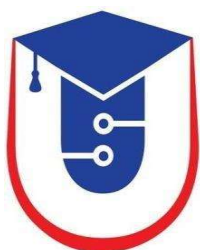


Final Project Proposal of Bachelor in Computer Engineering

**AI-Enhanced: Smart Community Complaint App**



ESTD: 2065

**UNITED  
TECHNICAL  
COLLEGE**

Affiliated to Pokhara University

**Submitted by:**

<b>Aashutosh Sapkota</b>	<b>BEC [2077-01]</b>
<b>Abhishek Sharma</b>	<b>BEC [2077-02]</b>
<b>Hom Bdr. Pathak Kshetri</b>	<b>BEC [2077-16]</b>
<b>Rajeev Paudel</b>	<b>BEC [2077-24]</b>

**United Technical College**  
**Faculty of Science and Technology**  
**Pokhara University, Nepal**

January 24, 2025

## **Abstract**

The Smart Community Complaint App is an innovative mobile application that empowers citizens to report various community issues, including roads, traffic, drinking water, drainage, electricity, waste management, animal concerns, and more. Users can easily submit complaints using geolocation services, upload images for better context, and receive real-time updates through push notifications. The integration of AI allows for automated analysis of reported images, streamlining the issue identification and prioritization process. Additionally, the offline feature ensures that users can report problems even in areas with limited internet connectivity. This system enhances public engagement and facilitates direct communication between citizens and local authorities. The application will be built using Flutter, Dart, Firebase, and Python ensuring a user-friendly interface and efficient data management. The project is expected to take 10 weeks.

**KEYWORDS:** *Problem, Complaint, Acknowledgment, Authorized, Android, Ios, Smart Community Complaint, Artificial Intelligence, etc.*

# TABLE OF CONTENTS

Abstract .....	ii
List of Figures .....	v
List of Table .....	vi
ACRONYMS/ABBREVIATIONS .....	vii
Chapter 1: INTRODUCTION.....	1
1.1. Background.....	1
1.2. Problem statement.....	2
1.3. Objectives .....	3
1.4. Motivation and Significance .....	3
1.5. Scope of the Work.....	4
1.6. Limitation.....	5
1.7. Application.....	5
Chapter 2: Literature Review .....	6
2.1. Case study .....	6
2.1.1. Civic Social Complaint Android App .....	6
2.1.2. E-Complaint Mobile Application.....	7
2.1.3. Nagarik App .....	8
Chapter 3: Methodology .....	10
3.1. Development and Planning.....	10
3.1.1. Concept and Initiation.....	10
3.1.2. Definition and planning .....	11
3.1.3. Software Used.....	11
3.1.4. Development.....	12
3.1.5. Launch and Execution.....	14

3.1.6. Performance and Control .....	14
3.2. System Design .....	14
3.2.1. System Flow Diagram.....	15
3.2.2. E-R diagram .....	16
3.2.3. DFD (Data Flow Diagram) .....	17
3.2.4. Use case Diagram .....	19
3.2.5. Software Development Model .....	20
3.3. Software and hardware requirements.....	21
3.4. Testing and Maintenance .....	22
Chapter 4: Expected Outcome .....	23
REFERENCES .....	24
APPENDICES .....	26

## **List of Figures**

Figure 3.1 System Flow Diagram .....	15
Figure 3.2: ER Diagram.....	16
Figure 3.3: Level 0 Data Flow Diagram .....	17
Figure 3.4: Level 1 Data Flow Diagram .....	18
Figure 3.5: Level 2 DFD for Login Process .....	18
Figure 3.6: Level 2 DFD for Complaint Registration.....	19
Figure 3.7: Level 2 DFD for Update Status.....	19
Figure 3.8 Use case Diagram .....	20
Figure 3.9: Agile Development Model .....	21

## **List of Table**

Table 1: Gantt chart of the project .....	26
---	----

## **ACRONYMS/ABBREVIATIONS**

DFD	:	Data Flow Diagram
AI	:	Artificial Intelligence
ERD	:	Entity Relationship Diagram
GB	:	Giga-Bytes
GDP	:	Gross Domestic Product
HTTP	:	Hypertext Transfer Protocol
MB	:	Mega-Bytes
ML	:	Machine Learning
RAM	:	Random Access Memory
RDBMS	:	Relational Database Maintenance System
SDK	:	Software Development Kit
VS Code	:	Visual Studio Code

# **Chapter 1: INTRODUCTION**

## **1.1. Background**

Nepal's road infrastructure faces severe challenges, with 77% of national highways and 82% of feeder roads in poor condition, as per the Economic Survey of 2018/19. This deterioration increases vehicle operating costs, journey times, and fuel consumption. Moreover, delays in major projects like the Postal Highway Project, where 78% of work has been delayed due to design mistakes and weak contract management, highlight systemic inefficiencies. Such issues emphasize the need for a streamlined reporting mechanism to address road-related problems effectively (Maharjan, 2019).

Urban areas in Nepal, particularly the Kathmandu Valley with over 4 million residents, suffer from severe traffic congestion. Insufficient public transport and uncoordinated planning among federal, provincial, and municipal agencies exacerbate the problem. Roads are often dug up multiple times a year for maintenance by different agencies without coordination, worsening traffic conditions. Addressing these issues requires citizen engagement to report traffic bottlenecks and advocate for better management (Thapa, 2024).

Access to clean drinking water remains a critical issue. In Kathmandu Valley alone, the daily water demand is 400 million liters, but supply falls far short of this requirement. Projects like the Melamchi Drinking Water Project have faced prolonged delays due to bureaucratic hurdles and poor governance. Citizens frequently face water shortages and poor-quality supply, necessitating a platform to report such grievances (Purbey, 2023).

Poor drainage systems in Nepal lead to frequent flooding during heavy rains, particularly in urban areas like Kathmandu, where blocked or inadequate drainage causes regular waterlogging and health risks. Additionally, the country experiences frequent power outages due to aging infrastructure and natural disasters, with



landslides damaging electricity towers and prolonging outages in rural regions. Waste management is another significant challenge, with delays in garbage collection and illegal dumping disrupting sanitation efforts, especially in Bharatpur due to landfill disputes. Stray animals also pose safety concerns for pedestrians and can lead to accidents on the roads. Moreover, various other community issues, such as broken streetlights and environmental hazards, often go unreported due to the lack of accessible platforms (Thapa, 2024).

To tackle the diverse challenges faced by communities, the Smart Community Complaint App provides a platform for citizens to report issues related to roads, traffic, drinking water, drainage, electricity, waste management, and animal concerns. This application empowers users to actively participate in improving their surroundings by directly communicating with relevant authorities (Purbey, 2023).

The main objective of the Smart Community Complaint App is to foster transparent collaboration between citizens and government organizations. By leveraging citizen engagement and advanced technologies like artificial intelligence for image recognition, the system streamlines the reporting and resolution of problems such as potholes, traffic congestion, water shortages, and power outages (Thapa, 2024).

This app offers a promising solution to enhance community infrastructure and services. Promoting active participation and efficient collaboration has the potential to improve living conditions significantly, leading to safer roads and better overall community well-being.

## **1.2. Problem statement**

Nepal faces critical infrastructure challenges across roads, traffic, drinking water, drainage, electricity, and waste management. Poor road maintenance, traffic congestion in urban areas like Kathmandu Valley, inadequate drainage causing monsoon flooding, unreliable electricity supply, and delays in waste collection due to landfill disputes are common issues. Additionally, access to clean drinking water

remains insufficient, and stray animals on roads pose safety risks (Lamichhane, 2024).

A major issue is the lack of a platform for citizens to report these problems, leaving concerns like potholes, drainage blockages, and power outages unaddressed. This gap undermines public trust and perpetuates inefficiencies. This lack of responsiveness undermines public trust in local governance (Adhikari, 2024).

The Smart Community Complaint App aims to address these challenges by providing a user-friendly platform for citizens to report issues in real-time across various categories—roads, traffic, drinking water, drainage, electricity, waste management, animal concerns, and more. By leveraging technology for efficient complaint resolution processes, the app seeks to improve infrastructure quality and enhance public confidence in governance (Acharya, 2023).

### **1.3. Objectives**

The objectives of the proposed system are as follows:

1. To develop a mobile application using Flutter for reporting various community issues, including roads, traffic, drinking water, drainage, electricity, waste management, and animal concerns.
2. To integrate features like artificial intelligence for image recognition to streamline issue identification and prioritization, along with an offline submission option for areas with limited internet connectivity.
3. To empower citizens with a platform that encourages active participation in community improvement and fosters collaboration with local authorities.

### **1.4. Motivation and Significance**

Nepal's existing municipal complaint systems often lack real-time tracking and efficient response mechanisms. Citizens frequently face challenges reporting issues through traditional channels like phone calls or physical visits, leading to delayed responses and inadequate resolutions. Existing mechanisms for lodging grievances

are inefficient, time-consuming, and often lack transparency. Many citizens are unaware of how to contact the concerned authorities, and the process often involves bureaucratic delays. The rapid development of mobile technology and AI provides an opportunity to bridge this gap by creating a platform that simplifies issue reporting and ensures prompt action. The motivation to choose this topic stems from the need to empower citizens, enhance government accountability, and foster smarter urban management.

### **Significance:**

#### **1. Innovation**

- AI-powered categorization and prioritization of complaints
- Real-time location tracking and photo evidence capabilities
- Automated routing to relevant government departments
- Smart analytics for identifying recurring issues and patterns

#### **2. Practical Impact**

- Improves transparency in complaint handling
- Enables data-driven decision-making for urban planning
- Streamlines communication between citizens and authorities

#### **3. Technical Contribution**

- Integration of Flutter for cross-platform compatibility
- Implementation of ML models for complaint classification
- Real-time database management for complaint tracking

### **1.5. Scope of the Work**

The proposed system will enable citizens in Nepal to report various civic issues through an AI-driven mobile application, covering the following areas:

1. Road Maintenance: Report potholes, damaged roads, and obstructions.

2. **Waste Management:** Notify authorities about improper garbage disposal and collection issues.
3. **Accident Reporting:** Alert emergency services about road accidents with details and GPS.
4. **Traffic Issues:** Report congestion, accidents, or roadblocks affecting traffic flow.
5. **Drinking Water Issues:** Report contaminated water sources or lack of supply.
6. **Drainage Problems:** Notify about blocked drains leading to flooding or sanitation issues.
7. **Electricity Issues:** Report power outages or faulty streetlights.
8. **Animal Issues:** Report stray animals causing safety concerns.
9. **Other Community Grievances:** General reporting for various local issues impacting public welfare.

## **1.6. Limitation**

1. **Data Reliability:** The accuracy and reliability of user-generated data may vary, requiring the implementation of validation and verification mechanisms to ensure credible reporting.
2. **Connectivity and Access:** Effective use of the system depends on a stable internet connection and access to smartphones or internet-enabled devices, which may limit its functionality in areas with connectivity challenges.

## **1.7. Application**

The primary goal of this application is to create a platform that empowers citizens to actively participate in improving community safety and infrastructure. Users can report a variety of issues, including poor road conditions, inadequate signage, and other hazards. By facilitating these reports, the application aims to enhance the overall quality and safety of the community's infrastructure.

## **Chapter 2: Literature Review**

The use of mobile-based solutions for managing community issues, such as roads, traffic, drinking water, drainage, electricity, waste management, and public complaints, has become increasingly relevant and effective in addressing these challenges.

### **2.1. Case study**

#### **2.1.1. Civic Social Complaint Android App**

This app aims to modernize the process of grievance reporting by offering a digital solution leveraging the "Digital India" initiative. This application enables users to register complaints related to public infrastructure, sanitation, and safety. Key functionalities include real-time location tracking, complaint prioritization based on user engagement, and the use of Firebase's real-time database for immediate updates. By promoting transparency and fostering community engagement, the system bridges the gap between citizens and government authorities, ensuring efficient resolution of civic issues (Danish Khan, 2023).

The app introduces several innovative features, such as:

1. **Real-Time Updates & Location Tracking:** Utilizes GPS for precise complaint location, improving response efficiency.
2. **Community Engagement:** Users can like and comment on complaints, fostering collaboration for prioritization.
3. **Complaint Prioritization:** Higher-liked complaints are addressed first, ensuring urgent issues are prioritized.
4. **Statistical Reporting:** Offers detailed reports on complaint statuses (pending, in-progress, resolved) for performance monitoring and problem identification.

While the app introduces significant advancements, it lacks certain critical aspects relevant to Nepal, such as:

1. **Limited Offline Functionality:** The app relies heavily on real-time updates, which might be less practical in Nepal's rural areas with poor internet connectivity.
2. **Focus on Specific Complaint Types:** The existing solution primarily caters to general civic issues but does not extend to emergencies like road accidents or waste management challenges unique to Nepal.
3. **AI Integration & Cross-Platform:** While innovative, the app does not incorporate machine learning for automated complaint categorization, the cross-platform solution with enhanced scalability and analysis, which could significantly enhance resource allocation (Dishant Banga, 2023).

In Nepal, where civic reporting mechanisms are often slow and disconnected, a system incorporating offline support, AI-driven prioritization, and integration of various grievance categories would address these limitations effectively. The proposed system for Nepal will adapt the core features of this app while adding these capabilities to meet local needs.

### **2.1.2. E-Complaint Mobile Application**

The "E-Complaint" application leverages crowdsourcing to address community grievances efficiently within the Philippine barangay system. It automates the traditional complaint-handling process, enabling residents to file, track, and analyze grievances related to environmental and human-to-human conflicts. With cloud-based Software-as-a-Service (SaaS) infrastructure, it provides a scalable solution accessible on major mobile platforms, bridging the gap between citizens and local authorities (Mary Jane C. Samonte).

Key features of the application include:

1. **User-Friendly Interface:** Residents can register complaints with photos and location data.
2. **Crowdsourcing for Community Engagement:** Enables collective participation through complaint categorization and prioritization.

3. **Real-Time Tracking and Notifications:** Users receive acknowledgment and updates about their complaints, enhancing transparency.
4. **Automated Analysis:** Generates data-driven insights, including statistics on complaint types, frequencies, and resolutions, to assist in future decision-making.
5. **Role-Based Access Control:** Administrators manage case details, notifications, and system updates through web and mobile interfaces.

Despite the paper's thorough analysis, several key research gaps remain. One major issue is the focus on a limited range of complaint categories, which restricts the system's applicability and effectiveness. This narrow scope may prevent the system from addressing a wider array of user concerns.

Additionally, the absence of artificial intelligence integration is another critical gap. Without AI, the system lacks the potential for enhanced efficiency and responsiveness, which could significantly improve user experience and complaint resolution.

### **2.1.3. Nagarik App**

The Nagarik App, launched by the Government of Nepal in 2021, aims to digitize public services and enhance citizen engagement. It allows users to report community issues, such as road conditions and waste management while providing access to essential government services like tax payments, vehicle registration, and police clearance certificates. The app promotes transparency by enabling users to track complaints and receive real-time updates (Nepal, 2025).

Despite its success in improving service delivery and empowering citizens, challenges such as the digital divide, low user adoption in rural areas, and concerns over data privacy persist. The app has seen significant engagement, resolving thousands of complaints and simplifying public service access (Nepal, 2025).

## Key Features

1. **Complaint Registration:** Users can report various community issues, including road conditions, waste management problems, and public safety concerns. Complaints can be tracked through the app, ensuring accountability.
2. **Access to Government Services:** The app provides access to essential services such as vehicle registration, tax payments, police clearance certificates, and health insurance information.
3. **User-Friendly Interface:** Designed for ease of use, the app allows users to navigate through its features seamlessly. It requires basic information for registration, such as a valid mobile number and identification documents.
4. **Real-Time Notifications:** Users receive updates on the status of their complaints and service requests, fostering better communication between citizens and government officials.

Since its launch, the Nagarik App has seen significant user engagement, with thousands of complaints registered and resolved. The app has improved transparency in public service delivery and empowered citizens to take an active role in addressing community issues. Feedback from users indicates a positive reception, particularly regarding the convenience of accessing government services.



## **Chapter 3: Methodology**

It provides a structured framework for executing and completing a project. In the context of this project, the methodology outlines the step-by-step process that will be followed to design, develop, and deploy the Smart Community Complaint App. The methodology encompasses various stages, each stage involves specific tasks and activities that contribute to the successful implementation of the project. The methodology ensures a systematic and organized approach, allowing for efficient project Maintenance and the achievement of project objectives.

The purpose of the project methodology is to allow for controlling the entire Maintenance process through effective decision-making and problem-solving while ensuring the success of specific processes, approaches, techniques, methods, and technologies (Balduino, 2021).

In the proposed methodology there are 4 different stages which are the flow of the project, system design, software and hardware requirements and testing and maintenance.

### **3.1. Development and Planning**

#### **3.1.1. Concept and Initiation**

Before starting any project, it's important to collect and check if the requirements make sense. The project can continue if the requirements are feasible. During this phase, everyone involved in the project, like citizens and government officials, works together to gather all the necessary information needed to create and build the project. It's like putting together all the puzzle pieces to make sure we have everything we need. This way, the developers and designers can understand what needs to be done and create a system that solves road-related problems.

### 3.1.2. Definition and planning

In our Smart Community Complaint App project, we aim to develop a mobile application that streamlines the reporting and management of community issues. Key components of the system will include features for user complaint submission, services for addressing these complaints, efficient administration and maintenance of the application, and comprehensive tracking and resolution of reported issues across various categories.

### 3.1.3. Software Used

1. **Flutter:** Flutter is an open-source UI toolkit by Google for building natively compiled applications across mobile, web, and desktop from a single codebase. Its standout features include "Hot Reload" for rapid development, expressive UI elements for visually appealing designs, and high-performance output on both iOS and Android devices. Flutter's adoption by major companies and a supportive community makes it an increasingly popular choice for efficient and cross-platform app development (Google, Flutter, 2025).
2. **Visual Studio Code:** VS Code is a popular source code editor developed by Microsoft. It is widely used by developers for various programming languages and platforms. VS Code offers a lightweight yet powerful environment with features like syntax highlighting, code completion, debugging, and Git integration. It supports a wide range of extensions that enhance functionality, allowing developers to customize their workflow. With its user-friendly interface and extensive community support, VS Code has become a preferred choice for many developers seeking a versatile and efficient code editor (Microsoft, 2025).
3. **Android Studio:** Android Studio is the official integrated development environment (IDE) for Android app development by Google. It's built on IntelliJ IDEA, uses the Gradle build system, and supports a visual UI

designer. With features like an emulator, code analysis, and Google Play integration, Android Studio streamlines app development. It's regularly updated, supports Kotlin, and has a rich plugin ecosystem. Android Studio offers a feature-rich experience, including a visual UI designer, emulator support, and seamless integration with the Android SDK (Google, Android Studio, 2025).

4. **Version Control System – GIT:** Git is a decentralized version control system designed for collaborative software development. Git tracks changes in source code, allowing developers to work on projects simultaneously. Key features include commits for tracking changes, branching for independent development, and merging for combining changes. Git repositories can be local or hosted remotely on platforms like GitHub. Its popularity is attributed to its efficiency, flexibility, and essential commands like: init, add, commit, push, and pull. Git is integral to modern development, streamlining collaboration and ensuring codebase integrity (Team G. , 2025).

#### **3.1.4. Development**

The Smart Community Complaint App project utilized the following technologies:

1. **Dart Programming Language:** Dart is a versatile programming language for building web, mobile, and server applications. Known for its strong typing and object-oriented structure, Dart is particularly popular in conjunction with Flutter, allowing developers to create apps for multiple platforms from a single codebase. It is used to build native mobile apps for Android or iOS, desktop apps, and servers. It uses a source-to-source compiler to arrange codes more quickly. Hence, it has a more efficient UI in comparison to other programming languages (Team D. , 2025).

2. **Flutter SDK:** Flutter SDK facilitates cross-platform app development. It uses Dart as its programming language and offers a rich set of customizable widgets for building expressive user interfaces. Key features include Hot Reload for real-time code changes, a single codebase for iOS and Android, and seamless integration with native features. With a vibrant community and regular updates, Flutter is known for its versatility in creating visually appealing and high-performance applications across mobile, web, and desktop platforms (Google, Flutter, 2025).
3. **Python:** Python, with its simplicity and versatility, is well-suited for developing the Smart Community Complaint App through frameworks like Kivy and BeeWare, which enable cross-platform compatibility. These tools allow developers to write code once and deploy it on both Android and iOS, reducing development time and costs. Python's extensive libraries and efficient resource utilization make it an ideal choice for creating user-friendly and scalable applications like this project (Foundation, 2025).
4. **AI/ML Integration:** The Smart Community Complaint App incorporates AI/ML to enhance efficiency and accuracy in handling complaints. Machine Learning models will primarily be used for:
  - **Image Recognition:** Identifying the type and severity of the issue (e.g. potholes, drainage blockages) from uploaded images.
  - **Complaint Categorization:** Automatically classifying reports into predefined categories based on keywords and image analysis.
  - **Priority Determination:** Leveraging algorithms to prioritize complaints based on severity, location density, and community impact.

**Technological Approach:**

- TensorFlow or PyTorch: For building and deploying image recognition models.
- OpenCV: For preprocessing and enhancing image data.

- Data Collection: Collecting labelled datasets of common complaints, such as pothole images or drainage issues.
- Data Augmentation: Enhancing the dataset with varied lighting, angles, and conditions to improve model robustness.

By including AI/ML at key stages of development and maintenance, the Smart Community Complaint App is equipped with advanced capabilities to deliver a seamless and efficient user experience (Anglen, 2024).

### **3.1.5. Launch and Execution**

Once the development of the Smart Community Complaint App is complete, it will be launched for real-world usage. Users will be able to access the system, submit 15 complaints, and interact with its features. Compatibility will be ensured, testing conducted, and valuable user feedback gathered to make necessary improvements.

### **3.1.6. Performance and Control**

The system's speed, stability, and scalability will be continuously monitored. Regular performance tests will be conducted to ensure it meets project goals. Measures for data backup and security will also be implemented to maintain control and ensure a seamless user experience.

## **3.2. System Design**

After gathering and analyzing the requirements, the Mobile Road Maintenance Complaint System was given a solid structure. During this phase, the project's architecture was designed based on the collected requirements. Creative and intuitive diagrams such as system flow, ER diagrams, system architecture, DFDs, and use cases were created. Those visual representations helped in visualizing the system's structure, data flow, and interactions. By carefully designing the architecture, the foundation was set for the successful development and implementation of the road complaint maintenance system.

### 3.2.1. System Flow Diagram

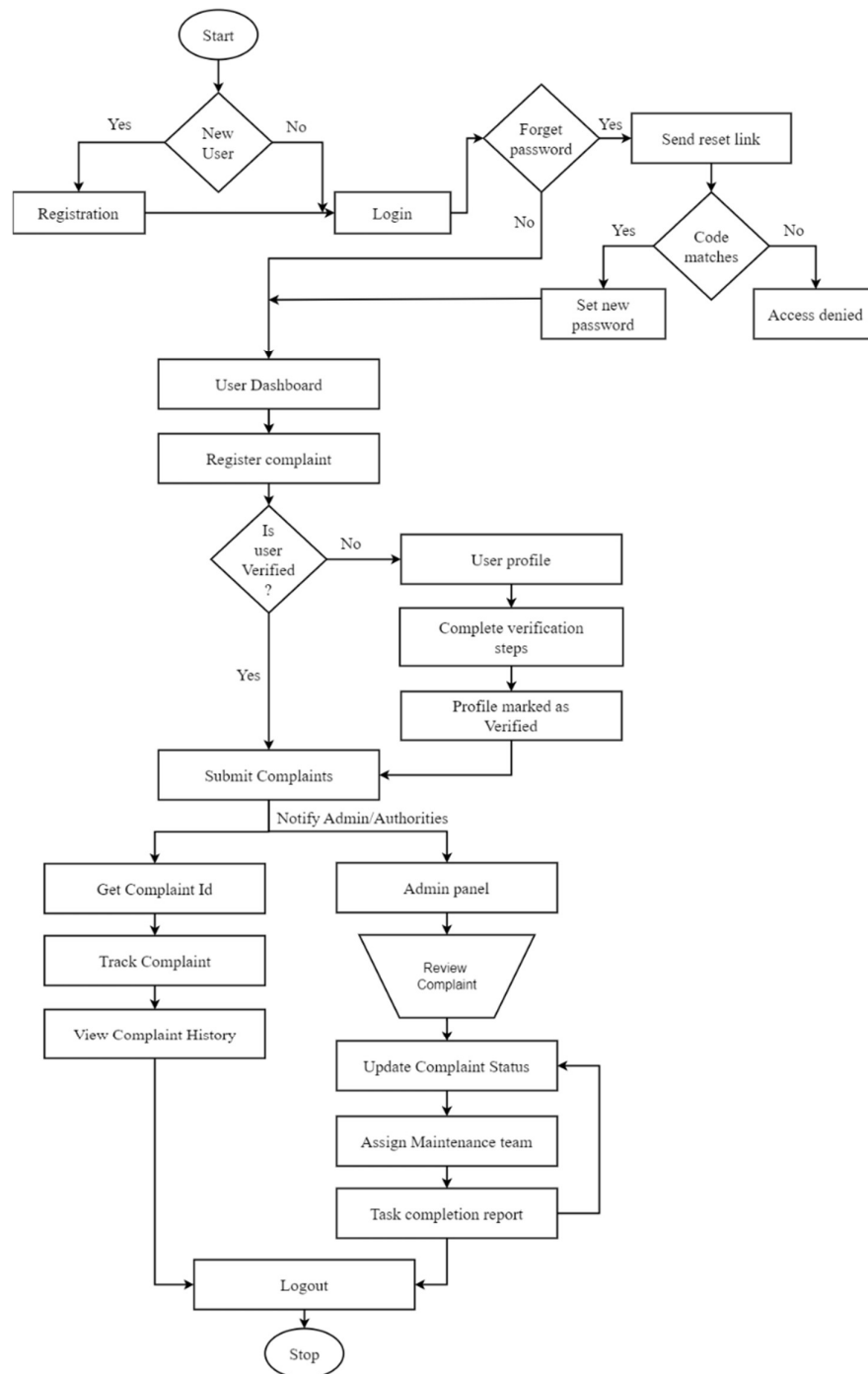


Figure 3.1 System Flow Diagram  
(Group Study, 2025)

### 3.2.2. E-R diagram

E-R Diagram stands for Entity Relationship Diagram, also known as ERD, which is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes, and relationships. ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes, and diamond shapes to represent relationships

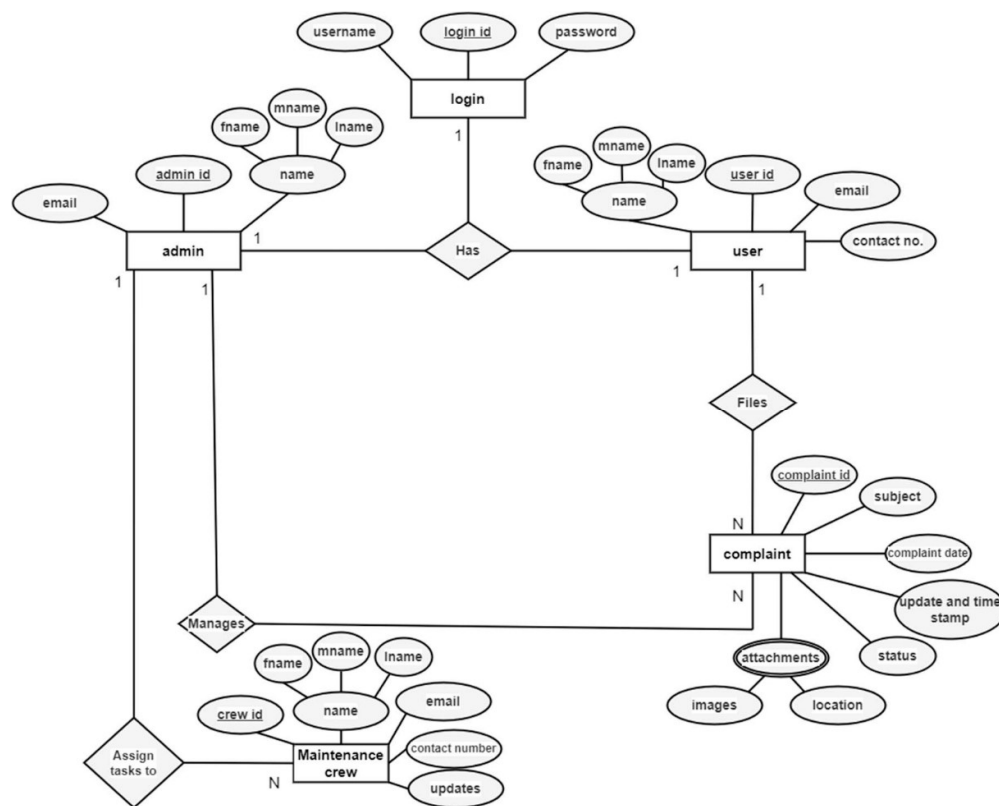


Figure 3.2: ER Diagram  
(Group Study, 2025)

### 3.2.3. DFD (Data Flow Diagram)

Data Flow Diagrams (DFDs) were important tools for our Road Complaint Maintenance System project. DFDs visually showed how data moved in our system, helping us find and fix issues to make it work better. They gave us a clear picture of how our system was built and what it did, making it easier for our team to work together. DFDs could be used at different levels, from big pictures to small details, to help us design and improve our system.

They also helped us understand how data connected and moved, ensuring everything worked smoothly. Using DFDs in the project improved the understanding, design, and implementation of the Road Maintenance Complaint System.

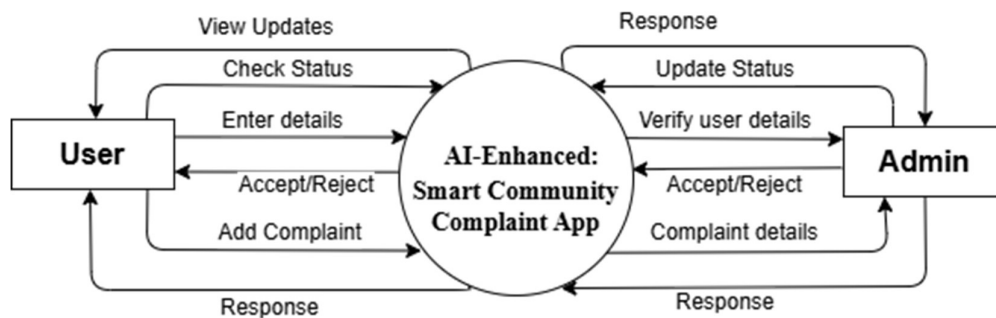


Figure 3.3: Level 0 Data Flow Diagram  
(Group Study, 2025)



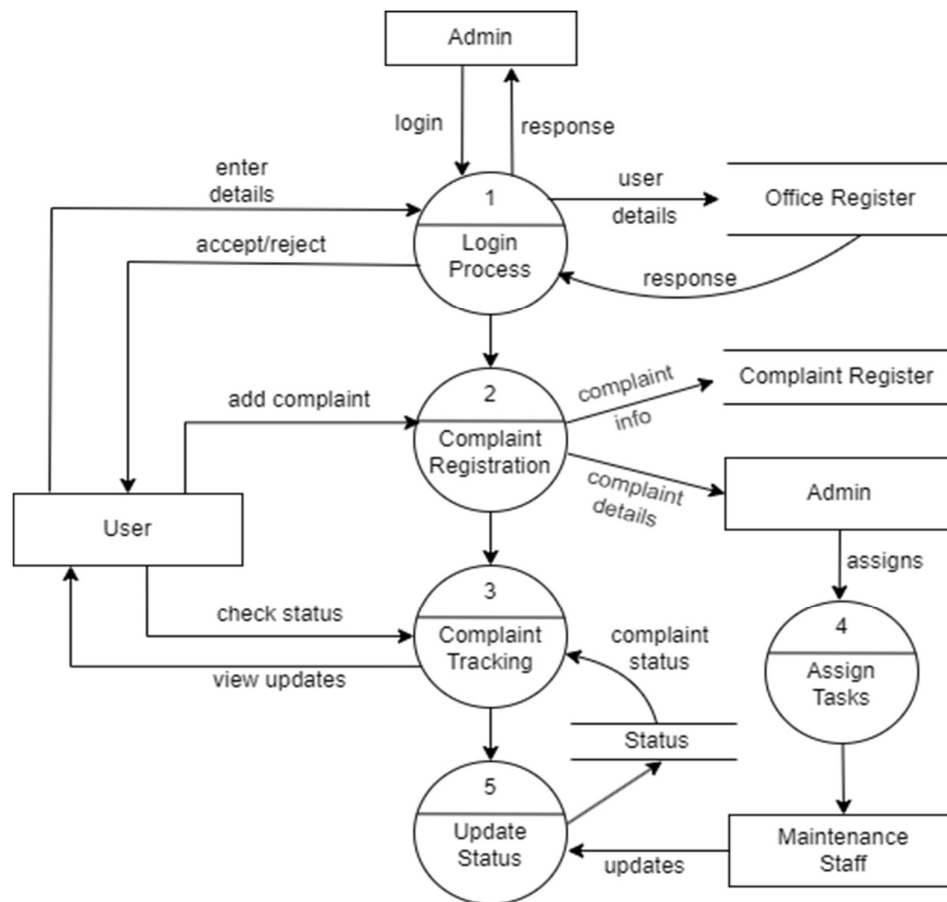


Figure 3.4: Level 1 Data Flow Diagram  
(Group Study, 2025)

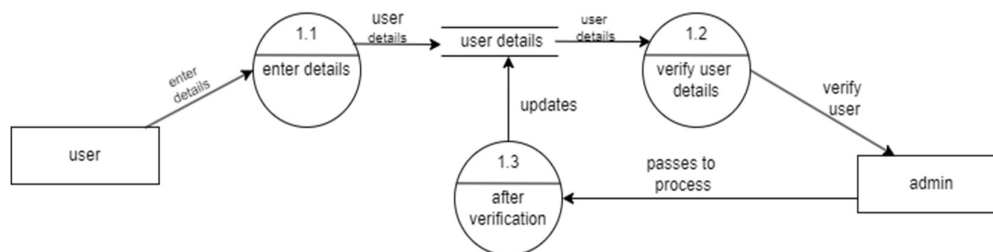


Figure 3.5: Level 2 DFD for Login Process  
(Group Study, 2025)

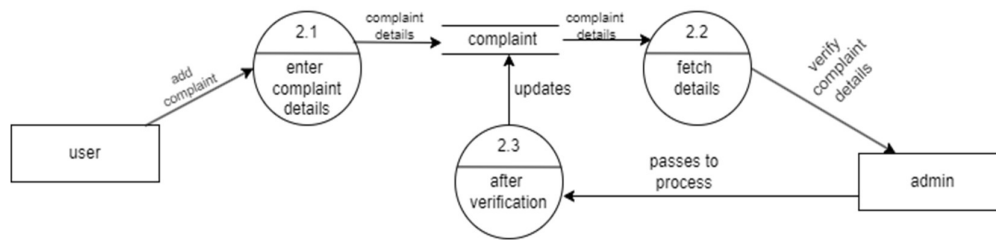


Figure 3.6: Level 2 DFD for Complaint Registration  
(Group Study, 2025)

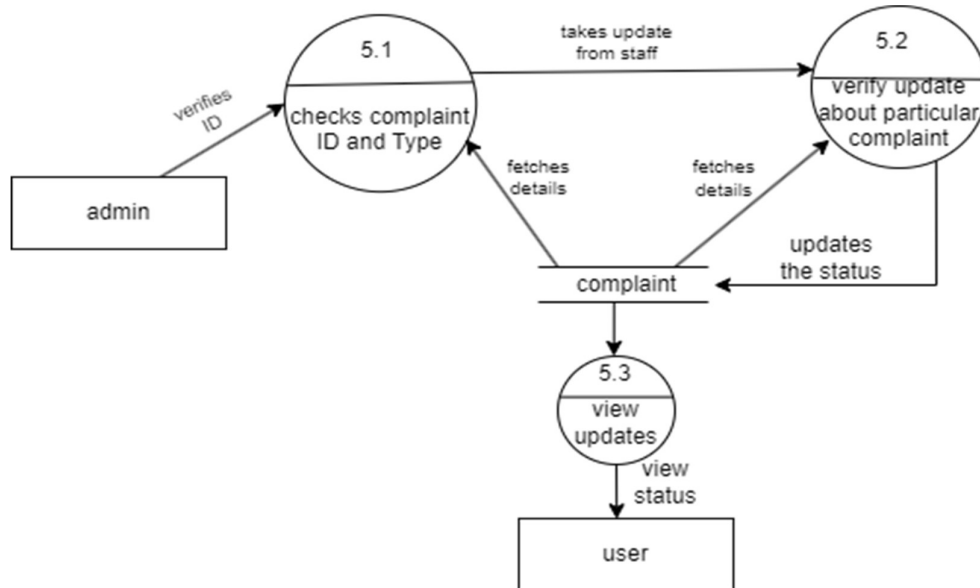


Figure 3.7: Level 2 DFD for Update Status  
(Group Study, 2025)

### 3.2.4. Use case Diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

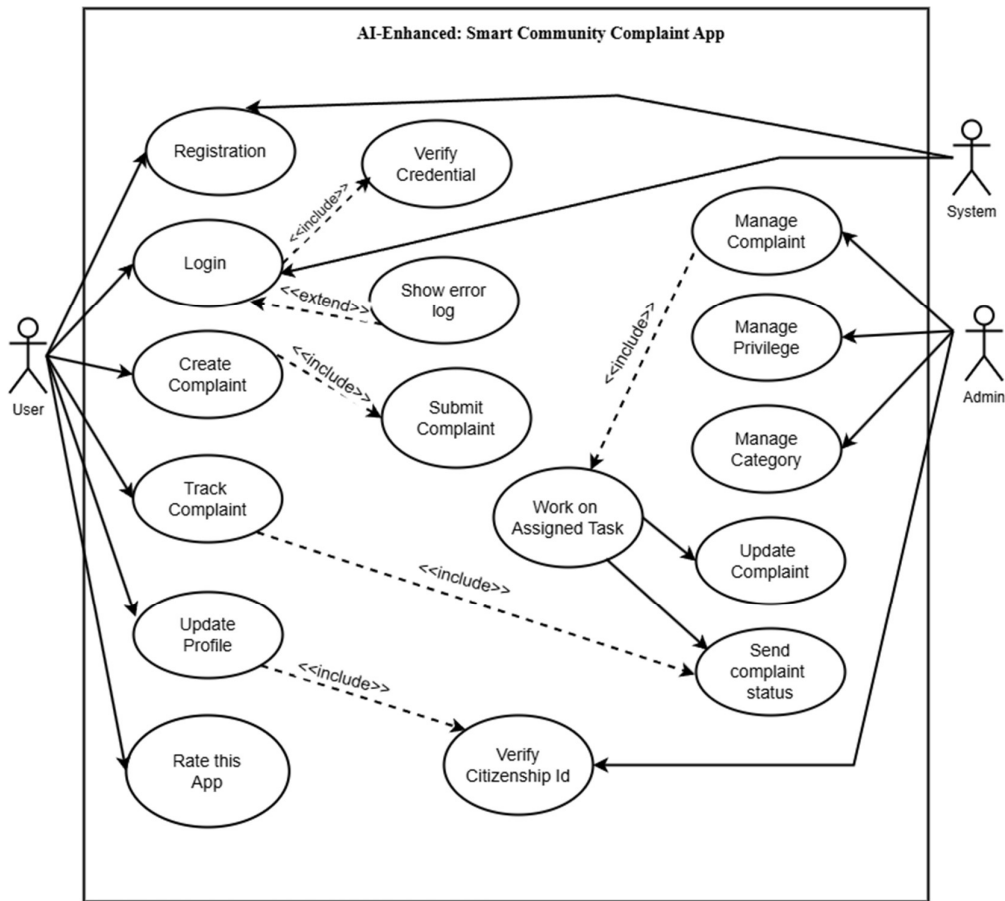


Figure 3.8 Use case Diagram  
(Group Study, 2025)

### 3.2.5. Software Development Model

The Agile model is a flexible and iterative approach to software development that emphasizes collaboration, customer feedback, and small, incremental releases. Instead of planning the entire project upfront, Agile breaks down the work into smaller, manageable chunks called "sprints," typically lasting 1-4 weeks. Each sprint results in a potentially shippable product increment, allowing teams to adapt to changing requirements and deliver value to customers continuously. Agile encourages close collaboration between developer teams and stakeholders, ensuring that the final product closely aligns with user needs and business goals (Team G. , 2024).

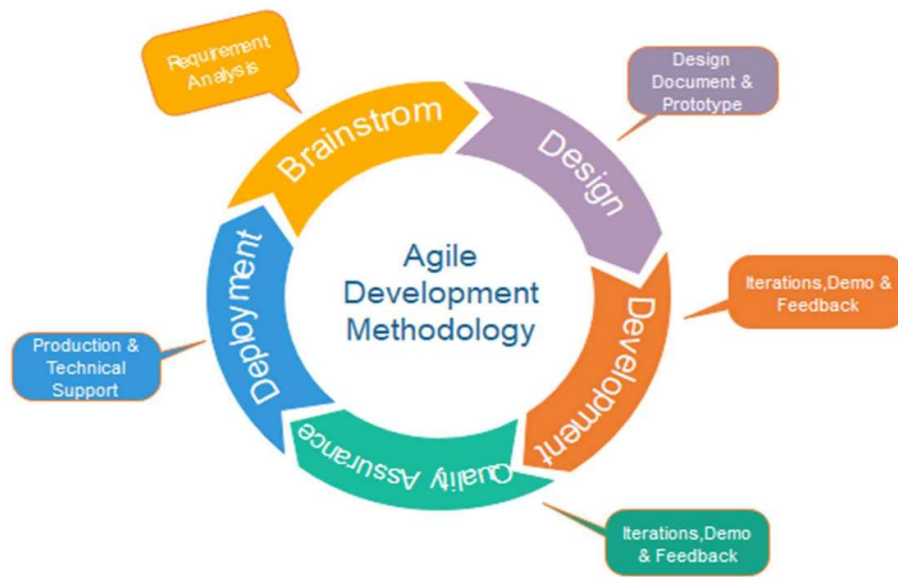


Figure 3.9: Agile Development Model  
(Group Study 2025)

### 3.3. Software and hardware requirements

For our Smart Community Complaint App project, the hardware and software requirements are as follows:

#### Hardware Requirements:

1. **Mobile Devices:** Any smartphone or tablet running the supported operating system.
  - **Android:**  
Flutter often supports Android versions 4.1 (API level 16) and above.
  - **iOS:**  
For iOS, Flutter typically supports iOS versions 9.0 and above.

#### Software Requirements:

1. **Operating System:**  
Compatible with various operating systems, including Android and iOS.

**2. Internet Connectivity:**

Access to a stable internet connection for submitting road complaints and receiving real-time updates.

**3. Camera Access:**

Permission for the app to access the device's camera if users want to include photos when reporting road issues.

**4. Location Services:**

Permission for the app to access the device's location services for precise geolocation when reporting road problems.

By ensuring compatibility with a wide range of devices, Our Smart Community Complaint App can be accessed and used seamlessly by users on their smartphones.

### **3.4. Testing and Maintenance**

Before launching a project, it's important to test it thoroughly to make sure it works as expected. Mobile apps run on different devices like smartphones and tablets, which is handy. Testing involves checking that everything works correctly—like buttons, forms, and how the app connects to databases. This step helps catch and fix any problems before people start using the mobile app. Ultimately, this testing phase plays a crucial role in mitigating potential issues before the application's release to end users.

## Chapter 4: Expected Outcome

The expected outcome of our project is to improve road conditions in Nepal. By implementing our Smart Community Complaint App, we aim to achieve the following:

1. **Efficient Issue Reporting:** Provide a user-friendly platform for citizens to report road problems promptly and accurately.
2. **Timely Response:** Enable authorities to respond quickly to reported issues, leading to prompt repairs and maintenance.
3. **Road Safety Improvement:** Address problems like potholes, damaged signs, and unsafe conditions to enhance overall road safety.
4. **Data Analysis:** Collect and analyze data on reported issues to gain insights for decision-making and future infrastructure developments.
5. **Community Engagement:** Empower citizens to actively participate in road improvement efforts, fostering a sense of community involvement.
6. **Transparency and Accountability:** Ensure visibility and accountability in addressing reported issues, building trust and confidence in the system.

Our project aims to create an effective system that streamlines issue reporting, enhances road safety, promotes community engagement, and ensures accountability in addressing road-related problems.

## REFERENCES

- Acharya, D. S. (2023, November 29). Retrieved from nepallivetoday: <https://www.nepallivetoday.com/2023/11/29/challenges-and-prospects-in-sustainable-infrastructure-development-in-nepal/>
- Adhikari, E. R. (2024, December 05). Retrieved from thehimalayantimes: <https://thehimalayantimes.com/opinion/transportation-challenges-in-nepal-consequences-and-solutions>
- Anglen, J. (2024). Retrieved from Rapid Innovation: <https://www.rapidinnovation.io/post/top-10-open-source-computer-vision-libraries-you-need-to-know>
- Balduino, R. (2021, March 9). Retrieved from Medium: How AI can transform your complaints management strategy-82cba16a82eour complaints management strategy [https://medium.com/ibm-data-ai/how-ai-can-transform-your-complaints-management-strategy-82cba16a82eour complaints management strategy](https://medium.com/ibm-data-ai/how-ai-can-transform-your-complaints-management-strategy-82cba16a82eour-complaints-management-strategy)
- Danish Khan, S. S. (2023). CIVIC SOCIAL COMPLAINT ANDROID BASED APP. *International Research Journal of Modernization in Engineering Technology and Science*.
- Dishant Banga, K. P. (2023). Artificial Intelligence for Customer Complaint. *International Journal of Computer Trends and Technology* .
- Foundation, P. S. (2025). Retrieved from Python: <https://www.python.org/>
- Google. (2025). Retrieved from Android Studio: <https://developer.android.com/studio>
- Google. (2025). Retrieved from Flutter: <https://docs.flutter.dev/resources/faq>
- Lamichhane, R. (2024, February 11). Retrieved from The Kathmandu Post: <https://kathmandupost.com/columns/2024/02/11/paving-the-path-to-prosperity>
- Maharjan, N. (2019). Retrieved from Nepal Economic Forum: <https://nepaleconomicforum.org/revisiting-nepals-infrastructure-bottlenecks-and-progress/>
- Mary Jane C. Samonte, J. M. (n.d.). E-Complaint: An Analytical Crowdsourcing Mobile Application for Community Peace and Order System.

Microsoft. (2025). Retrieved from Visual Code: <https://code.visualstudio.com/>

Nepal, G. o. (2025). Retrieved from Nagarik App: <https://nagarikapp.gov.np/>

Purbey, A. (2023, June 14). Retrieved from nepallivetoday: <https://www.nepallivetoday.com/2023/06/14/what-impedes-nepals-infrastructure-governance/>

Team, D. (2025). Retrieved from Dart: <https://dart.dev/>

Team, G. (2024, july 3). Retrieved from GeeksforGeeks: <https://www.geeksforgeeks.org/software-engineering-agile-development-models/>

Team, G. (2025). Retrieved from Git SCM: <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>

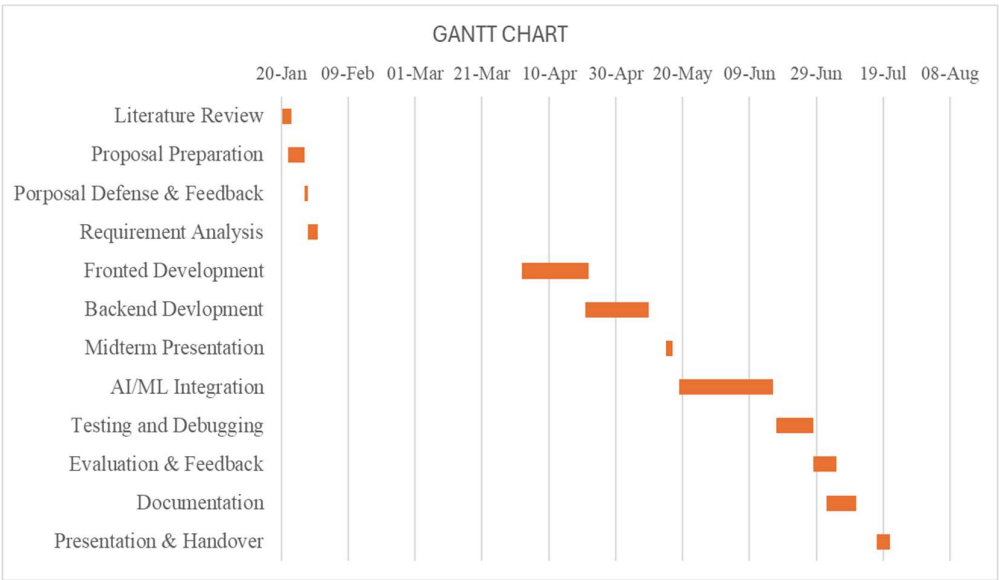
Thapa, E. K. (2024, August 05). Retrieved from nepalinfrastructuresummit: <https://nepalinfrastructuresummit.com.np/blog/sustainable-urban-infrastructure>



# APPENDICES

Before starting any project, it's important to create a clear schedule that outlines the tasks and milestones for the development phase. To ensure effective time Maintenance, we have prepared a Gantt chart covering 3 months, spanning 10 weeks from the phase after the proposal defense to the final report submission and defense. This schedule acts as a roadmap, helping us stay organized, track progress, and meet deadlines. By following the allocated time frames for each task, we can ensure efficient project execution and timely completion.

Table 1: Gantt chart of the project



(Group Study, 2025)

The proposed research project is structured as a comprehensive academic initiative, strategically aligned with the academic calendar across the 7th and 8th semesters. The initial phase encompasses critical preparatory activities, beginning with a rigorous literature review to establish a robust theoretical foundation. This initial stage is immediately followed by meticulous proposal preparation and a focused

proposal defense, allowing for comprehensive requirement analysis and strategic project scoping.

The development trajectory progresses through distinct technical stages, commencing with frontend development and subsequently transitioning to backend infrastructure construction. A midterm presentation provides a critical checkpoint for progress evaluation and stakeholder engagement. The project distinguishes itself through a substantial AI/ML integration phase, representing a significant technological component that demands 28 days of dedicated implementation and refinement.

The concluding phases prioritize systematic validation and documentation, featuring comprehensive testing and debugging protocols, followed by structured evaluation and feedback mechanisms. The project culminates in a comprehensive documentation process and a formal presentation and project handover, ensuring a structured and professional conclusion to the research initiative. This methodical approach underscores a rigorous, academically-oriented project management strategy that balances technical innovation with methodical execution.