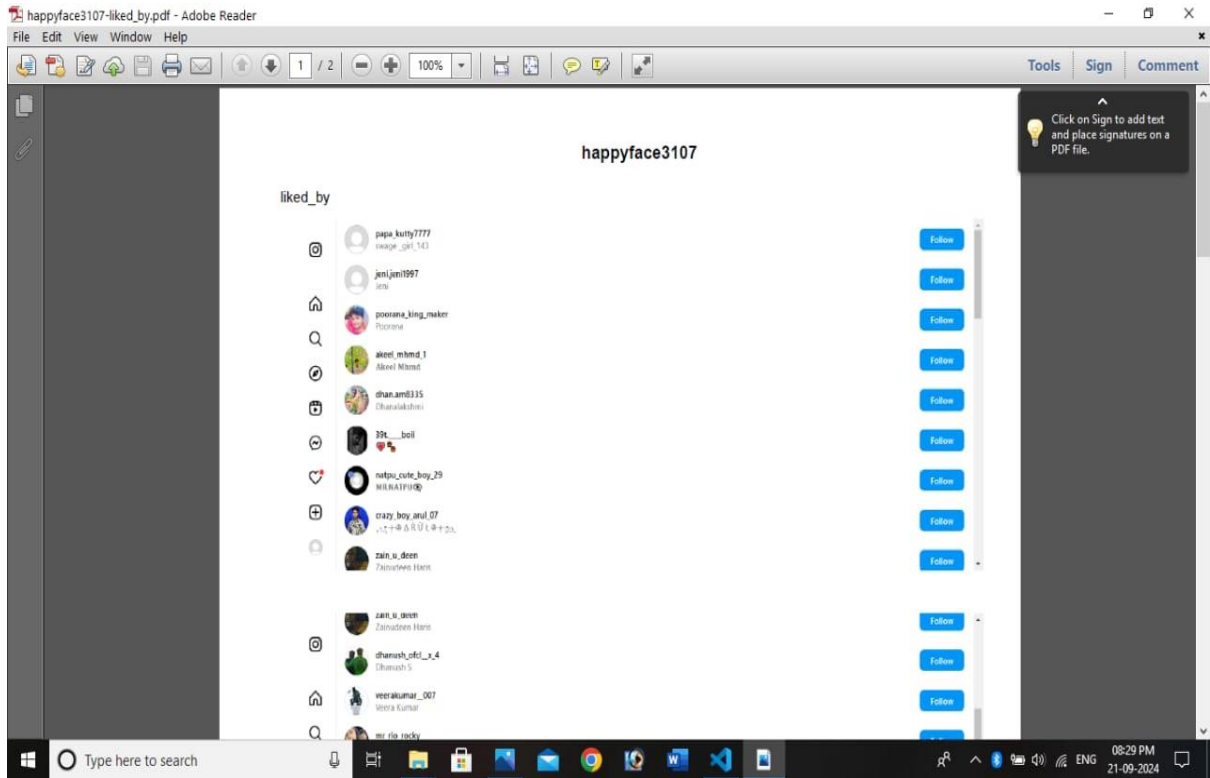
3.Data Collection (Backend with API or Web Scraping)

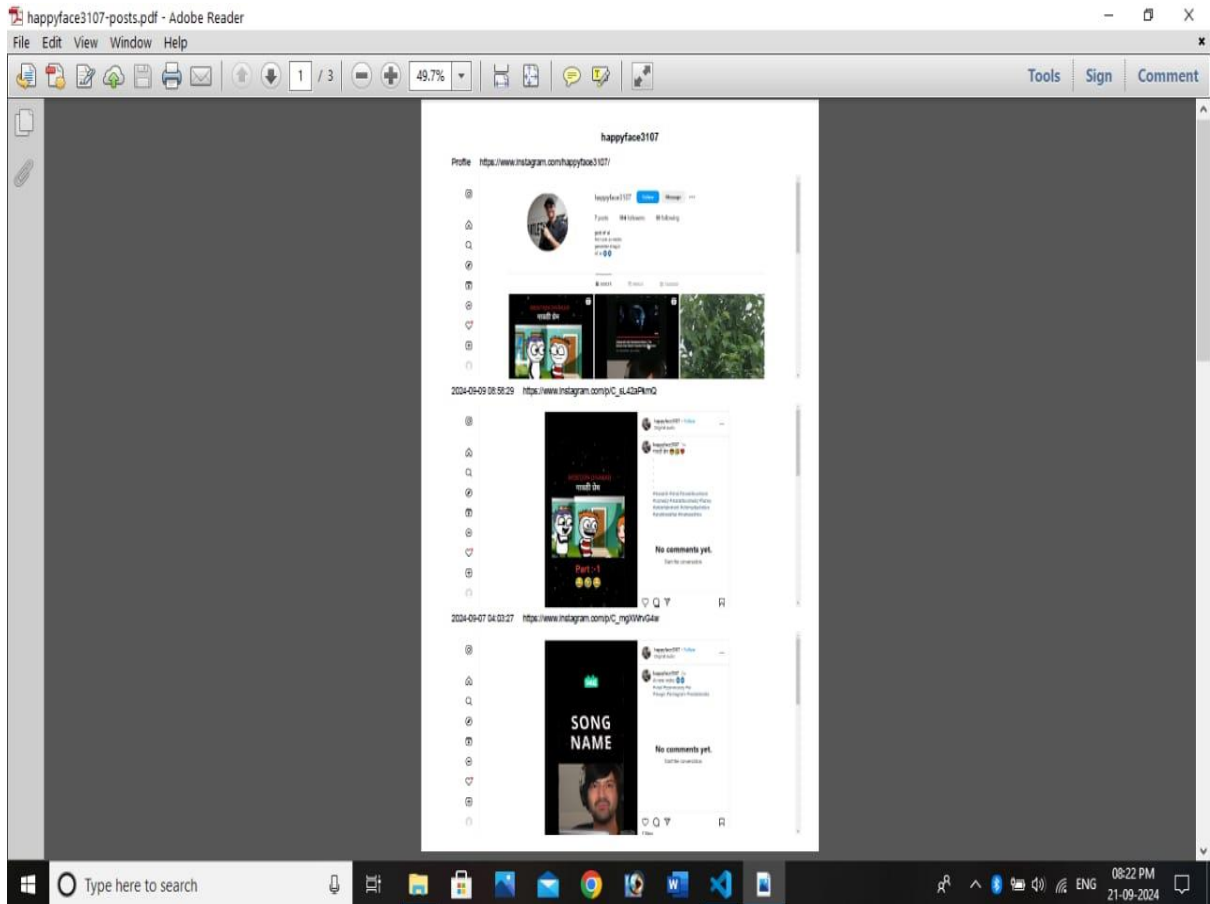
Technology: Python (APIs, Selenium, BeautifulSoup) The backend system begins collecting the required data based on the platform selected: APIs: For platforms with accessible APIs (like Instagram, Facebook, and Twitter), the backend Python scripts send requests to the platform's API using the input (username or profile URL). The API returns structured data such as posts, comments, likes, or profile information. WhatsApp: Using the WhatsApp Business API, the system retrieves user messages or interaction data linked to the mobile number provided by the user. Telegram: Python uses the Telegram Bot API or Telethon to interact with Telegram servers and fetch messages or profile data using the input mobile number. Web Scraping: For platforms with no API access (or restricted API), the system uses Selenium to automate browser actions like logging in, navigating pages, and extracting content. BeautifulSoup then parses the HTML content to retrieve structured data (e.g., post text, comments, and likes).

4.Automated Screenshots (Backend)

Technology: Python (PyAutoGUI) After data collection is complete, the system takes screenshots of the data (e.g., posts, profiles, comments) to provide visual documentation. PyAutoGUI automates the browser or application window, taking screenshots of the desired sections (for instance, capturing a post, comment section, or profile). It can also simulate user interactions like scrolling or clicking on particular posts to capture the entire content. These screenshots are saved as image files on the backend server, ready for the next stage (PDF generation).

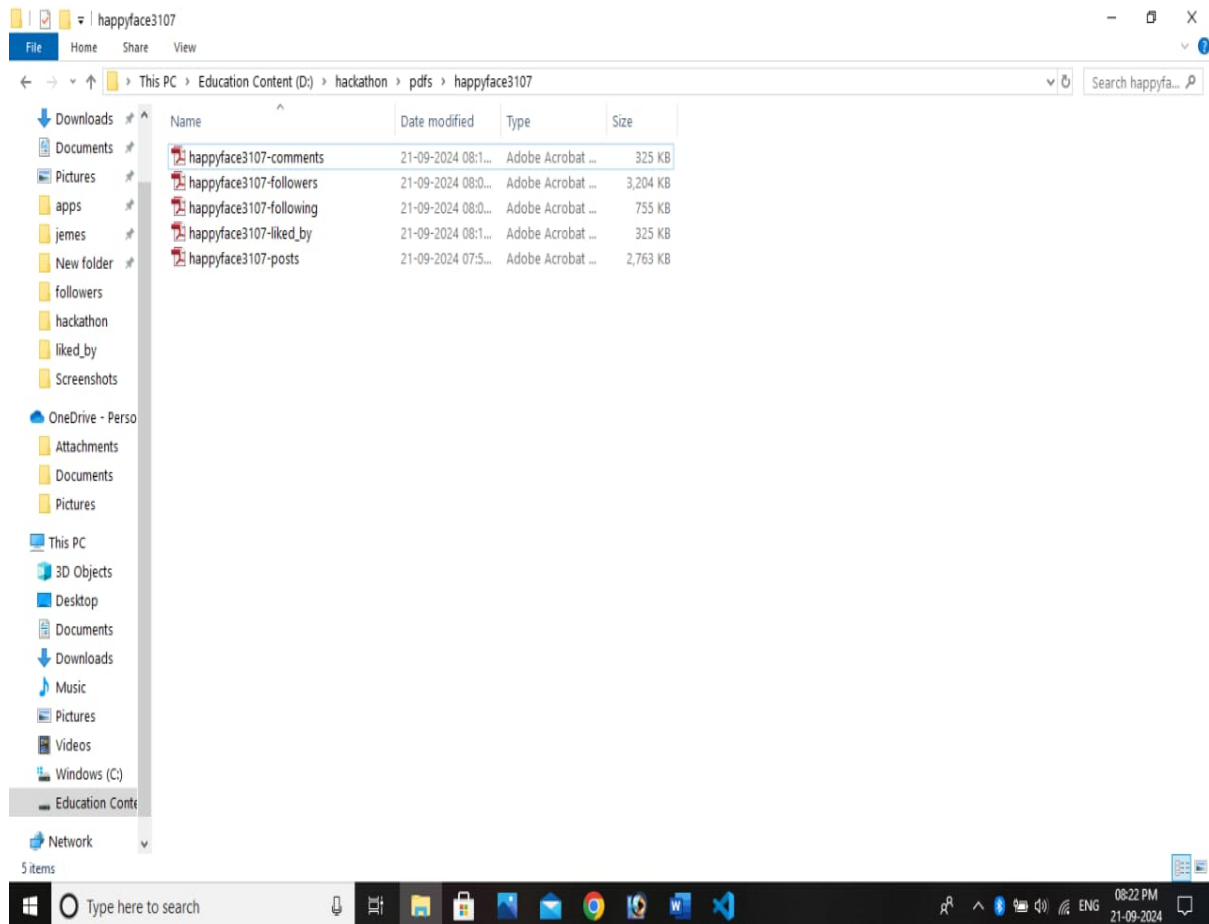






5.PDF Generation (Backend)

Technology: Python (Pillow, FPDF) The captured screenshots are compiled into a report using Pillow or FPDF, both of which are Python libraries that allow image manipulation and PDF creation. Pillow formats the images, allowing them to be resized or annotated if needed. FPDF organizes the screenshots into a multi-page PDF report, along with any additional data (such as text from posts, profile details, or comments). The PDF is created in a structured format, making it easy to read, and is saved on the server for user download.



6.Display & Download (Frontend Interface)

Technology: HTML/CSS/JavaScript Once the PDF is generated, the website interface displays the data collected and the screenshots in a user-friendly format. Users can browse through the information directly on the website and are given the option to download the generated PDF. The Download PDF button allows users to save the PDF report on their local device.

7.Output to User

The user now has access to: The real-time data that was scraped from the selected social media platform, displayed on the website. A PDF report containing screenshots of the posts, profiles, and other relevant data, which they can download for further use. The output provides a complete, documented view of the scraped social media data, with screenshots and structured content in a convenient, portable PDF format.