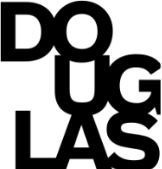
**Douglas College** 

**Module Code:**

CSIS 4495 – 071

**Module Title:**

Applied Research Project

**Year & Semester:**

2025 Fall

Project Proposal

**Alternative Industries: Weather Driver**

|  |  |
| --- | --- |
| **Student ID** | **Student Name** |
| 300392983 | Pukar Ojha |
| 300391004 | Chetan Kaur (Team Lead) |
| 300392663 | Pravesh Poudel |

**Instructor: Bambang, Sarif**

**Due Date: Sept 27th, 2025**

**Table of Contents**

[Introduction 1](#_Toc209886916)

[Problem Statement 1](#_Toc209886917)

[Literature Context and Gaps 2](#_Toc209886918)

[Project as a Solution 2](#_Toc209886919)

[Assumptions & Hypotheses 2](#_Toc209886920)

[Benefits 2](#_Toc209886921)

[Proposed Research Project 3](#_Toc209886922)

[Design and Objectives 3](#_Toc209886923)

[Aims 3](#_Toc209886924)

[Objectives 3](#_Toc209886925)

[Methodology 3](#_Toc209886926)

[Considered Methodologies 3](#_Toc209886927)

[Selected Methodology: Agile with Scrum 4](#_Toc209886928)

[Why Agile with Scrum 4](#_Toc209886929)

[Phases of Agile 4](#_Toc209886930)

[Initiate 4](#_Toc209886931)

[Plan and Estimate 5](#_Toc209886932)

[Implement 5](#_Toc209886933)

[Review and Retrospect 5](#_Toc209886934)

[Release 6](#_Toc209886935)

[Expected Outcomes and Deliverables 6](#_Toc209886936)

[Expected Outcomes 6](#_Toc209886937)

[Key Deliverables 6](#_Toc209886938)

[Resources Required 7](#_Toc209886939)

[Human Resources 7](#_Toc209886940)

[Technologies And Tools 7](#_Toc209886941)

[Other Resources 7](#_Toc209886942)

[Riipen External Partner 8](#_Toc209886943)

[Project planning and Timeline 9](#_Toc209886944)

[Roles & Responsibilities 9](#_Toc209886945)

[Work Breakdown Structure (WBS) 9](#_Toc209886946)

[Milestones 10](#_Toc209886947)

[Gantt Charts 10](#_Toc209886948)

[Project Contract 12](#_Toc209886949)

[Overview 12](#_Toc209886950)

[Communication & Meetings 12](#_Toc209886951)

[Task Management & Collaboration 12](#_Toc209886952)

[Conflict Resolution 12](#_Toc209886953)

[Work Logs 13](#_Toc209886954)

[Pukar Ojha 13](#_Toc209886955)

[Pravesh Poudel 16](#_Toc209886956)

[Chetan Pal Kaur 18](#_Toc209886957)

[Conclusion 21](#_Toc209886958)

[References 22](#_Toc209886959)

**Table of Figures**

[Figure 1: Work Breakdown Structure 9](#_Toc209886960)

[Figure 2: Milestone Chart 10](#_Toc209886961)

[Figure 3: Gantt Chart 11](#_Toc209886962)

**Table of Tables**

[Table 1: Work log of Pukar Ojha 15](#_Toc209886963)

[Table 2: Work log of Pravesh Poudel 18](#_Toc209886964)

[Table 3: Work logs of Chetan Pal Kaur 20](#_Toc209886965)

# Introduction

Driving is not just about getting from one place to another it is also about staying safe and being prepared for unexpected conditions on the road. Severe weather, limited fuel stops, and sudden hazards can all create risks for drivers if they are not warned in time. While navigation apps like Google Maps provide directions and traffic updates, they rarely integrate real time weather alerts or extra driver-focused tools in a single platform.

The Weather Driver mobile app is designed to close this gap. It combines navigation with live hazardous weather notifications, giving users an added layer of safety while traveling. Beyond navigation and weather, the app also includes practical features such as gas station locators, a built-in speedometer, toll pass tracking, customizable notifications, and user-friendly settings. These additions make the app not just a route planner, but a complete driver’s companion.

Studies on transportation and road safety apps show that it’s very important for drivers to get live updates about road hazards. The problem is that many existing solutions only work with expensive car systems or are not easy for everyday drivers to use. Our project is based on the idea that a mobile app can make these features more accessible, helping drivers stay more aware, confident, and prepared while traveling. The benefit of this work is twofold: it can make driving safer by reducing risks from sudden weather changes, and it can also make the overall driving experience easier by bringing several useful tools together in one app.

## Problem Statement

Road accidents and travel disruptions due to severe weather conditions are a growing concern. According to the U.S. Department of Transportation, nearly 21% of vehicle crashes annually are weather-related, leading to thousands of deaths and injuries each year. Current navigation apps like Google Maps offer route planning but lack integrated real-time hazardous weather alerts. (U.S. Department of Transportation, Federal Highway Administration, 2025)

## Literature Context and Gaps

Research on transportation and road safety apps highlights the importance of live hazard updates for drivers. However, existing solutions are either tied to expensive in-car systems or fail to provide a seamless mobile experience. This project addresses that gap by making safety-focused features more accessible via a mobile app.

## Project as a Solution

The weather driver mobile app is designed to make driving safer by combining Google Maps navigation with real-time weather hazard alerts. The app not only provides directions through Google Maps but also delivers timely warnings about severe weather conditions and potential hazards. By receiving these alerts, users can make safer route choices during their trips. In addition, the app will be hosted on AWS to ensure it remains scalable, reliable, and accessible anytime.

## Assumptions & Hypotheses

Several assumptions and hypotheses have been considered while working on the Weather Driver project. We assume that mobile app users will benefit from receiving real-time weather hazard alerts integrated with navigation. By combining Google Maps with weather APIs such as the National Oceanic and Atmospheric Administration (NOAA), the app is expected to reduce travel risks and support safer decision-making. Furthermore, hosting the cross-platform mobile application on AWS will ensure scalable, reliable, and consistent access to these services.

## Benefits

This solution will improve driver safety, increase weather awareness, and provide a foundation for future growth through ad revenue and feature expansion.

# Proposed Research Project

## Design and Objectives

Our project is designed as applied research, which means we are building something practical that will be tested in the real world. The main goals are to create a mobile app that combines navigation with live weather hazard alerts, expand it with useful features like gas stations, toll passes, a speedometer, settings, and notifications, and ensure it runs smoothly and reliably.

### Aims

To develop and deploy a mobile app that enhances road safety by integrating navigation with real-time weather warnings.

### Objectives

A project should have clear objectives going forward, similarly we discussed objectives with the client and described them as below:

* Implement frontend in React Native using existing Figma designs.
* Integrate Google Maps API for directions and hazard overlays.
* Add weather API (NOAA) integration for real-time alerts.
* Connect to existing backend and database.
* Deploy on AWS for scalability and availability.
* Deliver a working app with technical and user documentation.

## Methodology

### Considered Methodologies

Before initiating the project, a software methodology is important to consider for successful completion of the project. In this project, we have considered several methodologies, and they are waterfall methodologies which have clear structure but inflexible for mid-project changes and second is agile methodologies which is Iterative, ideal for student projects.

### Selected Methodology: Agile with Scrum

We chose the Agile methodology because it focuses on flexibility, teamwork, and delivering results step by step. To implement Agile, we will follow the Scrum framework, which organizes our work into sprints. In each sprint, we will plan tasks, develop features, test them, and then review what we achieved. Scrum also emphasizes roles and ceremonies:

**Scrum Roles:** Our team will act as the development team, with one person taking on the role of Scrum Master to keep meetings and progress on track, another as Product Owner responsible for prioritizing features from the backlog and the development team.

**Scrum Ceremonies:** We will have short weekly standups to share progress, sprint planning meetings to assign tasks, sprint reviews to demo completed work, and retrospectives to discuss improvements.

### Why Agile with Scrum

Scrum is well-suited for our project because it allows us to focus on one feature at a time (like maps first, then weather, then gas stations, and so on), while still delivering working parts of the app regularly. It also helps us stay flexible if we face technical issues with APIs or integrations, since we can adjust priorities for the next sprint. With three team members, Scrum’s structure makes collaboration clearer, since each of us will have assigned tasks but still work closely together. (scrumstudy.com, 2025)

## Phases of Agile

Our Scrum-based Agile process will still follow the main SDLC phases, but in an iterative loop:

### Initiate

In this phase, we will set the foundation of the project. Our project vision is to build a driver-focused mobile app that combines navigation with real-time weather hazard alerts and other helpful tools. Within our team, one member will take the role of Product Owner to manage the backlog and priorities, another will be the Scrum Master to facilitate meetings, and all three of us will work as the development team. We will start by defining the main features, such as navigation, weather alerts, gas stations, toll passes, notifications, and a speedometer, then organize them into a prioritized backlog and plan the release to be completed within our 10-week timeline.

### Plan and Estimate

In this phase, we will define the main Epics for our project, such as navigation, weather alerts, gas stations, toll passes, notifications, and a speedometer. These Epics represent the big features of the app and will guide how we plan and organize our sprints.

### Implement

This is where development happens. Each sprint (2 weeks) will focus on completing a set of features, along with unit testing to ensure they work correctly:

**Sprint 0:** Team Formation, Defining the project Scope, goals, and objectives, defining team roles, setting up the tools like JIRA, and working on the Project Proposal.

**Sprint 1:** Set up the environment, repository, and AWS, implement navigation with Google Maps, analyze the existing system architecture, and adjust the screen color (as per the Figma screens).

**Sprint 2:** Integrating the weather API and hazard alerts, performing unit tests for API responses and alert displays, and creating the user profile.

**Sprint 3:** Add features such as gas station locator, toll passes tracking, with unit tests for each new feature.

**Sprint 4:** Developing the notifications and settings screens, integrating them with the backend, and performing unit tests to ensure notification triggers and user preferences work correctly.

**Sprint 5:** Carry out final user acceptance testing, fix bugs, complete final deployment, and prepare documentation and product handover to the client.

We will hold weekly standups to share progress and refine the backlog. (scrumstudy.com, 2025)

### Review and Retrospect

The project is divided into five sprints, and each sprint will include both a review and a retrospective. At the end of every sprint, the team will conduct a Sprint Review to demonstrate the completed features and gather feedback from teammates, peers, and the instructor. This will be followed by a Sprint Retrospective, where the team will reflect on what went well, identify any challenges faced, and discuss strategies for improvement in the upcoming sprint.

### Release

While the sprints focus on development, a project is not complete until it is properly delivered and released to the client. Before the handover, tasks such as hosting the project on AWS, building test cases, and preparing documentation will be carried out to ensure a successful release. As part of the release phase, the team will deploy the working app on AWS and generate a test build for users. The final delivery will include key features such as navigation, weather alerts, settings, notifications, and gas stations. In addition, both user documentation and technical documentation will be prepared to support future use and maintenance. Finally, the team will reflect on the lessons learned throughout the project, using these insights to guide future improvements or expansions of the app.

## Expected Outcomes and Deliverables

### Expected Outcomes

Before starting a project, a team should have clear understanding of its aims and objectives also the expected outcomes, which are directed towards the successful project and our expected outcomes are as: a mobile app that alerts drivers about weather hazards, seamless integration of mapping and real-time weather data and a scalable and deployable solution hosted on AWS.

### Key Deliverables

In discussions with the client, the key deliverables for the project were clearly outlined. These include the implementation of the front end using React Native, incorporating core features such as navigation, location search, and hazard or weather warnings. The app will integrate Google Maps with weather APIs to provide real-time information for safer travel. Additionally, deployment scripts and setup will be prepared for AWS hosting to ensure scalability and reliability. The final outcome will be a fully functional mobile application, delivered as an APK or test build. Alongside the application, comprehensive technical documentation, including setup guides and API references, will be provided, as well as user documentation to assist with app usage.

## Resources Required

### Human Resources

A successful project requires adequate manpower to ensure timely completion. For this project, the team is composed of various roles, including React Native Developers, a Project Mentor, a UI/UX Designer, and a Backend Developer, each contributing their expertise to achieve the project’s objectives.

### Technologies And Tools

The key part of building a software/ app is understanding the technologies and tools to be used for the development. Here, we discussed the tools we’ll be using for the project.

* React Native
* Expo, GitHub, Postman
* Google Maps API, Weather APIs
* Node.js/Express (backend)
* MongoDB/MySQL (DB)
* AWS (EC2, S3, SQS)

### Other Resources

The development process involves more than just writing code, it also requires proper documentation, design resources, a suitable development environment, and mentorship. The team will rely on existing UI/UX documentation to guide design decisions, while development environments will include both local machines and AWS accounts for building and testing the application. In addition, weekly consultations with the project mentor will provide valuable guidance and support throughout the development cycle.

# Riipen External Partner

* **Project Host:** Alternative Industries (via Riipen)
* **Contact/Founder:** Juliet (project sponsor)
* **Riipen Link:** [Weather Driver – Douglas College Riipen](https://douglascollege.riipen.com/projects/KVGwXgVM)

# Project planning and Timeline

## Roles & Responsibilities

The team consists of three members, and their roles and responsibilities are listed below.

* **Chetan Kaur (Team Lead):** Coordination, frontend, documentation, QA, Scrum Master.
* **Pravesh Poudel:** Frontend, Ui Design, and Backend and DB assistance, Product Owner.
* **Pukar Ojha:** API integration, testing, backend integration, and Database implementation.

## Work Breakdown Structure (WBS)

A Work Breakdown Structure is a method of breaking down the overall project into smaller, more manageable parts. It organizes the project into tasks and sub-tasks, making it easier to assign responsibilities, estimate timelines, and track progress.

**A screenshot of a computer

AI-generated content may be incorrect.**

Figure 1: Work Breakdown Structure

## Milestones

Milestone can be simply known as important events. A chart that represents important events, planned in a timeline, is known as a milestone chart. The milestone chart of the wager app is constructed to mark important events with proper timelines. Many tasks set proper timeline for timely completion of the project and ease the flow of the project and development methodology.

A diagram of a company

AI-generated content may be incorrect.

Figure 2: Milestone Chart

## Gantt Charts

A Gantt chart is a project management tool that shows tasks along with their start and end dates in a timeline format. It helps the team track progress, manage deadlines, and keep the project on schedule.

In this project, the Gantt chart displays the full timeline, including each sprint, its main tasks, and duration. The work begins with Sprint 0 (Initiation) from September 3 to September 17, 2025, where the team sets up the scope, objectives, roles, and tools. After that, development follows the Agile Scrum approach with 2-week sprints.

The sprint -1 starts on September 18, 2025, and the process continues until Sprint 5, which ends on November 26, 2025. Each sprint runs for 14 days, focusing on specific tasks such as UI design, API integration, testing, and deployment. This planned approach makes it easier to track work, meet deadlines, and stay aligned with project goals and stakeholder expectations.

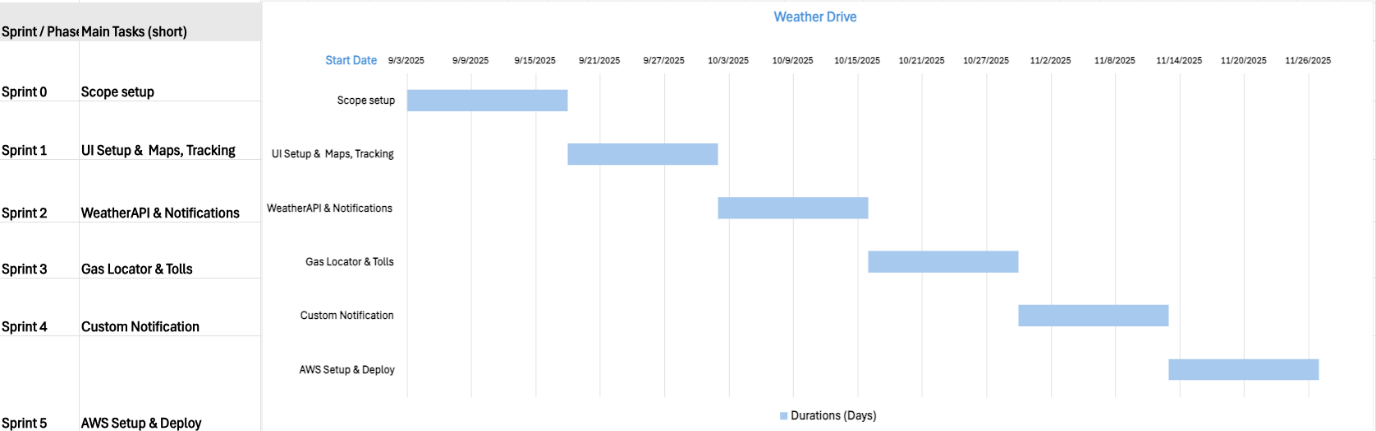


Figure 3: Gantt Chart

# Project Contract

## Overview

This contract outlines team collaboration guidelines, ensuring equal contributions, professional communication, and task accountability. Members are expected to meet deadlines and provide progress updates.

## Communication & Meetings

* **Meeting Frequency:** Team meetings will be held at least twice a week, either in-person or virtually via Google Meet or Zoom.
* **Communication Tools:** Zoom for scheduled meetings, Riiipen & WhatsApp group for quick discussions.
* **Response Time:** Team members should respond to messages within 10 hours to ensure smooth workflow.

## Task Management & Collaboration

* **Collaboration**: Team members are expected to collaborate effectively using Riipen for collaboration & communication, Microsoft Word Online for documentation and GitHub as a Repository.
* **Deadlines:** Each task will have a clear deadline, and members are expected to update the team on their progress.
* **Accountability:** If a member is unable to complete a task, they must notify the team in advance to discuss reassignments or solutions.

## Conflict Resolution

Disagreements will be discussed openly within the team to find a mutually beneficial resolution. If a resolution cannot be reached, the issue will be escalated to the course instructor.

**Pukar Ojha**: Pukar

**Pravesh Poudel**: Pravesh

**Chetan Pal Kaur**: Chetan

# Work Logs

## Pukar Ojha

|  |  |  |
| --- | --- | --- |
| **Date** | **Number of hours** | **Description of work done** |
| Sept 3, 2025 | 1 | Participated in group meetings/discussions about exploring and choosing the project for the course. |
| Sept 6, 2025 | 0.5 | Met with Professor about the choice of project, exploring the ideas, asking for suggestions and advice, and working on the project. |
| Sept 9, 2025 | 0.5 | Created a Riipen account with the professor’s invitation, explored the app's functions and features, tried connecting with the project manager, as well as scheduled an interview with the Project Manager for Sept 10, from 1:30-2:30 pm. |
| Sept 10, 2025 | 1 | Participated in an online Zoom meeting with the Project Manager, trying to understand the project, requirements, and deliverables. |
| Sept 13, 2025 | 0.5 | Explained to the professor about the progress and the meeting outcomes with the project manager. Discussed the proposal submission, deliverables, and expected outlines. |
| Sept 13, 2025 | 3 | Worked on proposal drafting, researched the project Weather Driver app, looked after the UI provided, and extracted the functions and functionalities to be considered.  Drafted the project contract and signed it between group members going forward.  Explored, learned, and discussed the SDLC to be used and considered going forward with the project. |
| Sept 16, 2025 | 3.5 | Attended a Zoom meeting with the Project Manager, discussing project deliverables and GitHub collaboration.  Worked on the Project Proposal, Milestone chart, and review of the document. |
| Sept 20, 2025 | 5 | Attended the meeting with the professor, understanding the proposal’s outcome and expectations.  Attended a meeting with the client and discussed the technologies to be used for the project.  Worked on reviewing and updating the project proposal along with all the diagrams and their descriptions. |
| Sept 23, 2025 | 4 | Exported the provided Figma screen to the project.  Worked on creating work logs and maintaining the structure. |
| Sept 24, 2025 | 1 | Attended a meeting with the client and discussed resolving the requirement issues(as the provided code platform/ language differs from their requirement documents)  Learned about their code flow, architecture with UI screens validation, and connection with backend and its integration. |
| Sept 26, 2025 | 1 | Edited and analyzed the code workflow and oversaw the group members’ GitHub code, and resolved the issues. |
| Sept 27, 2025 | 5 | Worked on fixing the commit conflict as the group member’s commit conflicted with the package version, which disturbed the whole repo in the GitHub repo, and worked on fixing packages and code conflicts using Stack Overflow and the latest update.  Reviewed and updated the proposal-resubmission. |

Table 1: Work log of Pukar Ojha

## Pravesh Poudel

|  |  |  |
| --- | --- | --- |
| **Date** | **Number of hours** | **Description of work done** |
| Sept 3, 2025 | 1 | Joined group discussions to explore ideas and decide on the project for the course. |
| Sept 6, 2025 | 0.5 | Met with the professor to talk about possible project options, share ideas, and get advice. |
| Sept 9, 2025 | 0.5 | Set up a Riipen account with the professor’s invitation, explored its features, and connected with the project manager. |
| Sept 10, 2025 | 1 | Attended a Zoom kick-off meeting with the Project Manager to understand the project requirements and deliverables. |
| Sept 13, 2025 | 0.5 | Updated the professor on our progress and shared details from the meeting with the Project Manager. Also talked about proposal submission, deliverables, and expected outlines. |
| Sept 13, 2025 | 3 | Worked on drafting the proposal and researching the Weather Driver app. Reviewed the Figma screens, studied the UI, and listed important functions and features. Discussed which SDLC model to follow. |
| Sept 16, 2025 | 4 | Attended a Zoom meeting with the Project Manager to review deliverables and GitHub collaboration. I went through the document shared by the project owner and worked on the proposal, creating and updating the WBS. |
| Sept 18, 2025 | 4 | Analyzed the handout GitHub repo to understand the workflow and the operation of the code. |
| Sept 19, 2025 | 3 | Created a GitHub repo, to transfer handout frontend and backend code.  Merged and sync the code to the same repo and helped members setup on their devices. |
| Sept 20, 2025 | 2 | Met with professor to understand how to make better proposals, understanding key ideas, and space for improvement.  Conveyed the difference in client requirement document and actual code.  Scheduled the meeting with the client to discuss the differences in their code and requirements to eliminate the confusion and misunderstanding. |
| Sept 23, 2025 | 2 | Revisited the proposal and made some formatting changes, changed the Milestone diagram, and put Paragraphs instead of Bullet points. |
| Sept 24, 2025 | 1.5 | Conducted a meeting with the client’s technical team and discussed the project backend architecture, UI Screens, and Flow of the frontend. |
| Sept 25, 2025 | 1.5 | Started working on the Frontend UI and Screens based on the Figma. Designed splash screen, Login, and signup screens. |
| Sept 26, 2025 | 3 | Work on the Forget Password, Reset Password, and Map welcome screen with profile icon. |

Table 2: Work log of Pravesh Poudel

## Chetan Pal Kaur

|  |  |  |
| --- | --- | --- |
| **Date** | **Number of hours** | **Description of work done** |
| Sept 3, 2025 | 1 | Took part in group meetings and discussions to explore and select the project for the course. |
| Sept 6, 2025 | 0.5 | I had a meeting with the professor to review potential project choices, discuss ideas, and receive guidance and suggestions for the project. |
| Sept 9, 2025 | 0.5 | Created a Riipen account using the professor’s invitation, explored its functions and features, and connected with the project manager. |
| Sept 10, 2025 | 1 | Attended an online Zoom kick-off meeting with the Project Manager to discuss and gain a clear understanding of the project requirements and deliverables. |
| Sept 13, 2025 | 0.5 | Updated the professor on project progress and shared the outcomes of the meeting with the Project Manager. Also discussed the proposal submission, deliverables, and expected outlines. |
| Sept 13, 2025 | 3 | Worked on drafting the proposal and researching the Weather Driver app. Reviewed the provided Figma screens, analyzed the UI, and identified key functions and features to be included. Focused on Agile methodology and selected the Scrum framework. Also explored, learned, and discussed the SDLC to be followed as we move forward with the project. |
| Sept 16, 2025 | 4 | Attended a Zoom meeting with the Project Manager to discuss project deliverables and GitHub collaboration. Worked on the project proposal, created and updated Gantt charts, updated the proposal document, set up Jira, and worked on Agile Scrum methodology, including sprint planning. |
| Sept 19, 2025 | 4 | Cloned and set up the repositories on the local system. Configured the project environment, set up the front-end, and used Expo to successfully run the initial front-end build |
| Sept 20, 2025 | 3 | Explored backend technologies that were new to us. Scheduled a meeting with the tech team to clarify, but they were unavailable. Encountered issues running Expo due to errors, which blocked further progress. |
| Sept 22, 2025 | 2 | Worked on updating the project proposal, fixing formatting issues, and preparing the final reviewed version. |
| Sept 24, 2025 | 2 | Connected with the tech team to clarify backend and Figma questions. Learned that front-end code is in TypeScript, but they approved using JavaScript. They agreed to provide AWS keys for the backend connection. Start with front-end screens, that is, login screens. |
| Sept 26, 2025 | 2 | Reviewed the shared backend setup video and studied the process for integrating the system. |
| Sept 27, 2025 | 3 | Worked on “Progress Report-1”, compiling work logs, descriptions, and repo updates into the report document. |

Table 3: Work logs of Chetan Pal Kaur

# Conclusion

The WEATHER DRIVER project is a practical initiative that combines mobile app development with real-world applications in navigation and safety. By integrating Google Maps API, NOAA, and weather alerts and deploying on AWS, the app will provide a valuable tool for safer travel. The project also allows the team to gain essential technical and collaborative skills aligned with industry standards.

# References

scrumstudy.com. (2025, September 15). *Scrum Phases and Processes*. Retrieved from www.scrumstudy.com: https://www.scrumstudy.com/whyscrum/scrum-phases-and-processes

U.S. Department of Transportation, Federal Highway Administration. (2025, July 15). *How Do Weather Events Affect Roads*. Retrieved from ops.fhwa.dot.gov: https://ops.fhwa.dot.gov/weather/roadimpact.htm#:~:text=On%20average%2C%20over%203%2C800%20people,during%20rain%20or%20mist%20conditions.