Module Code & Module Title

CS6P05NI Project

Proposal

5% Individual Coursework

Submission: Final Submission

AY 2025 2026

Academic Semester: Autumn Semester 2025

Credit: 30 Credit Year Long Module

Student Name: Bikash Kumar Das

London Met ID: 23049171

College ID: np01cp4a230215

Assignment Due Date: Wednesday, December 3, 2025

Assignment Submission Date: Wednesday, December 3, 2025

Internal Supervisor: Aayush Pandey

External Supervisor: Ankit Tandukar

I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Contents

Table of Figure

Table of Table

1. INTRODUCTION

1.1 Introduction to topic

Overseas education and visa consultancies play a crucial role in guiding students and applicants through complex admission, documentation, and visa processes (Business360, 2024). However, many consultancy organizations still rely on traditional manual systems such as spreadsheets, paperwork, and offline communication, which often lead to data loss, inefficiency, communication delays, and limited transparency for clients (Rising Nepal, 2024). As the number of students seeking international opportunities continues to rise, there is a growing demand for digital solutions that can streamline operations, improve service delivery, and enhance overall client experience (ICEF Monitor, 2025)

To address these challenges, this project proposes a web-based **Consultancy Management System** that automates common consultancy operations including user registration, client and counselor management, document handling, appointment scheduling, progress tracking, and communication. The system is built using **Next.js** for an intuitive and responsive frontend interface and **Python Flask** as a lightweight, scalable backend for API development. **MySQL** is used as the relational database to securely store structured client, process, and document data. Modern security measures such as password hashing, CORS protection, and JSON Web Token (JWT) authentication are implemented to ensure data privacy and role-based access control.

1.2 Problem Scenario

• Over **3,500** consultancies exist in Nepal, but only about **1,500** have official approval, creating regulatory inconsistencies (Rising Nepal, 2024).

• More than **90%** of students depend on consultancies, yet many firms lack automated platforms for service delivery (Business360, 2024).

• Manual, spreadsheet-based operations contribute to data errors, duplication, and documentation loss, slowing the application process (Business360, 2024).

• Clients have limited real-time visibility, leading to repeated inquiries and reduced (Poidata, 2025).

• Administrators lack centralized analytics for visa success rates, counselor workload, and client conversion trends, restricting strategic planning (Rising Nepal, 2024).

• Over **60%** of consultancies still rely on telephone inquiries and walk-in communication as their primary support channel, leading to slow response times and poor record-keeping (ICEF Monitor, 2025).

• Nearly **35%** of student applications submitted through consultancies contain documentation errors or incomplete files due to the absence of standardized digital verification systems (Ministry of Education Nepal, 2025).

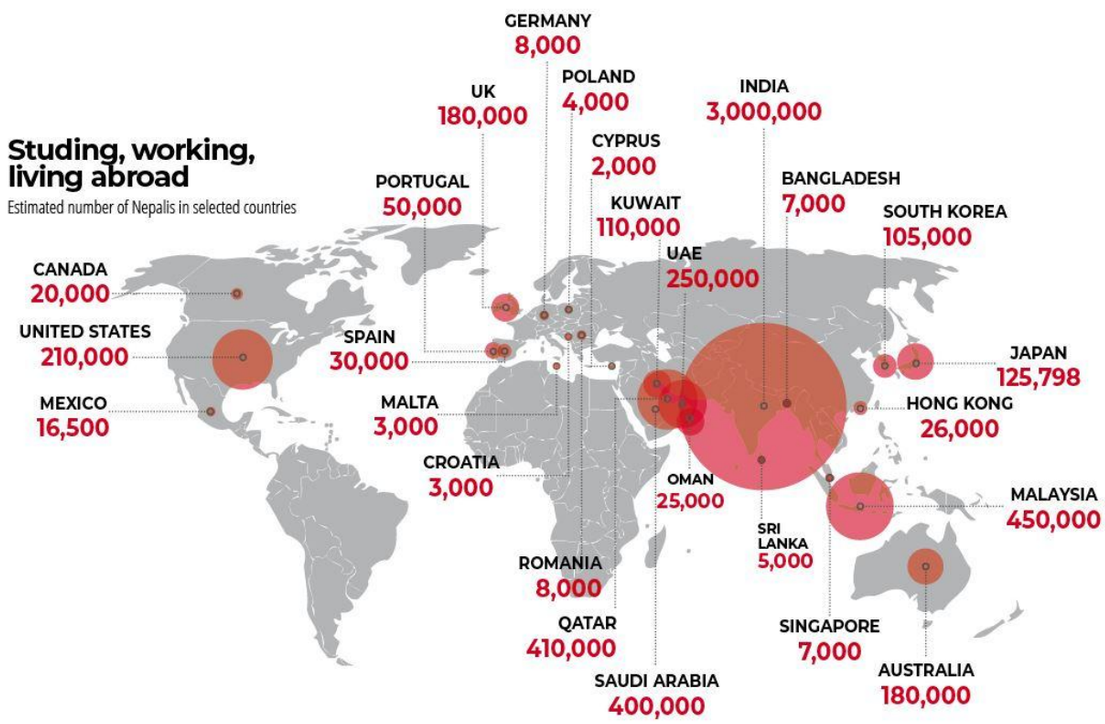

Figure 1:Estimated number of Nepalis studying, working and living abroad in selected countries (Nepali Times, 2025)

1.3 The project as a solution

The Consultancy Management System modernizes operations by replacing manual processes with a centralized platform built using Next.js, Flask, and MySQL. It improves workflow efficiency, real-time visibility, and user experience while ensuring strong data security through role-based access, hashed passwords, and JWT authentication.

It could provide the following benefits:

- Enhances operational efficiency by digitizing manual processes and centralizing data management.
- Improves client transparency by allowing real-time access to application progress and document requirements.
- Enables administrators to make informed decisions using analytics on counselor performance, student flow, and visa outcomes.
- Reduces data loss, duplication, and communication errors caused by paper files and spreadsheet-based systems.
- Increases client satisfaction and accountability through traceable workflows and secure, structured information handling.

2. AIM AND OBJECTIVES

2.1 Aim

The primary aim of the project is to develop a centralized, secure, and automated web-based Consultancy Management System that digitizes client handling, improves transparency, enhances workflow efficiency, and provides real-time visibility of application progress. The system helps consultancies modernize their operations, reduce manual workload, and improve service quality through structured data management and analytical insights.

2.2 Objectives

The following are the objectives of the project to improve consultancy operations and service delivery:

- Carry out an in-depth study of existing consultancy workflows and identify inefficiencies in manual processes.
- Design and develop a web-based system using Next.js, Flask, and MySQL for reliable and scalable consultancy management.
- Implement role-based authentication and authorization using password hashing and JWT to ensure secure user access.
- Enable clients to register, upload documents, book appointments, and track their application progress in real time.
- Provide counselors with tools to manage assigned clients, verify documentation, schedule meetings, and update application stages.
- Create an administrator dashboard with analytical insights such as visa success rates, client conversion, counselor workload, and operational performance.
- Integrate optional AI-based features such as automated FAQ responses or course recommendations to enhance client support.
- Reduce data loss and miscommunication by replacing spreadsheets and physical files with centralized digital record management.
- Ensure that the system enhances accountability and transparency across clients, counselors, and administrators.
- Produce complete technical documentation and a user manual for deployment and maintenance.

3.EXPECTED OUTCOMES AND DELIVERABLES

3.1 Expected Outcomes

Upon successful implementation of the proposed system, a fully functional web-based **Consultancy Management System** is expected to be developed using modern technologies (Next.js, Flask, and MySQL). The system is designed to automate core consultancy workflows, improve transparency, and provide stakeholders with secure and reliable access to information. The major outcomes of the system are outlined below:

1. (Client/Student) Features

A.Primary Features

1. User Registration & Login

- Create account using email/phone
- Secure login with JWT
- Password reset via email

2. Profile Management

- Update personal information
- Add educational background
- Add preferred country, course, budget

3. Appointment Booking

- View counselor availability
- Book appointment (online/offline)
- Reschedule or cancel bookings

4. Application Progress Tracking

- View timeline such as:
  - Document Collection
  - Application Submitted
  - Offer Letter
  - Visa Lodged
  - Visa Approved/Rejected
- Real-time status updates

5. Document Upload

- Upload passport, transcripts, SOP, bank statements
- View verified/not verified status
- Replace or update documents

6. Communication

- Send messages or queries to counselor
- Receive replies
- Notification alerts

B. Secondary Features

1. AI Chatbot / FAQ Assistant

- Ask questions like:
  - “Requirements for Canada?”
  - “What documents do I need?”
- Automated responses

2. Course/Country Recommendation

- AI suggests program based on:
  - Grade
  - Budget
  - Preferences

3. Email Notifications

- Appointment reminders
- Progress updates
- Document verification alerts

2. Consultant / Counselor Features

A. Primary Features

1. Counselor Dashboard

- View all assigned students
- See pending tasks
- Today’s appointments

2. Client Management

- Add/edit/update client info

- Assign or reassign clients
- Check client's progress

3. Appointment Handling

- Accept/Reject appointments
- Reschedule if needed
- Mark completed consultations

4. Document Verification

- View documents uploaded by client
- Mark documents as:
  - Verified
  - Rejected
  - Needs correction
    - Add comments on documents

5. Application Processing

- Update application stages
- Upload offer letters
- Prepare visa documents
- Mark visa outcome

6. Communication

- Reply to client queries
- Send instructions/messages

B. Secondary Features

1. Analytics for Counselor

- Number of clients
- Offer letter success rate
- Visa approval rate

2. Internal Notes

- Keep private notes about client cases

3. Admin Features

A.Primary Features

1. Admin Dashboard

- Overview of:
  - Total clients
  - Total counselors
  - Visa success rate

2. User & Role Management

- Add new counselors
- Edit counselor profile
- Activate/deactivate accounts
- Manage admin roles

3. Reports & Analytics

- Lead conversion report
- Counselor performance report
- Country popularity chart
- Visa success statistics

B. Secondary Features

1. Payment & Commission Management

- Track student payments
- Track commission from universities
- Generate invoices

2. Notification Management

- Configure email template
- Set automated reminder rules

3.2 Deliverables

After achieving the above expected outcomes, the project is expected to produce the following deliverables intended for relevant stakeholders:

Deliverable 1: Functional Prototype

A fully working prototype of the Consultancy Management System demonstrating the following capabilities:

- Secure role-based login using JWT and hashed passwords
- Client-side features including registration, uploads, appointment booking, and progress tracking

- Counselor dashboard for application management, document review, and communication
- Administrator dashboard displaying system-wide analytics and performance insights
- Centralized MySQL database for structured, secure information storage
- Optional AI module for automated support or recommendations

Deliverable 2: Technical Documentation

A comprehensive technical document containing:

- System architecture diagrams
- Backend and frontend component descriptions
- Database schema and data flow models
- API endpoints and integration details
- Installation, and configuration instructions

4. PROJECT RISKS, THREATS, AND CONTINGENCY PLANS

Since all software systems face risks during development, several technical and managerial challenges may arise in building the Consultancy Management System. The following risks, along with contingency plans, help reduce system failure and ensure smooth development and deployment.

4.1 Risks and Threats

Risk Category	Description
Security Risks	Weak access control or insecure storage may expose sensitive student data.
API Reliability Risks	Flask API errors or downtime may disrupt data flow to the frontend.
Database Integrity Risks	Poor schema or query issues may cause data corruption or duplication.
System Performance Risks	High traffic may slow API responses and affect key features.
Time & Project Management Risks	Limited time may lead to delayed or incomplete feature development.
Data Loss Risks	Server failures or accidental deletions may result in lost files or records.
Integration Risks	Miscommunication between Next.js and Flask may cause route or CORS issues.

Table 1 : Risks and Threads

4.2 Contingency Plans

Risk Category	Contingency Plan
Security Risks	Use JWT, hashing, RBAC, validation, HTTPS, and strict CORS.
API Reliability Risks	Add validation, error handling, monitoring, and fallback UI.
Database Integrity Risks	Use good schema, indexing, transactions, and backups.
System Performance Risks	Optimize queries, use caching/pagination, monitor performance.
Project Management Risks	Use sprints, track progress, prioritize key features.

Risk Category	Contingency Plan
Data Loss Risks	Enable backups, cloud storage, and versioned recovery.
Integration Risks	Test early, document APIs, fix CORS settings.

Table 2: Contingency Plans

.

5. METHODOLOGY

5.1 Considered Methodology

At the planning stage of my project, I explored several software development methodologies to determine which one would best support my objectives and project scope. I researched the **Waterfall model** (Sommerville, I., 2016), focusing on its linear structure and clear phase-by-phase progression. I also studied the **Agile methodology** (Pressman, 2020), examining how its iterative cycles and flexibility could help adapt to changing requirements. In addition, I looked into **Scrum**, an Agile framework that uses sprint-based development and continuous feedback to improve efficiency (Schwaber, K. & Sutherland, J., 2020).

For More Detail Refer to Appendix 12.3

5.2 Selected Methodology

Among the approaches that were taken into consideration, the Agile Methodology was chosen for the project. The agile methodology is an iterative methodology for project management that divides work into many flexible stages known as sprints. Agile values team collaboration, working software, customer collaboration and adaptability (Pressman, 2020).

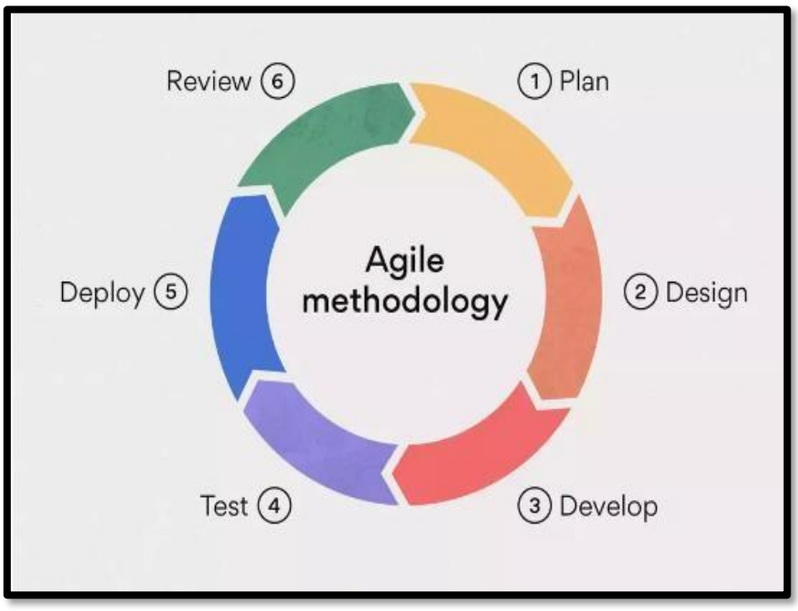

Figure 2 : Agile methodology phases (Anon., n.d.)

More More Detail Refer to Appendix 12.4

Scrum:

Scrum is a management methodology used by software teams to self-organize, collaborate, and learn from experience, enabling efficient and sustainable problem-solving in complex projects (Schwaber, K. & Sutherland, J., 2020).

Why I Am Using Agile?

- Requirements can naturally evolve as the project moves forward.
- Each sprint ends with testing and feedback, helping ensure the project stays on track.
- Early testing helps identify and reduce risks before they become major issues.
- New features can be continuously integrated throughout the development process.
- This approach works especially well for multi-module systems like this project.

How Agile Will Be Used

- The project is divided into **8 sprints**, each lasting **1–2 weeks**.
- At the end of every sprint, the completed module will be tested and reviewed.
- Feedback from supervisors, mentors, or test users will be incorporated.
- Sprint adjustments can be made if requirements change.

Sprint Deliverables Include

- Working modules (billing, hiring, chatbot, inventory, etc.)
- Documentation updates
- Testing outcomes
- UI improvements
- Bug fixes and refinements

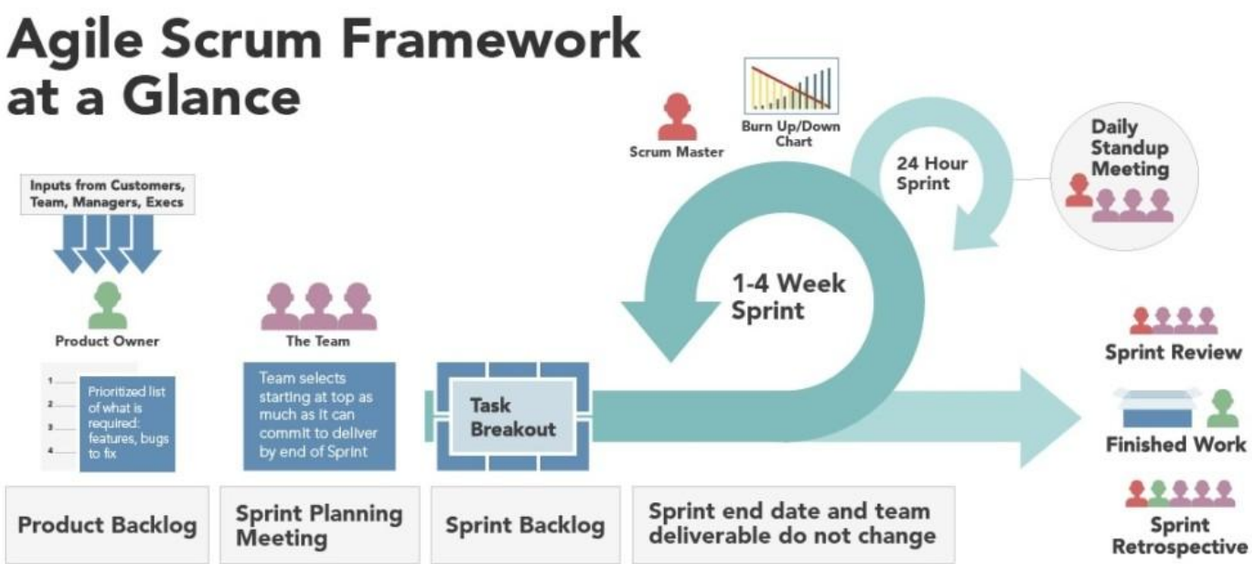

Figure 3:Agile Scrum Framework (Anon., 2025)

6. RESOURCE REQUIREMENTS

To start up the project and create the system there are components that will be used which are as follows:

6.1 Hardware

Hardware	Specification
Laptop	11th Gen Intel Core i5-11400H, 8GB RAM, 512GB SSD, NVIDIA GeForce RTX 3050 GPU

Table 3 : Hardware Requirements

6.2 Software Required

Software	Short Specification
Next.js	Frontend development framework.
Node.js	Runtime to run Next.js.
Python	Backend programming language.
Flask	Framework for building APIs.
MySQL	Database for storing system data.
JWT	Secure token-based authentication.
Password Hashing	Secure password storage.
CORS	Controls API access from frontend.
Axios / Fetch	Sends requests between frontend & backend.
Git	Version control for code.
GitHub	Online code repository & backup.
Cloud Storage	Stores uploaded documents.
Deployment Platform	Hosts frontend & backend online.
API Testing Tools	Tests API endpoints.
Documentation Tools	Used for writing and diagrams.
Backup Storage	Stores project backups.

Table 4: Software Requirements

7. WBS

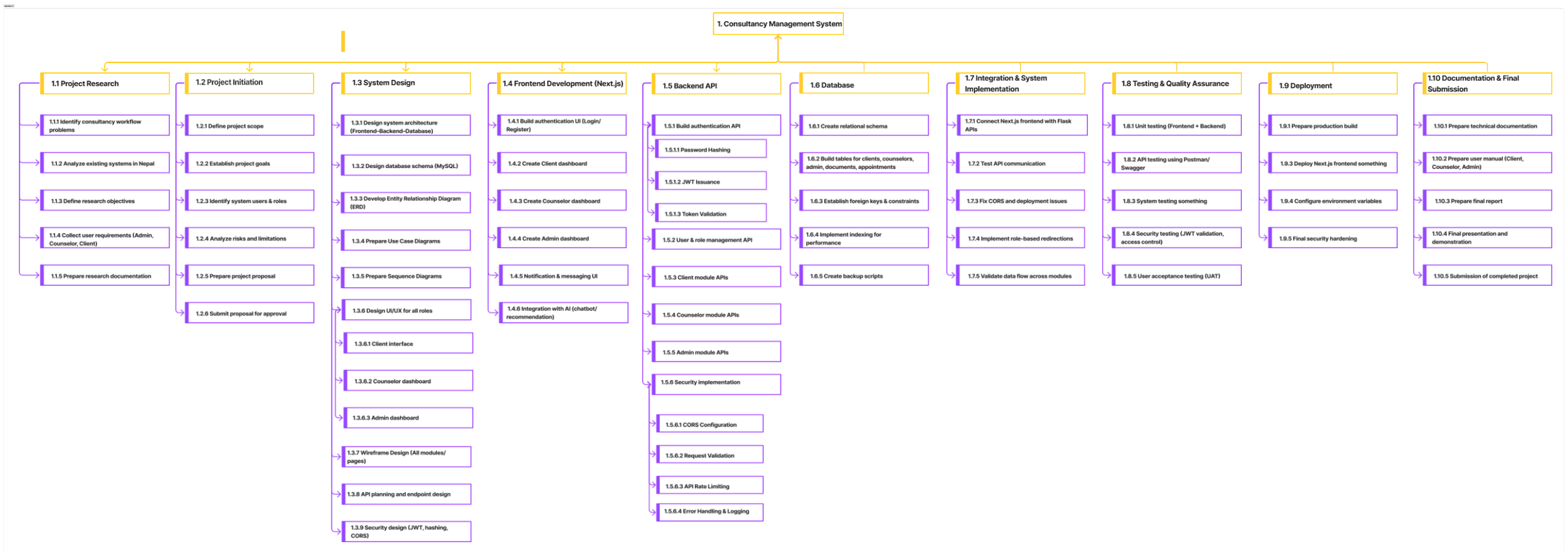

Figure 4: WBS

For More Detail of this WBS refer to Appendix 12.2

8. MileStone

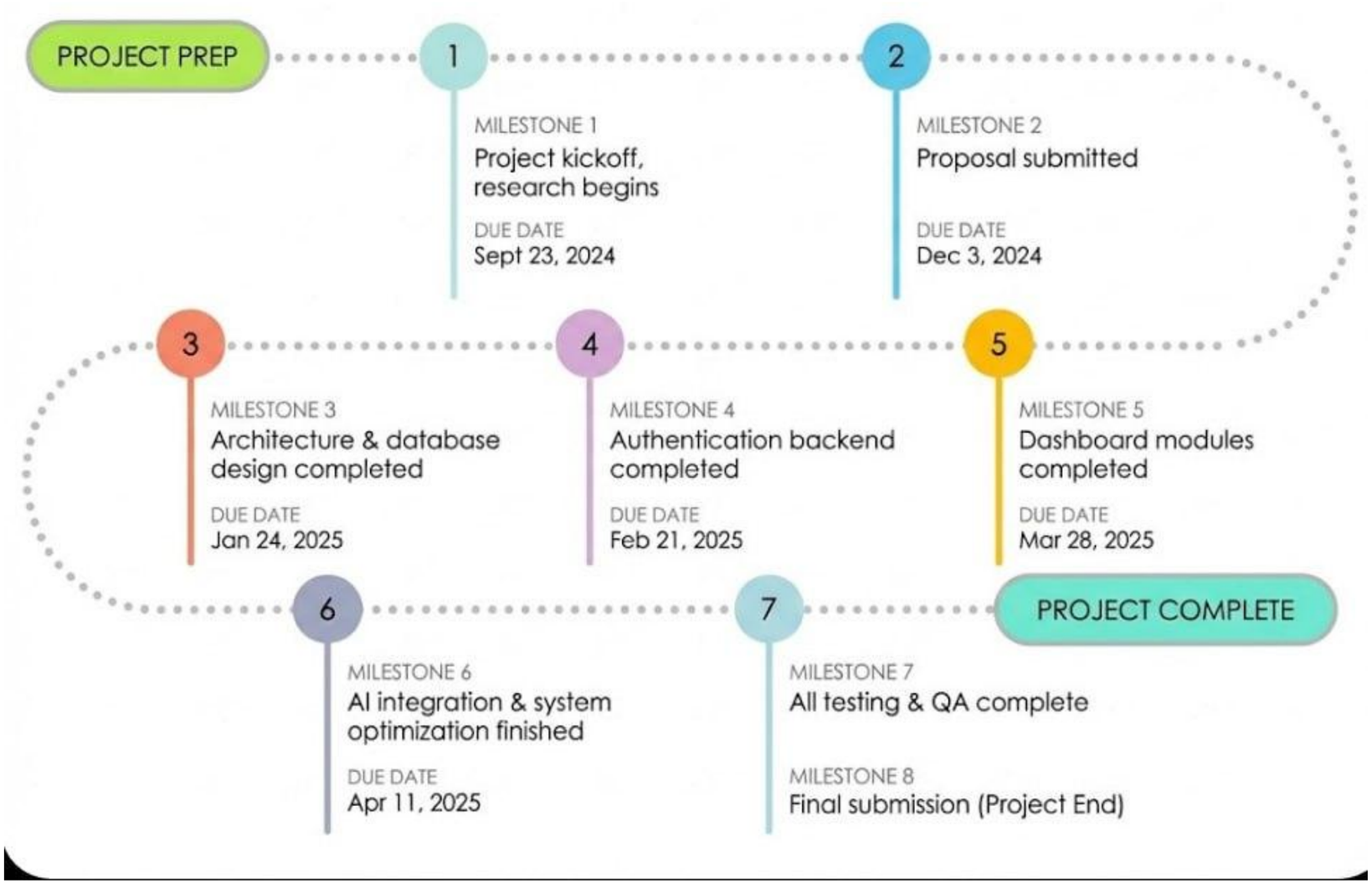

Figure 5: Milestones for Fyp

9. Gantt Chart

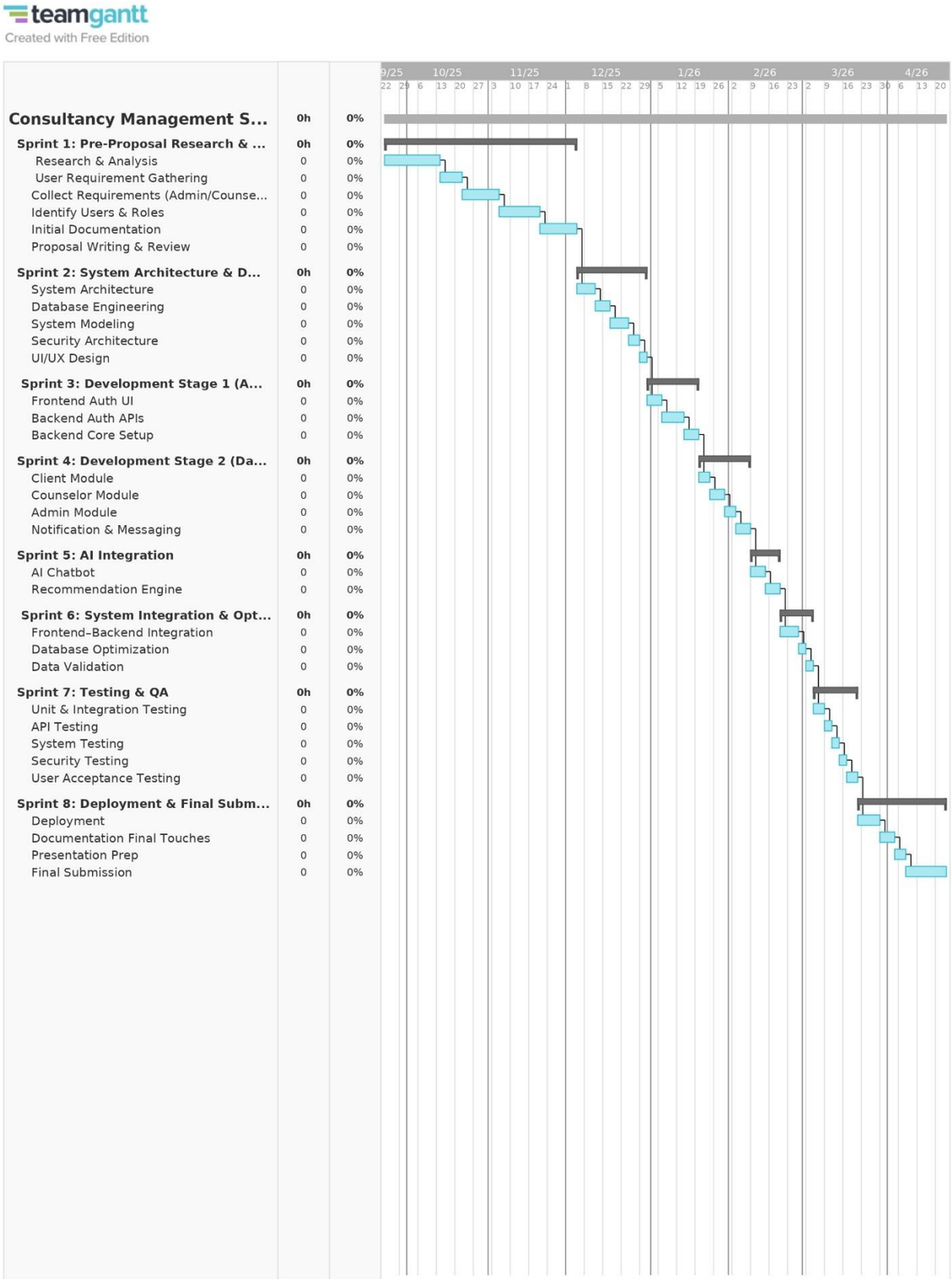

Figure 6: Gantt Chart

For more detail Refer to Appendix 12.1

10. Conclusion

The proposed Consultancy Management System will replace the traditional manual process with a secure, automated, and user-friendly digital platform. Using Next.js, Flask, and MySQL, it enhances efficiency, transparency, and communication between clients, counselors, and administrators. With Agile methodology, well-structured sprints, flexibility, and scalability will be ensured within the project to deliver reliable features at each stage. In this way, this proposal acts as a perfect foundation toward constructing a modern system that upgrades consultancy operations and improves user experiences.

11. References

Anon., n.d. *Agile Methodology.* [Online]
Available at: https://asana.com/resources/agile-methodology
Business360, 2024. *Student Migration and Consultancy Market Analysis in Nepal.* [Online]
Available at: https://www.b360nepal.com/detail/23946/navigating-the-complexities-of-education-consultancy-business
ICEF Monitor, 2025. *Global demand for international education continues to rise.* [Online]
Available at: https://monitor.icef.com
Ministry of Education Nepal, 2025. *Education Consultancy Monitoring and Documentation Standards Report.* [Online]
Available at: https://onespherelaw.com/education-consultancy-license-in-nepal/
Nepali Times, 2025. *Nepal's hard working students overseas.* [Online]
Available at: https://nepalitimes.com/here-now/nepal-s-hard-working-students-overseas
Poidata, 2025. *Registered Consultancy Statistics in Nepal.* [Online]
Available at: https://english.onlinekhabar.com/over-4100-education-consultancies-operating-without-registration-across-nepal.html
Pressman, R. &. M. B., 2020. Software Engineering: A Practitioner's Approach. In: s.l.:McGraw-Hill.
Rising Nepal, 2024. *Education Consultancy Regulation in Nepal.* [Online]
Available at: https://risingnepaldaily.com
Schwaber, K. & Sutherland, J., 2020. *The Scrum Guide: The Definitive Guide to Scrum.* [Online]
Available at: https://scrumguides.org
Sommerville, I., 2016. Software Engineering. In: s.l.:Pearson.

12. Appendix

12.1 Table: Gantt Chart Task Table

Task Name	Start Date	End Date
1. Sprint 1: Pre-Proposal Research & Planning		
1.1 Research & Analysis	09/23/2024	10/11/2024
1.2 User Requirement Gathering	10/14/2024	10/25/2024
Collect Requirements (Admin/Counselor/Client)	10/14/2024	10/18/2024
Identify Users & Roles	10/21/2024	10/25/2024
1.3 Initial Documentation	10/28/2024	11/08/2024
Proposal Writing & Review	11/25/2024	12/03/2024
2. Sprint 2: System Architecture & Design		
2.1 System Architecture	12/09/2024	12/20/2024
2.2 Database Engineering	12/23/2024	01/03/2025
2.3 System Modeling	01/06/2025	01/10/2025
2.4 Security Architecture	01/13/2025	01/17/2025
2.5 UI/UX Design	01/20/2025	01/24/2025
3. Sprint 3: Development Stage 1 (Authentication & Core Backend)		
3.1 Frontend Auth UI	01/27/2025	01/31/2025
3.2 Backend Auth APIs	02/03/2025	02/14/2025
3.3 Backend Core Setup	02/17/2025	02/21/2025
4. Sprint 4: Development Stage 2 (Dashboard Modules)		
4.1 Client Module	02/24/2025	02/28/2025
4.2 Counselor Module	03/03/2025	03/14/2025
4.3 Admin Module	03/17/2025	03/21/2025
4.4 Notification & Messaging	03/24/2025	03/28/2025
5. Sprint 5: AI Integration		
5.1 AI Chatbot	03/31/2025	04/04/2025
5.2 Recommendation Engine	04/07/2025	04/11/2025
6. Sprint 6: System Integration & Optimization		
6.1 Frontend–Backend Integration	03/31/2025	04/04/2025
6.2 Database Optimization	04/07/2025	04/09/2025
6.3 Data Validation	04/10/2025	04/11/2025
7. Sprint 7: Testing & QA		
7.1 Unit & Integration Testing	04/14/2025	04/15/2025
7.2 API Testing	04/16/2025	04/16/2025
7.3 System Testing	04/17/2025	04/17/2025
7.4 Security Testing	04/18/2025	04/18/2025
7.5 User Acceptance Testing	04/14/2025	04/18/2025
8. Sprint 8: Deployment & Final Submission		
8.1 Deployment	04/21/2025	04/21/2025
8.2 Documentation Final Touches	04/21/2025	04/21/2025

8.3 Presentation Prep	04/21/2025	04/21/2025
8.4 Final Submission	04/22/2025	04/22/2025

Table 5: Gantt Chart Task Table

Back to previous

12.2 Work Breakdown Structure (WBS)

A **Work Breakdown Structure (WBS)** is a structured, hierarchical breakdown of a project into smaller, manageable parts.
It divides the entire project into **phases** → **modules** → **tasks**, helping teams clearly understand:

- What work must be completed
- How activities relate to each other
- How to schedule, track, and allocate resources
- How progress will be measured

Sprint No.	Sprint Name	Description
Sprint 1	Project Discovery & Requirements	Focuses on researching consultancy workflows, gathering user requirements, identifying roles, and preparing initial project documentation.
Sprint 2	Planning & Proposal	Preparation of the detailed project proposal, high-level architecture, modules identification, and workflow planning.
Sprint 3	Architecture & Database Design	Designing the system architecture, creating database schema, ERD, constraints, indexing strategy, and system modelling diagrams.
Sprint 4	UI/UX & Wireframing	Designing user interfaces for client, counselor, and admin dashboards; creating wireframes and API blueprints.
Sprint 5	Authentication & Backend Foundation	Developing login/register UI, implementing authentication APIs, JWT, password hashing, permissions, and backend core setup.
Sprint 6	Dashboard Module Development	Building dashboards for client, counselor, and admin; developing APIs; implementing notifications and messaging modules.
Sprint 7	AI Integration	Integrating AI chatbot, recommendation system, and connecting AI APIs with the frontend.
Sprint 8	System Integration & Optimization	Connecting frontend with backend, fixing CORS, role-based redirects, DB optimization, indexing, backups, and data validation.
Sprint 9	Testing & Quality Assurance	Performing frontend/backend testing, API testing, E2E testing, security testing, and user acceptance testing (UAT).

Sprint No.	Sprint Name	Description
Sprint 10	Deployment & Final Submission	Deploying the system, configuring environment variables, hardening security, preparing documentation, slides, and final presentation.

Table 6: Sprints and Description

Back to Previous

12.3 Waterfall Methodology

The waterfall model is a linear and sequential approach to project management. A project is divided into fixed phases in which each phase needs to be completed fully before moving on to the next phase: Requirement → Design → Development → Testing → Deployment.

It works best when the requirements are clearly defined, stable, and unlikely to change.

Characteristics of Waterfall

- Follows a step-by-step procedure
- No return to earlier phases once completed
- Heavy documentation
- Limited customer involvement after the requirement phase
- Less flexible to changes

Back To Previous

12.4 Agile Methodology

Agile is an iterative and flexible approach whereby the project is divided into small cycles known as sprints. Each sprint delivers a working feature or module.

Agile allows for continuous improvement, user feedback, and adaptation throughout the project.

Agile Characteristics

- Development occurs in short sprints
- Regular feedback from users and stakeholders
- Continuous Testing and Integration
- Flexible and adaptable to requirement changes
- Working software is delivered regularly

Back to Previous