Project Plan

Rana Feyza Soylu, Dipika

Bhandari, Saikrishna Laxmipathi Gundeti

Illinois Institute of Technology; College of Computing

ITMS 448/548 Cyber Security Technology

Dr. Maurice Dawson

Contents

1. Tasks

The tasks were divided up among the team members equally. Each team member worked on at least one task to ensure adequate teamwork among all members of the group.
The tasks that were assigned were:

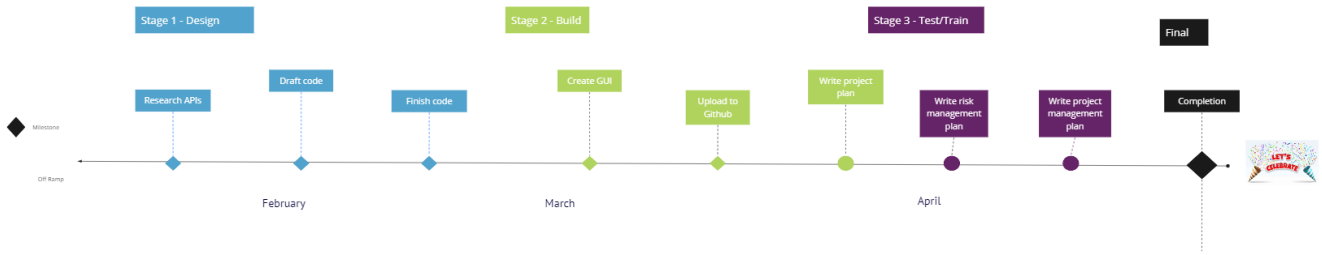
1. Writing the project plan
2. Drafting up the risk management log
3. Creating the Earned Value Sheet
4. Writing the project management plan
5. Writing the code for the data retrieval program
6. Creating the GUI for the program
7. Analysis of the data collected.
8. Uploading work to GitHub.

2. Milestones

Our team came together to reach various milestones to complete the project on time.
These milestones were:

Milestone #1	Research for technologies to use to collect data from APIs
Milestone #2	Select which APIs we will collect data from
Milestone #3	Draft up code to collect data
Milestone #4	Upload work to GitHub
Milestone #5	Finish draft to collect data from APIs
Milestone #6	Create GUI for application
Milestone #7	Write project plan
Milestone #8	Write project management plan
Milestone #9	Create risk management log
Milestone #10	Analyze data collected

3. Timelines



4. Roles

Role	**Person**
Project manager	Dipika
Developer	Rana, Dipika, Saikrishna
Tester and Analysis	Rana
Risk Management Spreadsheet	Dipika
Earned Value Spreadsheet	Saikrishna
Project management plan	Dipika

4. Responsibilities

1. Project management:

The project manager is tasked with leading the team, gathering information for the project, creating milestones for the team to follow, and giving suggestions to team members when they need it. A project manager is the nexus of project execution, responsible for planning, organizing, and overseeing tasks to meet objectives within timelines and budgets. They facilitate communication, manage risks, and foster team collaboration while maintaining morale and adapting to changing circumstances, ultimately steering the project towards success.

2. Developing Application:

The developer in a team is the engine driving the creation of an application, translating ideas into tangible digital solutions. With expertise in coding languages and frameworks, they craft the backbone of the software, ensuring functionality and efficiency. Collaborating closely with designers and project managers, they transform concepts into reality, continuously refining and iterating to meet user needs. Their ingenuity and technical prowess are instrumental in shaping the final product, making them indispensable contributors to the team's success.

3. Research:

The research specialist in a team digs deep into data, market trends, and user insights to provide valuable guidance. Collaborating with designers, developers, and project managers, they ensure decisions are grounded in solid research, driving the project forward with precision and clarity.

4. Data analysis:

The data analyst in a team sifts through data to extract valuable insights, aiding decision-making, and project direction. Collaborating with teammates, they provide essential information to drive strategies and refine products effectively.

5. Development Phases

The team went through many phases of development when creating the program which scrapes data from 4 different APIs for analysis purposes. Here are the main phases of development the team went through:

1. Research APIs and framework.
2. Decide on APIs to use.
3. Draft code
4. Finish code.
5. Create GUI
6. Upload code to GitHub.
7. Write project plan.
8. Write risk management plan.
9. Write project management plan.
10. Completion and submission

7. Testing

Testing the app is crucial to ensure its reliability, functionality, and security. By systematically evaluating its performance against various scenarios and conditions, we verify that the app meets user expectations and operates as intended. Testing helps us identify and address potential issues early in the development process, reducing the risk of bugs, crashes, or security vulnerabilities. Ultimately, thorough testing instills confidence in the app's quality, enhancing user satisfaction and trust.

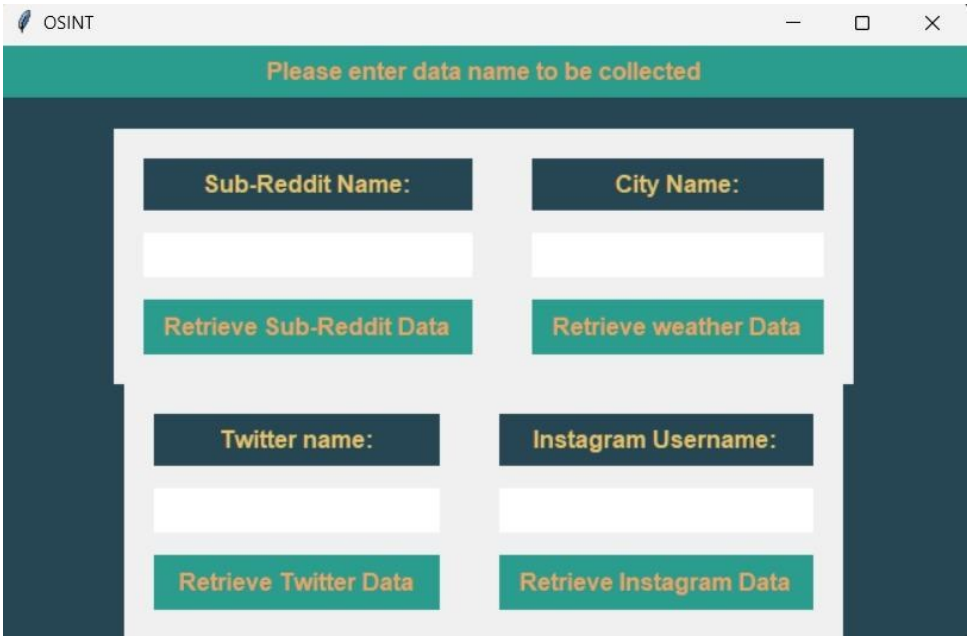
Test Name	**Description**
Input Validation Test	Check if the app properly handles various types of input, including unexpected or malicious data, to prevent injection attacks or crashes
API Response Validation	Verify that the app correctly parses and validates responses from each API, ensuring that it handles errors gracefully and securely.
Rate Limiting Test	Confirm that the app adheres to rate limits imposed by each API to prevent getting blocked or banned due to excessive requests.
Data Integrity Test	Ensure that the data retrieved from the APIs is accurately processed and displayed without any corruption or manipulation.
Authentication Test	Validate that the app securely handles authentication credentials for accessing

	the APIs, avoiding any potential leakage or unauthorized access.
SSL/TLS Test	Check if the app properly implements SSL/TLS encryption to protect data transmission between the client and the APIs, preventing interception or tampering.
Cross-Site Scripting (XSS) Test	Test for XSS vulnerabilities by injecting malicious scripts into input fields and verifying if the app sanitizes and escapes user-generated content properly.
Cross-Origin Resource Sharing (CORS) Test	Ensure that the app implements proper CORS policies to restrict access to APIs from unauthorized domains and prevent potential security risks.
Error Handling Test	Validate that the app handles errors and exceptions gracefully, providing informative error messages without exposing sensitive information.
Regression Testing	Perform regression testing after any updates or changes to ensure that existing functionality remains intact and no new security vulnerabilities are introduced.

8. Documentation

Files included in final product:

1. Project Plan
2. Risk management log.
3. EVM Workbook (Earned value sheet)
4. Project management plan
5. GUI/code
6. Data analysis
7. GitHub repo



Github Link

9. Python Application

This script is a comprehensive data collection tool using a graphical user interface (GUI) built with Tkinter. It interacts with several online platforms—Twitter, Instagram, Reddit, and a weather API—to fetch and save relevant data based on user input.

10. Features

- **Instagram Details:** Users can input an Instagram username to retrieve details such as username, profile Image, full name, follower count, bio, following count, total posts, private status.
- **Weather Report:** Users can input a city name to retrieve details about the weather of their city, this includes weather condition, temperature, humidity, wind speed and pressure.
- **Twitter Details:** Like Instagram, users can input a Twitter username to fetch details such as screen name, real name, follower count, following count, and total tweets.
- **Reddit Details:** Users can input a Reddit username to retrieve details such as username, post karma, comment karma, total posts, total comments, and the list of recent posts and comments.

Note: All the information above will be saved in their respective file based on the name they enter.

11. Requirements

- Python >=3.8.0

12. Setup

Install Dependencies:

- **pip install instaloader**
- **pip install snscrape**
- **pip install praw**

13. Running the Application:

Open the terminal where the file is placed and run the command.

- **python application.py**

14. Usage

Note: All the files for respective for details are being created where application.py is located.

1. Instagram Details:

Enter an Instagram username in the provided input field and click the "Retrieve" button.
View the retrieved user details and recent post details from the file saved.

2. Weather report:

Enter a city name in the provided input field and click the "Retrieve" button.
View the retrieved city weather report from the .txt file.

3. Twitter Details:

Enter a Twitter username in the provided input field and click the "Retrieve" button.
View the retrieved user details from the .csv file.

4. Reddit Details:

Enter a Reddit username in the provided input field and click the "Retrieve" button.
View the retrieved user details and recent post details.