1. Project Name: Metro buzz app

* Description:

DC Life is a mobile application designed to help users discover and enjoy events in the Washington Metropolitan area. The app updates event schedules, locations, and other key details. Users can easily filter events by age group and event type, such as music, theater, festivals, family-friendliness, and other criteria, to find events that match their interests.

* Team:

1. Product Owner: Nicholas Kenner

2. Scrum Master: Gabriela Jimenez Martinez

3. Lead Designer: Vannesa Valentine

4. Lead Programmer: Javon Peterson

5. Tester: Luis Calderon

6. Tester: Biniam Lemma

7. Programmer: Heaven Frazier

8. Senior Programmer: Dagmawi Mulualem

9. Designer: Asia Omary

10. Senior Programmer: Hugh Smith

2. Introduction

* Brief background on the problem domain:

The problem we aim to address is the difficulty in finding DC-specific events using major platforms like Google or Eventbrite. These platforms often return events happening throughout the broader DMV area, which can be inconvenient or irrelevant for users who want to stay within Washington, DC. Our goal is to create a platform that prioritizes events happening only in DC, catering specifically to DC residents. Since DC is highly metro-accessible and walkable, we want to highlight events that are easily reachable — whether by foot, bike, or a short bus ride — and enhance the sense of community and accessibility for those who live in this beautiful city.

* Motivation behind the project:

We aim to develop an application uniquely and exclusively available for DC residents.

3. Problem Statement

* Who experiences this problem and why it matters

DC residents often struggle to find events that are specifically happening within the city. When using platforms like Google or Eventbrite, users are frequently shown events from the entire DMV area — including Maryland and Virginia — which can be overwhelming and not always relevant to their needs. This is especially frustrating for people who prefer attending local events. It creates a disconnect between residents and the vibrant community around them. We believe this matters because Washington, DC, is a unique and walkable city with a strong sense of local identity. Our goal is to design an application specifically for DC residents, helping them discover nearby events without the clutter of unrelated listings — ultimately making staying connected to their neighborhood and community easier.

4. Project Goals & Objectives

* List of primary goals (e.g., build an MVP, implement a specific algorithm)
* Course-aligned learning outcomes (if relevant)

5. Requirements & Specifications

* Functional requirements (e.g., "User can log in")
* Non-functional requirements (e.g., performance, scalability)
* Constraints or assumptions

6. System Design & Architecture

A screenshot of a computer

AI-generated content may be incorrect.

7. Development Environment

* Tools & platforms

VS Code and GitHub

* Languages:

Python, HTML, CSS

8. Implementation & Testing

* How you built each core module
* Testing strategy and test coverage
* Challenges encountered during implementation

9. How to Run the Project

* Setup instructions (cloning repo, installing dependencies)
* How to run the project locally
* How to run tests:

After downloading the proper files you run the code in the terminal and then open it within our browser.

10. Results / Evaluation

* What works well and what doesn’t
* User feedback or usability testing (if done)

11. Lessons Learned

* Key takeaways from the project

At the beginning of the project, one of our biggest challenges was coordinating team meetings outside of class. With different schedules and commitments, it was difficult for everyone to agree on a consistent meeting time. This made early collaboration and planning harder than expected. We also struggled initially with team communication and alignment, especially when it came to deciding when and how to meet. However, as the project progressed, we became more organized, learned to be flexible, and found tools and times that worked for everyone. This experience taught us the importance of clear communication, planning, and being adaptable when working in a team.

* Skills or concepts learned

Through this project, we gained a variety of technical and collaborative skills. On the technical side, the developers continued to sharpen their coding skills through the usage of Python, HTML, and CSS and became more comfortable using version control systems like GitHub. We developed stronger communication skills on the collaborative side, especially as a team where coordination was initially a challenge. We also learned how to break down large tasks, assign roles effectively, and support each other in both planning and development. Overall, this project gave us hands-on experience with real-world software development practices and teamwork.

12. Limitations & Future Work

* Known bugs, incomplete features
* What could be improved or extended in the future