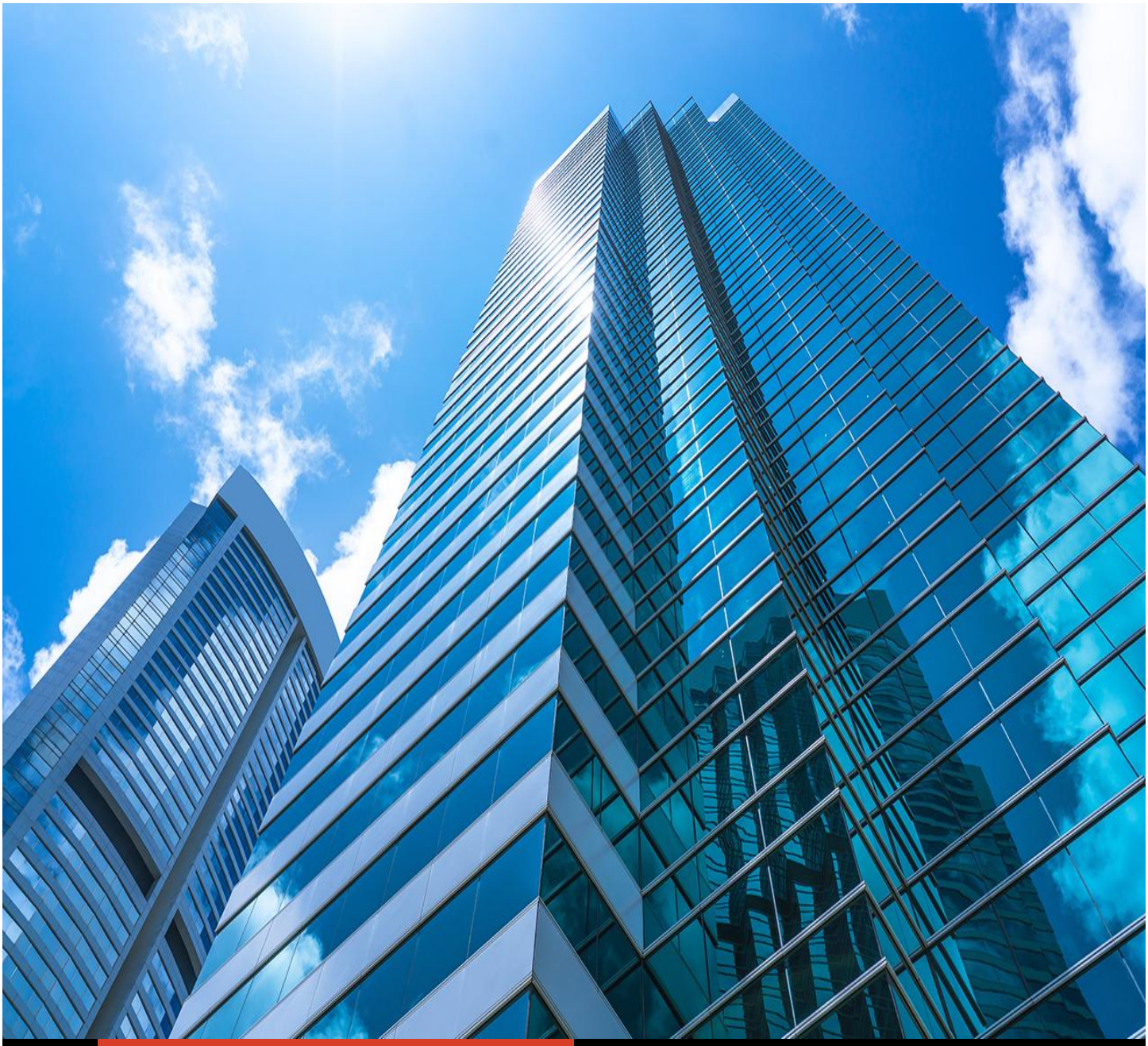

OPENSOURCE INTELLIGENCE (OSINT) APP – PROJECT PLAN

PROJECT LEAD(S): STEVE ANDRAWS, MATTHEW KALL, LUCAS JORDAN

PROJECT SPONSOR(S):
DR. MAURICE DAWSON JR.

I. Project Overview:

We set out to build a lightweight, opensource app that could pull public data from YouTube, Reddit, Wikipedia, and Media stack then show it all through a simple, usable Python GUI. It was not just a technical challenge. It was about making sense of scattered public information in one place and learning how to work like a dev team along the way.

II. Problem/Value Statement:

People struggle to make sense of huge volumes of online content. Our tool tackles that by pulling real time info from multiple sources, helping users get a fast and organized view of what is going on in the world without bouncing between tabs.

Goals, Scope & Measures of Success

Goal	How We'll Measure It
Modular API scripts working independently	Unit tests for each module and independent successful runs
Full integration with Tkinter GUI	GUI returns expected results and handles user input smoothly
Error handling and graceful failure implementation	App doesn't crash; users get meaningful error messages
Full documentation and enduser guide	README.md, PDF instructions, usage examples, submitted on time
Collaborative delivery using online tools	Regular checkins, updates on GitHub, task tracking in Planner

III. Stakeholders & Partnerships

1. Core Team: responsible for development, planning, and documentation.
2. PM: oversees deliverables, grading, and project criteria.
3. Users: future developers, students, researchers using the tool

IV. Deliverables & Milestones

Target Date	Deliverable / Milestone
April 4	Repo setup, API list finalized, roles assigned
April 9	API keys obtained, endpoints validated
April 14	YouTube and Reddit modules function independently
April 18	Mediastack and Wikipedia modules finished
April 20	GUI integration completed
April 23	Full testing of app and error handling
April 24	Documentation wrapped, reports finalized
April 25	Final submission and backup demo completed

Assumptions, Constraints & Dependencies

- We worked entirely remotely, so staying connected was essential.
- We assumed our APIs would not change dramatically during the project window, but we planned in case they did.
- We relied on stable internet, working APIs, and GitHub not going down at the worst possible time (thankfully, it did not).

Risk Mitigation

- API quota or key expiration
- Mitigation: Rotated keys regularly and cached data for fallback
- GUI crashes from bad input
- Mitigation: Exception handling and validation checks inside GUI input logic
- Missed deadlines or illness.
- Mitigation: Task reassignment via Microsoft Planner
- Tech issues during demo

V. Project Team & Roles

Name	Role	Main Contributions
Steve Andraws	Project Manager	Planning, documentation, team coordination, risk management
Matthew Kall	GUI/Backend Developer	Tkinter UI, YouTube & Wikipedia APIs, integration logic
Lucas Jordan	API & Testing Lead	Reddit & Mediastack APIs, exception handling, earned value sheet

Communication Plan:

We didn't want to guess who was doing what, so we stuck to a few solid tools that kept us organized and coordinated. MS planner and Teams.

1. Teams were our goal for weekly syncs and daily updates, it helped us catch issues early.
2. Planner gave us a visual of what was done, what was completed, and who owned what.
3. GitHub managed our code versions and README updates.
4. OneDrive was our cloud storage for everything from risk logs to drafts.

This management plan wasn't just paperwork, it's the reason we finished on time, with all deliverables intact and everyone still on speaking terms.

Thank you!

Team Members: Steve Andraws, Matthew Kall, Lucas Jordan