

Project Title

Sprint Number

Date

Name	Email Address
Aaron Downing	aaron.downing652@topper.wku.edu
Ryerson Brower	ryerson.brower178@topper.wku.edu
Kaden Hunt	email kaden.hunt144@topper.wku.edu

CS 360

Fall 2025

Project Organization Documentation

Contents

1	Project Team’s Organizational Approach	1
2	Schedule Organization	1
2.1	Gantt Chart v1:	1
2.2	Gantt Chart v2:	1
2.3	Gantt Chart v3:	1
2.4	Final Gantt Chart:	1
3	Progress Visibility	1
3.1	Sprint 1 Progress Visibility	1
3.2	Sprint 2 Progress Visibility	1
3.3	Sprint 3 Progress Visibility	1
3.4	Sprint 4 Progress Visibility	1
4	Software Process Model	1
5	Risk Management	1
5.1	Risk Identification	1
5.2	Risk Planning	2
5.3	Risk Monitoring	2

List of Figures

1 Project Team's Organizational Approach

Sprint 1:
Sprint 2:
Sprint 3:
Sprint 4:

2 Schedule Organization

2.1 Gantt Chart v1:

Text goes here.

2.2 Gantt Chart v2:

Text goes here.

2.3 Gantt Chart v3:

Text goes here.

2.4 Final Gantt Chart:

Text goes here.

3 Progress Visibility

3.1 Sprint 1 Progress Visibility

Text goes here.

3.2 Sprint 2 Progress Visibility

Text goes here.

3.3 Sprint 3 Progress Visibility

Text goes here.

3.4 Sprint 4 Progress Visibility

Text goes here.

4 Software Process Model

Text goes here.

5 Risk Management

5.1 Risk Identification

The following risks are listed and categorized by potential risk level.

- **Database Performance (High Priority):** The database may become too slow under heavy loads, affecting responsiveness and user experience.

- **Route Generation API Failure (High Priority):** The external route generation API may not function as expected, preventing key functionality of the application.
- **Compatibility Issues (Medium Priority):** The system may encounter compatibility problems across different devices, operating systems, or software versions.
- **Access to Android Phones (Medium Priority):** Limited availability of Android devices for testing may hinder progress.
- **Large File Download (Low Priority):** Users may face issues when downloading or uploading large files, leading to performance bottlenecks.

5.2 Risk Planning

The team has outlined the following plans for risk planning:

- **Database Performance:** Implement query optimization, proper indexing, and caching strategies. Load testing will be performed early to identify bottlenecks.
- **Route Generation API Failure:** Identify and test backup APIs during early development.
- **Compatibility Issues:** Use emulators and virtual machines to ensure broader compatibility across various platforms.
- **Access to Android Phones:** Gain access to an android phone temporarily and share them among team members as needed.
- **Large File Download:** Implement file compression. Clearly communicate file size limitations to users.

5.3 Risk Monitoring

The following monitoring strategies will be used for the remaining duration of the project:

- **Database Performance:** Monitor query execution times and system performance.
- **Route Generation API Failure:** Implement automated alerts for API failures or unusual latency.
- **Compatibility Issues:** Maintain a compatibility checklist and update it with each system change.
- **Access to Android Phones:** Track device usage and availability within the team. .
- **Large File Download:** Monitor client feedback related to file transfers.