Northeastern University

Mechanical and Industrial Engineering Department

'Project Proposal Final Draft'

EMGT 5220 Engineering Project Management

Sec 1- Spring 2025

Project Topic:

NU Mart: Connecting Huskies through Buying and Selling.

Team 4

Dishant Bafna

Rishiraj Budhaale

Chirag Chopda

Sarthak More

Naveen N. N. M.

Sonali Sawant

Bhavya Shah

Point of Contact:

Name: Rishiraj Budhaale

Email: budhaale.r@northeastern.edu

TABLE OF CONTENTS

LETTER OF TRANSMITTAL: ii

EXECUTIVE SUMMARY: iii

1.0 INTRODUCTION: 1

2.0 PURPOSE AND OBJECTIVES: 2

2.1 Purpose: 2

2.2 Objectives: 2

3.0 TECHNICAL OVERVIEW: 3

3.1 Architecture & Technology Stack: 3

3.2 Key Features: 4

3.3 Scalability & Performance: 4

4.0 IMPLEMENTATION PLAN: 5

4.1 Work Breakdown Structure (WBS): 5

4.2 Schedule: 6

4.3 Responsibility Chart: 6

4.4 Stakeholders: 7

5.0 EXECUTION PLAN: 8

5.1 Project Monitoring 8

5.2 Project Control: 9

5.3 Project Auditing: 10

5.4 Project Closure: 12

6.0 FINANCIAL PLAN WITH BUDGET: 13

6.1 High-Level Summary: 13

6.2 Resource allocation 14

7.0 RISK ASSESSMENT MANAGEMENT PLAN: 15

8.0 TEAM CREDENTIALS 19

APPENDICES: 21

APPENDIX A: WORK BREAKDOWN STRUCTURE (WBS) 21

APPENDIX B: SCHEDULE 25

APPENDIX C: RACI MATRIX 32

APPENDIX D: BUDGET JUSTIFICATION 43

APPENDIX E: RESOURCE ALLOCATION 47

REFERENCES: 48

LETTER OF TRANSMITTAL:

Date: 04/18/2025

Prof. Himlona Palikhe
Northeastern University
College of Engineering
360 Huntington Avenue
Boston, MA 02115

Dear Professor, Palikhe,

Enclosed is the submission of Team 4's project proposal for NU Mart: Connecting Huskies through Buying and Selling, created for EMGT 5220: Engineering Project Management (Spring 2025, Section 1).

Our proposal tackles typical challenges that Northeastern University students encounter when purchasing and selling necessary items on public platforms, such as safety issues, fraud, and hassle. NU Mart provides a safe and easy-to-use online marketplace tailored specifically for the NU community. It employs NU credentials for user verification, fosters sustainability through promoting item reuse, and includes features such as integrated messaging and structured listings. The attached document details the project's objectives, technical methods, execution timeline, budget plan, and strategies for minimizing risks.

We, the Team 4 members, genuinely value your support and insights during this course. We trust this proposal showcases our dedication to providing a platform that improves campus life, safety, and sustainability.

Thank you for your time and consideration.

Sincerely,

Team 4
(Rishiraj Budhaale, Chirag Chopda, Bhavya Shah, Naveen N.N.M., Sarthak More, Sonali Sawant, Dishant Bafna)
EMGT 5220 | Spring 2025 | Section 1
College of Engineering
Northeastern University

EXECUTIVE SUMMARY:

The Northeastern University team proposes a strategic initiative to create a digital web-based marketplace solely for Northeastern students to buy, sell, and trade essential items within their university community. This secure, student-only marketplace is designed to facilitate peer-to-peer transactions, particularly during peak times such as move-in and move-out periods.

Public marketplace platforms like Facebook Marketplace, WhatsApp groups, and Craigslist expose students to scams, unverified users, and irrelevant listings. NU Mart addresses these issues by implementing NU Email Verification, smart search filters, and in-app messaging. The platform promotes a straightforward, safe, and efficient user experience while encouraging sustainability through the reuse and recycling of student resources.

The project follows a five-phase Work Breakdown Structure - Initiation, Planning, Execution, Monitoring and Control, and Closure- accompanied by a proper schedule, RACI Matrix, and Stakeholder Mapping to ensure systematic application development. The proposed platforms incorporate frontend/backend development, authentication, risk tracking with mitigation strategies, and secure infrastructure components.

The estimated project cost is $204,328.12, with over 84% allocated to skilled labour. A cost-benefit analysis predicts a return of $250,000 over five years, resulting in a cost-benefit ratio of 1.23. This underscores the platform's long-term value and feasibility of operation.

Overall, NU Mart provides Northeastern students with a reliable, campus-exclusive platform that enhances everyday student life while contributing to cost savings, environmental sustainability, and community engagement.

1.0 INTRODUCTION:

NU Mart is a proposed web-based platform designed exclusively for Northeastern University students on the Boston campus, aimed at simplifying the buying, selling, and exchanging of essential items and academic materials. Students often look for affordable used goods, especially during move-in and move-out periods, but face several challenges with current public platforms like Facebook Marketplace and WhatsApp, including scams, irrelevant listings, and coordination difficulties.

NU Mart addresses these issues by offering a dedicated, student-only marketplace that ensures security, ease of use, and efficiency. Key features of the platform include two-factor authentication using NU credentials (NU email, NUID, Duo), a clean and intuitive user interface, and a built-in messaging system to facilitate secure communication without relying on third-party apps. Listings will be well-organized by categories with powerful search and filter tools to streamline product discovery.

The project deliverables include the development of a functional and interactive website, secure login and authentication, advanced listing and search capabilities, a built-in chat system, location-based safety measures, an admin dashboard for moderation, and a helpdesk for user support. The development plan spans multiple phases: requirement analysis, UI/UX prototyping, core development, security enhancement, testing, deployment, and ongoing maintenance. Each phase is structured to ensure a seamless rollout, driven by continuous feedback and optimization.

Looking ahead, NU Mart aims to evolve into a comprehensive digital ecosystem for the student community. Future enhancements may include AI-powered product recommendations, integrated payment options for smoother transactions, real-time notifications, and even delivery support within the campus. The platform will also support designated on-campus pickup zones to ensure safe and convenient exchanges. With constant updates based on student feedback, NU Mart will grow into a trusted and essential part of campus life, connecting students through a reliable and secure platform tailored to their unique needs.

2.0 PURPOSE AND OBJECTIVES:

2.1 Purpose:

The purpose of this project is to provide a trusted, convenient, and affordable way for the NU community to exchange items.

By keeping the platform exclusive to Northeastern students, they can feel more secure buying and selling without the risks of dealing with strangers on public marketplaces. This also means access to cheaper, used items from fellow Huskies, helping students save money while promoting sustainability through reusing and recycling within the community.

The website will make transactions simple and seamless, with features like easy listing, secure in-app messaging, and smart search filters. Overall, it is designed to connect Northeastern students in a way that makes buying and selling safer, easier, and more reliable than ever before.

2.2 Objectives:

- **Create a Secure buy-and-sell marketplace:** Ensure only Northeastern students can access the platform through university authentication, creating a safe and trusted environment.
- **Make Buying & Selling Easy:** Develop a simple and intuitive interface where users can quickly list, browse, and purchase items without hassle.
- **Enable Safe & Efficient Transactions:** Integrate features like in-app messaging, smart search filters, and suggested meet-up locations to streamline the buying and selling process.
- **Ensure Scalability & Reliability:** Design the website with a scalable architecture to handle growth, new features, and potential expansion across NU campuses.

3.0 TECHNICAL OVERVIEW:

The technical overview offers a comprehensive insight into the platform's construction using modern web technologies, guaranteeing smooth authentication, optimal data management, and robust availability.

The development emphasizes security, real-time engagement, and responsiveness using cloud-hosting and secure backend services. The project is divided into the following to provide more details about the various components involved.

3.1 Architecture & Technology Stack:

- **Frontend:** Will be built using React.js (or Next.js for server-side rendering) to create a fast, responsive, and intuitive user interface. Tailwind CSS or Bootstrap will be used for a modern and mobile-friendly design.
- **Backend:** The backend phase covers cloud and database setup, secure user authentication, and core features like profile APIs, real-time encrypted messaging, and service integrations.
- **Database:** Uses PostgreSQL, a relational database optimized for handling structured data, including user profiles, listings, and transactions. Indexing and optimizing queries will enhance search performance.
- **Authentication:** NU Mart will integrate secure third-party payment gateways like Zelle, PayPal, etc., using API/SDKs, ensuring encrypted transactions tied to verified NU student accounts via SSO.
- **Hosting & Deployment:** The application will be hosted on AWS (EC2, RDS, S3) or Firebase, ensuring scalability, security, and automatic failover mechanisms.

3.2 Key Features:

- **Secure Authentication**: NU-exclusive university credentials login prevents unauthorized access, ensuring that only verified students can use the platform. Role-based access control (RBAC) will enforce user permissions.
- **Product Listings & Search:** Users can list items with images, descriptions, and pricing. A robust search and filtering system will enable quick item discovery based on categories, price range, and item condition. AI-based recommendations can suggest relevant listings based on user preferences.
- **In-App Messaging:** A real-time messaging system powered by WebSocket (Socket.io) will allow buyers and sellers to communicate securely without external apps. Messages will be encrypted to protect user privacy.
- **Moderation & Security:** The platform will include automated fraud detection, user reporting, and admin moderation tools. Machine learning-based monitoring can flag suspicious activity or fraudulent listings.
- **Designated Meet-up Locations:** The system will include campus-based meet-up location suggestions to enhance transaction security, reducing risks associated with in-person exchanges.

3.3 Scalability & Performance:

- **Cloud Deployment:** Uses serverless architecture (AWS Lambda or Firebase Functions) to scale dynamically based on user activity, ensuring high availability without manual intervention.
- **CDN Integration:** CloudFront or Cloudflare will be implemented for content caching, minimizing load times, and improving page speed, especially for images and product listings.
- **CI/CD Pipeline:** GitHub Actions or Jenkins will handle automated testing, build processes, and seamless deployments, reducing downtime and ensuring stable updates.

This technical architecture ensures that NU Mart remains secure, scalable, and efficient, providing a seamless user experience while handling thousands of concurrent transactions and maintaining data integrity.

4.0 IMPLEMENTATION PLAN:

4.1 Work Breakdown Structure (WBS):

The Work Breakdown Structure (WBS) is a hierarchical breakdown of the required tasks to successfully develop and deploy the project "NU Mart: Connecting Huskies through Buying and Selling." The project ensures well-defined task allocation and efficient resource management with timeline adherence. The project is divided into five major phases:

- **Project Initiation Phase:** This phase creates a foundation for the development of NU Mart by involving activities such as project scope, objectives, and key deliverables. Budget planning is initiated, which is then followed by outlining the web objectives and identifying the core functionalities while setting up the risk management and compliance strategies, aligning with the cybersecurity standards.

- **Project Planning:** In this phase, the project team will conduct in-depth market research to understand existing student platforms and their limitations. Here, the functional and non-functional requirements would be defined along with the system architecture and workflows. Additionally, it includes selecting the right tech stack, finalizing cloud infrastructure, and outlining security and API frameworks.

- **Project Execution Phase:** It is the crucial phase of the project as it involves the development of the NU Mart web platform by implementing backend development, authentication system, by integration (NEU's SSO), then feature implementation consists of building API, user profile management and push notifications. Later, the frontend development will include the development of key UI components like navigation filters and product listings. Then, by integrating a payment gateway, the testing would be done to ensure system stability and resilience.

- **Project Monitoring and Control:** Throughout the execution of the platform, system testing will validate key functions such as login, product browsing and messaging. There will be continuous risk monitoring, progress tracking and reporting to ensure actual progress aligns with planned milestones.

- **Project Closure Phase**: The last part of the project involves the deployment of the platform within the university community and monitoring the post-launch maintenance. To ensure the project's growth, there will be a team to look out for continuous improvements and track system performance while documenting the project's learnings.

The detailed WBS of the project is attached in Appendix A.

4.2 Schedule:

The NU Mart project is planned to 1828 hours, starting with the project initiation phase of 168 hours, followed by the project planning phase of 215 hours, then comes the project execution phase, which would take 1110 hours, further the project monitoring and control phase would take 140 hours and lastly the project closure phase which would take 195 hours.

The detailed schedule of the project is attached in Appendix B.

4.3 Responsibility Chart:

The RACI matrix in the Appendix is for the Nu Mart web application development team. It defines who is **Responsible (R), Accountable (A), Consulted (C), and Informed (I)** for each task.

This matrix will help the team collaborate effectively by clarifying responsibilities, reducing confusion, and ensuring accountability throughout the NU Mart project. The following responsibility proportions have been considered for the RACI matrix in this project.

Legend: R = 100 %, A = 30%, C = 15 % and I = 5 %.

The detailed responsibility chart of the project is attached in Appendix C.

4.4 Stakeholders:

The following list provides a brief overview of the interested parties mentioned in this report throughout this report

1. Project Team

- Project Manager
- Product Managers
- Development Team
  - Frontend Developer
  - Backend Developer
  - QA Tester
- UI/UX Designer
- Cybersecurity Specialist
- Database Engineer
- Business Analyst
- Marketing and Outreach

2. University Administration & IT Department

- University IT Department
- Administration & Legal Team

3. End Users (Northeastern Students)

- Buyers & Sellers

4. Vendors & Third-Party Services

- Hosting & Cloud Service Providers
- Apple Pay, Google Pay, Zelle, PayPal, Bank Transfers

5.0 EXECUTION PLAN:

5.1 Project Monitoring

The purpose of project monitoring for NU Mart is to ensure a successful launch of the platform by defining the scope, budget and timeline requirements while maintaining the quality standards and user satisfaction. Using foundational tools like Work Breakdown Structure (WBS) and RACI Matrix, the team aims to implement a structured reporting system to monitor the following components throughout the project lifecycle.

1. Cost Monitoring:

The financial tracking is to ensure that the NU Marts are operating within the budget of $204,328.12. For this, the monitoring includes,

- Tracking labor costs, software tool expenses, hosting services and vendor fees.
- Comparing planned versus actual expenditures
- Generating cost variance reports for early identification and mitigation of overspending risks.

2. Schedule Monitoring:

The crucial part of the project is the schedule, using tools like Jira and Click Up will be used to manage the weekly progress and ensure timely completion of major tasks, such as:

- Backend and Frontend development milestones.
- Completion of user authentication and secure payment integration.
- Testing phases and issue resolution sprints.
- Final deployment and post-launch activities.

3. Staff and Resource Monitoring:

During the project execution phase, the monitoring of the human resource component will involve:

- Role assignment and task ownership clarity (based on the RACI Matrix).
- Team availability, workload balance and task distribution.
- Skill utilization and identifying the need for cross-training or support.

Any performance or resource bottlenecks will be addressed in daily sprint retrospectives.

4. Quality Monitoring:

To ensure that the NU Mart platform meets reliability, security and usability expectations, the quality will be monitored using:

- Adherence to defined UI/UX Standards.
- Automated unit and integration testing.
- Regular security assessments and vulnerability scans.
- Peer reviews and sprint-end QA checks.

5.2 Project Control:

Project control mechanisms ensure that the project stays true to its objectives. For the NU Mart team, structured meetings and assigned responsibilities based on management sectors will be the focus points in ensuring efficient Project Control:

Key Factor	**Responsibility**
Cost	Financial/Project Manager
Schedule	Project Manager
Quality	QA & Product Team
Scope Changes	Product Manager & Project Manager with Team Review

Table 1.0 Project Control Chart

Corrective actions will be implemented immediately if any deliverable deviates from the plan. The project will be at a Stop phase, where the deviated deliverable is brought back on track to keep the project moving forward. This project will basically follow the Go/NoGo Project Control protocol. The key factors are the responsibilities of assigned employees who are responsible for stopping progress if there is a deviation in deliverables. Regular meetings will be held to analyse deviations and make 'Stop' decisions based on Project progress. Tools like Jira and ClickUp will be used to track task velocity, sprint progress, and issue tracking.

5.3 Project Auditing:

Project audits are crucial to ensure NU Mart stays on track in terms of scope, timeline, and cost while consistently delivering high-quality outputs aligned with students' needs. These audits not only assess the progress of development activities but also reflect on team coordination and process effectiveness. For NU Mart, audits are scheduled upon the completion of key project phases to maintain alignment with strategic objectives. Each audit encompasses the three key stages: preparation, research and review, and report development. Below is an outline of the audit framework applied for this project:

5.3.1 Technical Overview:

During the development of NU Mart, a technical audit is conducted to evaluate all critical components and ensure architectural robustness and compliance with cybersecurity norms. This includes:

- Validating the defined functional and non-functional requirements.
- Reviewing UI/UX prototypes and design implementation.
- Inspecting backend development (Nodejs, PostgreSQL, REST API services).
- Verifying authentication and authorization logic using NU credentials and 2FA.
- Evaluating the integration of key modules, like in-app messaging, product listing, and filtering.
- Reviewing moderation and security tools.
- Assessing the deployment architecture hosted on AWS/Firebase for scalability and failover.

5.3.2 Project Status Overview:

As the project progresses, the audit team tracks deviations in schedule, budget, and scope using tools like Jira and Click Up. Status audits are performed after the following milestones:

- Requirement analysis and planning phase
- UI/UX design completion
- Backend and frontend feature integration
- Testing and debugging phase
- Payment gateway and security compliance completion
- Final deployment of the platform
- Post-launch marketing and feedback assimilation
- Completion of continuous improvement rollout

These audits enable timely corrections and effective communication among stakeholders

5.3.3 Final Audit:

A conclusive audit is conducted at the end of the project lifecycle to validate the comprehensive performance of the NU Mart initiative. The final audit focuses on:

- Adherence to the estimated budget ($204,328.12) and resource allocation.
- Milestone completion in accordance with the project schedule (10-12 months).
- Fulfilment of all technical deliverables, including secure authentication, chat features, and listing management.
- Measured satisfaction among students (end-users) via feedback tools post-launch.
- Verification of usability, reliability, and platform scalability.
- Documentation of professional collaboration and individual contributions.
- Review of the accuracy and completeness of the deployed web platform.
- Evaluation of NU Mart's potential to scale and incorporate features like AI pricing, payment integration, and blockchain.
- Through structured auditing, NU Mart maintains transparency, project health, and continuous stakeholder confidence while supporting the vision of a secure, student-centered marketplace.

5.4 Project Closure:

The NU Mart project will conclude successfully upon meeting all defined objectives within the allocated budget and projected timeline. This initiative will result in the development of a secure, web-based marketplace designed exclusively for Northeastern University students, simplifying peer-to-peer transactions during high-demand periods such as move-ins and move-outs.

All key deliverables—including secure NU credential-based login, categorized product listings, in-app messaging, administrative moderation tools, and a responsive user interface—will be completed and delivered as per the project scope. The platform will undergo rigorous testing to ensure it meets standards for usability, performance, and security.

The project is expected to adhere to the planned schedule of 1828 Hours and remain within the approved budget of $204,328.12. Stakeholder reviews and user acceptance testing will be conducted to validate that the platform meets user expectations and functional requirements.

Comprehensive documentation will be compiled, covering system architecture, test reports, user guides, and strategies for future improvements. A formal knowledge transfer will be conducted to the university's IT team to support post-launch operations and maintenance. [2][3]

Post-deployment performance monitoring and feedback collection will be established to identify enhancement opportunities. The platform will be positioned for future integration of features such as AI-powered product recommendations and secure payment gateways.

The successful closure of the NU Mart project will mark a significant milestone in digital innovation on campus. It will provide a sustainable, scalable solution that enhances student engagement and fosters a safe, connected campus community.

6.0 FINANCIAL PLAN WITH BUDGET:

6.1 High-Level Summary:

This project focuses on developing a peer-to-peer marketplace app that enables Northeastern University students to buy, sell, and trade items such as electronics, textbooks, and furniture within the campus community. The app aims to provide a secure, convenient, and student-exclusive platform, reducing dependence on external marketplaces. However, this initiative requires a well-structured financial plan to ensure successful development and implementation. A budget has been established and will be tracked throughout the project lifecycle. Below is the project's Budget Summary—refer to Appendix D for the complete budget justification.[1]

Budget Summary	
Resources	**Total Cost**
Labor	172135.86
Materials & Equipment	16825.00
Miscellaneous	2000.00
7% Contingency	13367.26
TOTAL	**$204,328.12**

Table 2.0 Budget Summary of Resources

Materials and Equipment	
Software tools and licenses (Figma, API services, etc.)	5000.00
Cloud hosting and database costs	9825.00
Development hardware	2000.00
TOTAL	**$16,825.00**

Table 2.1 Budget Summary of Materials and Equipment

Hourly Wage as per the Bureau of Labour Statistics	**Wage per hour ($)**
Project Manager	57.1
Product Manager (PM)	57.1
Business Analyst	50.33
UI/UX Designer	57.1
Frontend Developer	57.1
Backend Developer	57.1
Database Engineer	57.1
Cybersecurity Specialist	57.1
QA Tester	57.1
Marketing/Community Manager	50.33

Table 2.2 Hourly Wage as per the Bureau of Labour Statistics

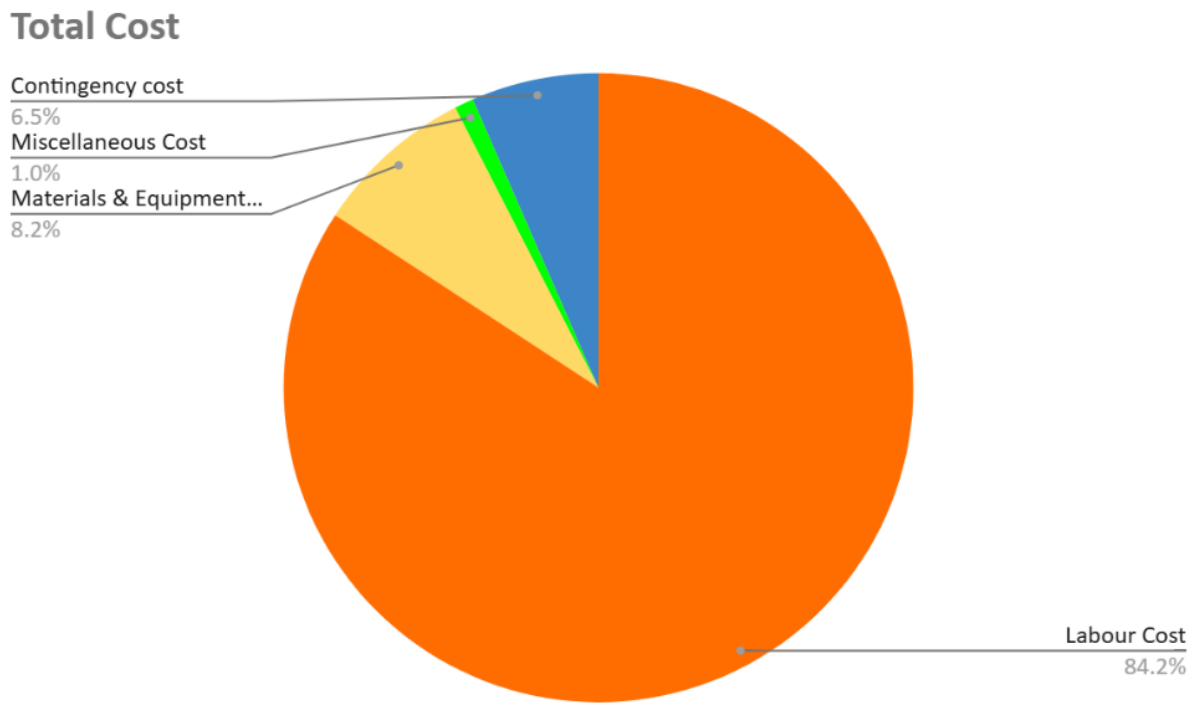


Fig. 1.0 Pie Chart of Budget Summary

Economic Analysis: Cost–Benefit Ratio
Northeastern is expected to save between $40,000 and $60,000 annually.
CBR = Total Expected Benefits for 5 years/ Total Project Cost
= $ 204,328.12/ $250,000 ~ 1.23

6.2 Resource allocation

The estimated total budget for the development of the NU Mart web platform is $204,328.12, covering a total of 1,828 planned project hours. The project is scheduled to start on April 28, 2025, and includes key allocations for labour, equipment, and contingency.

This budget supports essential roles such as project management, UI/UX design, frontend and backend development, cybersecurity, quality assurance, and business analysis. It also includes allocations for materials, miscellaneous expenses, and a 7% contingency buffer to accommodate unforeseen changes.

Please note that these figures are estimates, and actual costs may vary based on the project's specific requirements, the experience level of the professionals involved, and other factors unique to the project.

7.0 RISK ASSESSMENT MANAGEMENT PLAN:

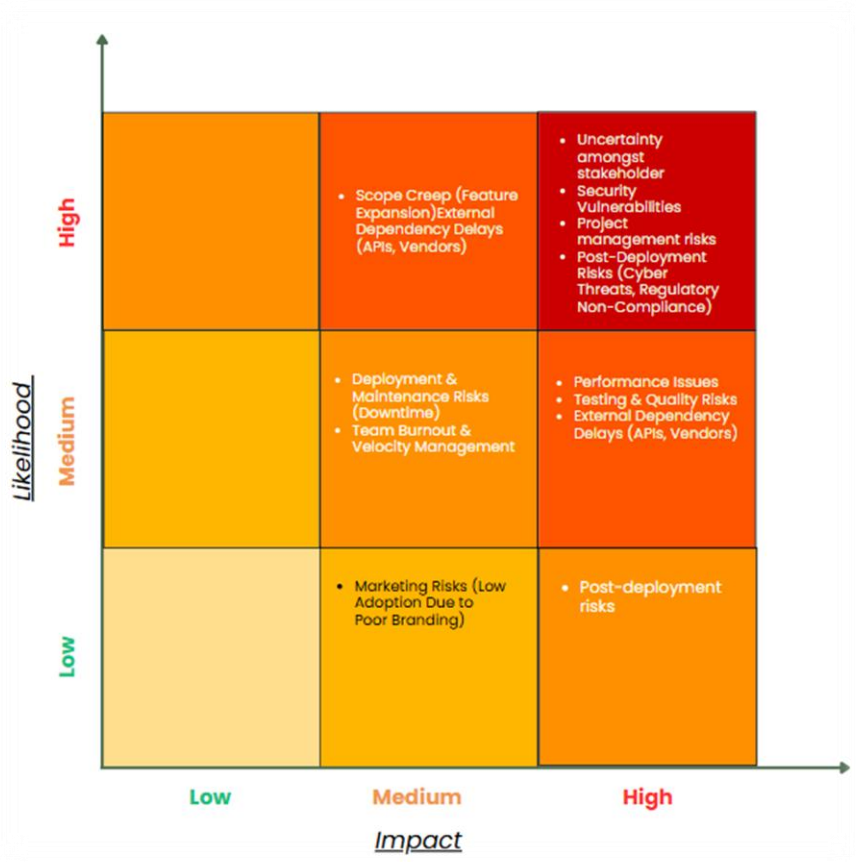


Fig. 2.0 Risk Matrix

No.	Risk Description	Likelihood	Impact	Response	Action Plan
1	Uncertainty among stakeholders	High	High	Mitigate	Hold bi-weekly stakeholder meetings, distribute sprint demo updates, engage key stakeholders in backlog grooming, and provide a data-backed decision rationale.
2	Incompatible Technologies	Medium	Medium	Mitigate	Perform technical evaluations before sprint planning, ensure API-first design, and guarantee modular system adaptability.
3	Performance Issues	Medium	High	Mitigate	Implement load balancing, optimize database queries, conduct regular performance testing, and introduce automated monitoring.
4	Security Vulnerabilities	High	High	Mitigate	Conduct automated security scans in CI/CD, perform penetration testing, enforce secure coding practices, and ensure timely patching.
5	Project Management Risks	High	High	Mitigate	Use Agile tracking tools (JIRA, Click Up), conduct daily stand-ups, hold retrospectives, and monitor sprint burndown rates.

No.	Risk Description	Likelihood	Impact	Response	Action Plan
6	Testing & Quality Risks	Medium	High	Mitigate	Implement a structured testing plan, conduct unit & integration tests, perform User Acceptance Testing (UAT), and maintain an active bug tracking system.
7	Scope Creep (Feature Expansion)	High	Medium	Control	Define a clear MVP scope, enforce a change management process, evaluate feature requests based on impact & priority, and use MoSCoW prioritization.
8	Deployment & Maintenance Risks (Downtime)	Medium	Medium	Mitigate	Use blue-green deployments, conduct pre-launch testing, establish rollback mechanisms, and monitor post-deployment uptime & performance.
9	Marketing Risks (Low Adoption Due to Poor Branding)	Low	Medium	Mitigate	Conduct early user testing, implement feedback loops, track user behavior analytics, adjust campaigns based on metrics, and actively collect customer input.
10	Post-Deployment Risks (Cyber Threats, Regulatory Non-Compliance)	High	High	Mitigate	Implement real-time security monitoring, conduct periodic compliance audits, quickly address vulnerabilities, and ensure fast bug fixes.

No.	Risk Description	Likelihood	Impact	Response	Action Plan
11	Team Burnout & Velocity Management	Medium	Medium	Mitigate	Track sprint velocity, encourage sustainable pacing, enforce work-life balance, and use retrospectives to address team concerns.
12	External Dependency Delays (3rd party APIs, Vendors)	Medium	High	Mitigate	Identify critical dependencies early, maintain buffer sprints, plan alternative integration paths, and establish contingency measures.

Table 3.0 Risk Assessment Summary

8.0 TEAM CREDENTIALS

1. Dishant Bafna

Dishant is a graduate student pursuing a Master's degree in Industrial Engineering at Northeastern University. He completed his undergraduate in Mechanical Engineering and has worked as a Manufacturing Quality Engineer at Jaguar Land Rover. He gained significant experience in the manufacturing and supply chain sector, working on various aspects including quality assurance, statistical process planning, root-cause analysis, and product audits. Additionally, he worked as a Production and Supply Chain Engineer for a startup, optimizing production processes, advancing production plans, and working on reducing procurement costs through vendor management and logistics efficiency. He also previously spent time at a supply chain startup he co-founded, running all operations from procurement to warehousing, sales, accounting, and logistics.

2. Rishiraj Budhaale

Rishiraj is currently pursuing an MS in Engineering Management at Northeastern University. He holds a bachelor's degree in Mechanical Engineering and has successfully led process improvement projects, implemented lean methodologies, and optimized production workflows to enhance efficiency, reduce defects, and improve resource utilization. He believes that Project management plays a critical role in manufacturing units by ensuring efficient resource utilization, process standardization, and timely project execution. He is excited to explore various concepts of PM in detail.

3. Chirag Chopda

Chirag is currently pursuing a master's degree in Engineering Management at Northeastern University. He holds a bachelor's degree in Mechanical Engineering and has professional experience in operations and manufacturing. As an Operations Manager at Essar Power Limited, he was responsible for plant manufacturing operations, ensuring efficiency and reliability. He excels in collaborative environments, integrating engineering principles with management strategies to drive innovation and process improvement.

4. Sarthak More

Sarthak is currently a student at Northeastern University pursuing a Master of Science in Engineering Management. He completed his undergraduate studies in Information Technology. During his bachelor's degree he completed various projects focusing on application development and data analysis, collaborating on both solo and team endeavors. These experiences enabled him to cultivate a systematic approach to addressing challenges and enhance his skills in data interpretation and analytical thinking.

5. Naveen N. N. M

With a work experience of 2 years as a SaaS implementation consultant, catering to large-scale CPG clients, Naveen is currently pursuing the second semester of his Master of Science in Engineering Management degree at Northeastern University. He incorporates his experience gained in developing product roadmaps and project delivery management into every aspect of work. He is a team player, and we are all looking forward to working with him.

6. Sonali Sawant

Sonali is an Industrial Engineering Graduate student at Northeastern University. Her background combines engineering, technology, and project management. She has 5 years of experience in industrial engineering and ERP consulting. In her previous roles, she worked on cost reduction and productivity improvement projects in a manufacturing company. She also worked on SAP agile projects as a Scrum Master, leading teams to complete projects successfully and on time.

7. Bhavya Shah

Bhavya is currently in his second semester in the College of Engineering pursuing a Master of Science in Engineering Management at Northeastern University in Boston. He has a background in Aerospace Engineering with an internship completed at Air India Engineering Services Limited where he served as a Flight Operations Intern. He is particularly interested in applying his technical expertise into project management principles to solve real-world problems. This project allowed him to integrate analytical skills with structured project management approaches.

APPENDICES:

APPENDIX A: WORK BREAKDOWN STRUCTURE (WBS)

NU Mart: Connecting Huskies through Buying and Selling.	
ID	**Task**
1	**1.0 Project Initiation Phase**
2	1.1 Conduct Project Kick-off Meeting
3	1.1.1 Define Objectives, Scope, and Constraints
4	1.1.2 Establish Project Timeline and Milestones
5	1.1.3 Identify Key Stakeholders
6	1.1.4 Draft Budget for the Project
7	1.2 Define Web Objectives
8	1.2.1 Establishing well-defined UI/design features
9	1.2.2 Focus on core features such as database management, payment gateways, etc.
10	1.3 Risk Management and Compliance
11	1.3.1 Identify potential risk and mitigation strategies
12	1.3.2 Ensure Compliance with Data Protection Regulations
13	1.3.3 Incorporate Cyber-security compliance and risk measures aligning with industry standards
14	**2.0 Project Planning Phase**
15	2.1 Conduct Market Research
16	2.1.1 Analyze existing platforms used by students
17	2.1.2 Identify key challenges and areas of improvement
18	2.2 Defining Requirements
19	2.2.1 Define functional and non-functional requirements
20	2.2.2 Define system architecture and workflow
21	2.3 Tech Stacks Selection
22	2.3.1 Research and finalize frontend and backend technologies
23	2.3.2 Define hosting and cloud infrastructure requirements
24	2.3.3 Identify security framework and API requirements
25	**3.0 Project Execution Phase**
26	3.1 Backend Development Phase
27	3.1.1 Choose Cloud Platform

28	3.1.1.1 Select and implement cloud platform services (AWS & Firebase platform)
29	3.1.2 Choose DBS systems
30	3.1.2.1 Setup both MySQL and PostgreSQL as Database systems
31	3.2 Implement authentication and user verification
32	3.2.1 Create user authentication system
33	3.2.1.1 Design login, logout and password reset functionalities
34	3.2.2 Coordinate with NEU tech services for SSO integration details
35	3.2.3 Check the website's compatibility, user data security and login procedures
36	3.3 Implement main features
37	3.3.1 Create main backend features
38	3.3.1.1 Integrate products and profile database
39	3.3.1.2 Create APIs for user profile CRUD operations
40	3.3.2 Develop messaging system between buyer and seller
41	3.3.2.1 Use Firebase for real-time messaging
42	3.3.2.2 Ensure all messages are end-to-end encrypted
43	3.3.3 Integrate key backend services
44	3.3.3.1 Use Firebase cloud messaging for push notification
45	3.3.3.2 Plugin system support
46	3.4 Frontend Development Phase
47	3.4.1 UI/UX Prototyping and Design
48	3.4.1.1 Develop wireframes and UI/UX prototypes
49	3.4.2 Core UI Components
50	3.4.2.1 Create product cards with images, titles, prices, and filters
51	3.4.2.2 Create standard font size, color, and UI elements
52	3.4.2.3 Ensure button display hover, press and disabled states
53	3.4.2.4 Build elements like navigation and search bars, filters, recommendations, etc.
54	3.4.3 Building user profile interface
55	3.4.3.1 Profile Page layout (Buyer/Seller view)
56	3.4.3.2 View Seller Profiles (Product page -> Seller info)
57	3.4.3.3 View Buyer Profiles (Order -> Buyer info)
58	3.4.3.4 Display ratings and reviews
59	3.4.4 Identify required third-party APIs
60	3.4.4.1 Payment processing (Apple Pay, Pay Pal, Credit Card, Debit Card, Net Banking)
61	3.4.4.2 Evaluate ease of API integration, secure gateways, and connections
62	3.5 Integration of API and Testing Phase
63	3.5.1 Integrate Frontend and Backend functionalities
64	3.5.1.1 Implementing and connecting frontend elements to backend services through APIs

65	3.5.1.1.1 Test API functionalities
66	3.5.1.1.2 Testing API functionalities for CRUD operations in user profiles
67	3.5.1.1.3 Verifcation of login, logout, and token refresh APIs
68	3.5.2 Maintain data flow and functionalities
69	3.5.2.1 Ensure data flows and matches backend database in frontend
70	3.5.3 Conduct user testing and gather feedback
71	3.5.3.1 Ensure API response times are responsive
72	3.5.3.2 Verify each component's feature and function
73	3.5.4 Refine and finalize UI/UX based on testing
74	3.5.4.1 Ensure smooth data transfer from user sign up to profile creation
75	3.6 Test Third-party Integrations
76	3.6.1 Secure external API integration
77	3.7 Security and Fraud Prevention
78	3.7.1 Implement secure transaction mechanisms
79	3.7.2 Payment gateway integration and plan transaction
80	3.7.3 Develop admin panel for moderation
81	3.8 Testing and Debugging
82	3.8.1 Conduct unit testing for individual modules
83	3.8.2 Perform system integration and stress testing
84	3.8.3 Identify and resolve bugs and performance issues
85	3.8.4 Conduct penetration testing and security
86	**4.0 Project Monitoring and Control**
87	4.1 Perform System Testing
88	4.1.1 Verify essential features like user login, product catalogue and messaging system
89	4.1.1.1 Ensure smooth log-in, log-out and catalogue browsing
90	4.1.1.2 Ensure seamless search experience and filter drilldown
91	4.2 Risk Monitoring and Control
92	4.2.1 Identify new risk that emerge during development
93	4.2.2 Implement contingency actions for triggered risks
94	4.2.3 Update risk register for effectiveness of mitigation
95	4.3 Progress tracking and reporting
96	4.3.1 Monitor actual progress against planned scheduled
97	4.3.2 Track resource utilization and allocation Implement contingency actions for triggered risks
98	**5.0 Project Closure Phase**
99	5.1 Deployment and Launch
100	5.1.1 Deploy platform on secure web server
101	5.1.2 Develop user guides and onboarding materials

102	5.1.3 Promote platform within the university community
103	5.2 Post Launch Monitoring
104	5.2.1 Monitor system performance and user feedback
105	5.2.2 Implement continuous improvements and updates
106	5.2.3 Document project learning and closure reports

Table 4.0 Work Breakdown Structure

APPENDIX B: SCHEDULE

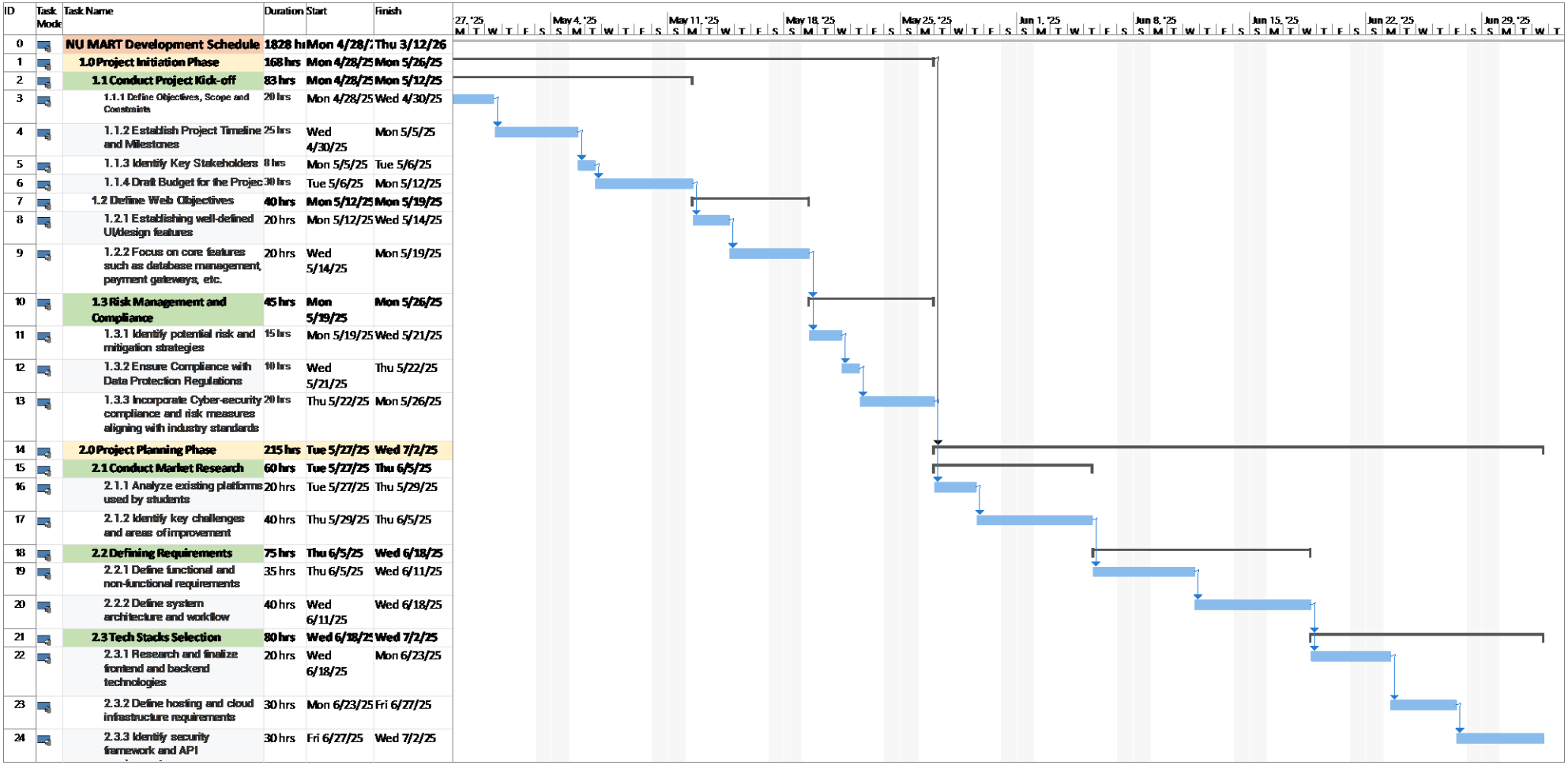
NU Mart: Connecting Huskies through Buying and Selling.		
ID	**Task**	**Duration [Hours]**
1	**1.0 Project Initiation Phase**	**168**
2	1.1 Conduct Project Kick-off Meeting	83
3	1.1.1 Define Objectives, Scope, and Constraints	20
4	1.1.2 Establish Project Timeline and Milestones	25
5	1.1.3 Identify Key Stakeholders	8
6	1.1.4 Draft Budget for the Project	30
7	1.2 Define Web Objectives	40
8	1.2.1 Establishing well-defined UI/design features	20
9	1.2.2 Focus on core features such as database management, payment gateways, etc.	20
10	1.3 Risk Management and Compliance	45
11	1.3.1 Identify potential risk and mitigation strategies	15
12	1.3.2 Ensure Compliance with Data Protection Regulations	10
13	1.3.3 Incorporate Cyber-security compliance and risk measures aligning with industry standards	20
14	**2.0 Project Planning Phase**	**215**
15	2.1 Conduct Market Research	60
16	2.1.1 Analyze existing platforms used by students	20
17	2.1.2 Identify key challenges and areas of improvement	40
18	2.2 Defining Requirements	75
19	2.2.1 Define functional and non-functional requirements	35
20	2.2.2 Define system architecture and workflow	40
21	2.3 Tech Stacks Selection	80
22	2.3.1 Research and finalize frontend and backend technologies	20
23	2.3.2 Define hosting and cloud infrastructure requirements	30
24	2.3.3 Identify security framework and API requirements	30
25	**3.0 Project Execution Phase**	**1110**
26	3.1 Backend Development Phase	200
27	3.1.1 Choose Cloud Platform	80
28	3.1.1.1 Select and implement cloud platform services (AWS & Firebase platform)	80
29	3.1.2 Choose DBS systems	120
30	3.1.2.1 Setup both MySQL and PostgreSQL as Database systems	120
31	3.2 Implement authentication and user verification	145
32	3.2.1 Create user authentication system	145

33	3.2.1.1 Design login, logout and password reset functionalities	15
34	3.2.2 Coordinate with NEU tech services for SSO integration details	60
35	3.2.3 Check the website's compatibility, user data security and login procedures	70
36	3.3 Implement main features	170
37	3.3.1 Create main backend features	50
38	3.3.1.1 Integrate products and profile database	30
39	3.3.1.2 Create APIs for user profile CRUD operations	20
40	3.3.2 Develop messaging system between buyer and seller	60
41	3.3.2.1 Use Firebase for real-time messaging	30
42	3.3.2.2 Ensure all messages are end-to-end encrypted	30
43	3.3.3 Integrate key backend services	60
44	3.3.3.1 Use Firebase cloud messaging for push notification	30
45	3.3.3.2 Plugin system support	30
46	3.4 Frontend Development Phase	205
47	3.4.1 UI/UX Prototyping and Design	20
48	3.4.1.1 Develop wireframes and UI/UX prototypes	20
49	3.4.2 Core UI Components	70
50	3.4.2.1 Create product cards with images, titles, prices, and filters	15
51	3.4.2.2 Create standard font size, color, and UI elements	15
52	3.4.2.3 Ensure button display hover, press and disabled states	15
53	3.4.2.4 Build elements like navigation and search bars, filters, recommendations, etc.	25
54	3.4.3 Building user profile interface	50
55	3.4.3.1 Profile Page layout (Buyer/Seller view)	10
56	3.4.3.2 View Seller Profiles (Product page -> Seller info)	15
57	3.4.3.3 View Buyer Profiles (Order -> Buyer info)	15
58	3.4.3.4 Display ratings and reviews	10
59	3.4.4 Identify required third-party APIs	65
60	3.4.4.1 Payment processing (Apple Pay, PayPal, Credit Card, Debit Card, Net Banking)	30
61	3.4.4.2 Evaluate ease of API integration, secure gateways, and connections	35
62	3.5 Integration of API and Testing Phase	165
63	3.5.1 Integrate Frontend and Backend functionalities	100
64	3.5.1.1 Implementing and connecting frontend elements to backend services through APIs	20
65	3.5.1.1.1 Test API functionalities	30
66	3.5.1.1.2 Testing API functionalities for CRUD operations in user profiles	30
67	3.5.1.1.3 Verifcation of login, logout, and token refresh APIs	20
68	3.5.2 Maintain data flow and functionalities	20
69	3.5.2.1 Ensure data flows and matches backend database in the frontend	20

70	3.5.3 Conduct user testing and gather feedback	20
71	3.5.3.1 Ensure API response times are responsive	10
72	3.5.3.2 Verify each component's feature and function	10
73	3.5.4 Refine and finalize UI/UX based on testing	25
74	3.5.4.1 Ensure smooth data transfer from user sign up to profile creation	25
75	3.6 Test Third-party Integrations	70
76	3.6.1 Secure external API integration	70
77	3.7 Security and Fraud Prevention	80
78	3.7.1 Implement secure transaction mechanisms	40
79	3.7.2 Payment gateway integration and plan transaction	25
80	3.7.3 Develop admin panel for moderation	15
81	3.8 Testing and Debugging	75
82	3.8.1 Conduct unit testing for individual modules	15
83	3.8.2 Perform system integration and stress testing	20
84	3.8.3 Identify and resolve bugs and performance issues	20
85	3.8.4 Conduct penetration testing and security	20
86	**4.0 Project Monitoring and Control**	**140**
87	4.1 Perform System Testing	60
88	4.1.1 Verify essential features like user login, product catalogue and messaging system	60
89	4.1.1.1 Ensure smooth log-in, log-out and catalogue browsing	30
90	4.1.1.2 Ensure seamless search experience and filter drilldown	30
91	4.2 Risk Monitoring and Control	40
92	4.2.1 Identify new risk that emerge during development	10
93	4.2.2 Implement contingency actions for triggered risks	20
94	4.2.3 Update risk register for effectiveness of mitigation	10
95	4.3 Progress tracking and reporting	40
96	4.3.1 Monitor actual progress against planned scheduled	20
97	4.3.2 Track resource utilization and allocation Implement contingency actions for triggered risks	20
98	**5.0 Project Closure Phase**	**195**
99	5.1 Deployment and Launch	105
100	5.1.1 Deploy platform on secure web server	10
101	5.1.2 Develop user guides and onboarding materials	40
102	5.1.3 Promote platform within the university community	55
103	5.2 Post Launch Monitoring	90
104	5.2.1 Monitor system performance and user feedback	30
105	5.2.2 Implement continuous improvements and updates	30
106	5.2.3 Document project learning and closure reports	30

Total Time in hours	1828 Hours

Table 5.0 Schedule Summary



ID	Task Mode	Task Name	Duration	Start	Finish
0		NU MART Development Schedule	1828 hr	Mon 4/28/2	Thu 3/12/26
1		1.0 Project Initiation Phase	168 hrs	Mon 4/28/25	Mon 5/26/25
2		1.1 Conduct Project Kick-off	83 hrs	Mon 4/28/25	Mon 5/12/25
3		1.1.1 Define Objectives, Scope and Constraints	20 hrs	Mon 4/28/25	Wed 4/30/25
4		1.1.2 Establish Project Timeline and Milestones	25 hrs	Wed 4/30/25	Mon 5/5/25
5		1.1.3 Identify Key Stakeholders	8 hrs	Mon 5/5/25	Tue 5/6/25
6		1.1.4 Draft Budget for the Project	30 hrs	Tue 5/6/25	Mon 5/12/25
7		1.2 Define Web Objectives	40 hrs	Mon 5/12/25	Mon 5/19/25
8		1.2.1 Establishing well-defined UI/design features	20 hrs	Mon 5/12/25	Wed 5/14/25
9		1.2.2 Focus on core features such as database management, payment gateways, etc.	20 hrs	Wed 5/14/25	Mon 5/19/25
10		1.3 Risk Management and Compliance	45 hrs	Mon 5/19/25	Mon 5/26/25
11		1.3.1 Identify potential risk and mitigation strategies	15 hrs	Mon 5/19/25	Wed 5/21/25
12		1.3.2 Ensure Compliance with Data Protection Regulations	10 hrs	Wed 5/21/25	Thu 5/22/25
13		1.3.3 Incorporate Cyber-security compliance and risk measures aligning with industry standards	20 hrs	Thu 5/22/25	Mon 5/26/25
14		2.0 Project Planning Phase	215 hrs	Tue 5/27/25	Wed 7/2/25
15		2.1 Conduct Market Research	60 hrs	Tue 5/27/25	Thu 6/5/25
16		2.1.1 Analyze existing platforms used by students	20 hrs	Tue 5/27/25	Thu 5/29/25
17		2.1.2 Identify key challenges and areas of improvement	40 hrs	Thu 5/29/25	Thu 6/5/25
18		2.2 Defining Requirements	75 hrs	Thu 6/5/25	Wed 6/18/25
19		2.2.1 Define functional and non-functional requirements	35 hrs	Thu 6/5/25	Wed 6/11/25
20		2.2.2 Define system architecture and workflow	40 hrs	Wed 6/11/25	Wed 6/18/25
21		2.3 Tech Stacks Selection	80 hrs	Wed 6/18/25	Wed 7/2/25
22		2.3.1 Research and finalize frontend and backend technologies	20 hrs	Wed 6/18/25	Mon 6/23/25
23		2.3.2 Define hosting and cloud infrastructure requirements	30 hrs	Mon 6/23/25	Fri 6/27/25
24		2.3.3 Identify security framework and API	30 hrs	Fri 6/27/25	Wed 7/2/25

Fig. 3.0 Gantt Chart

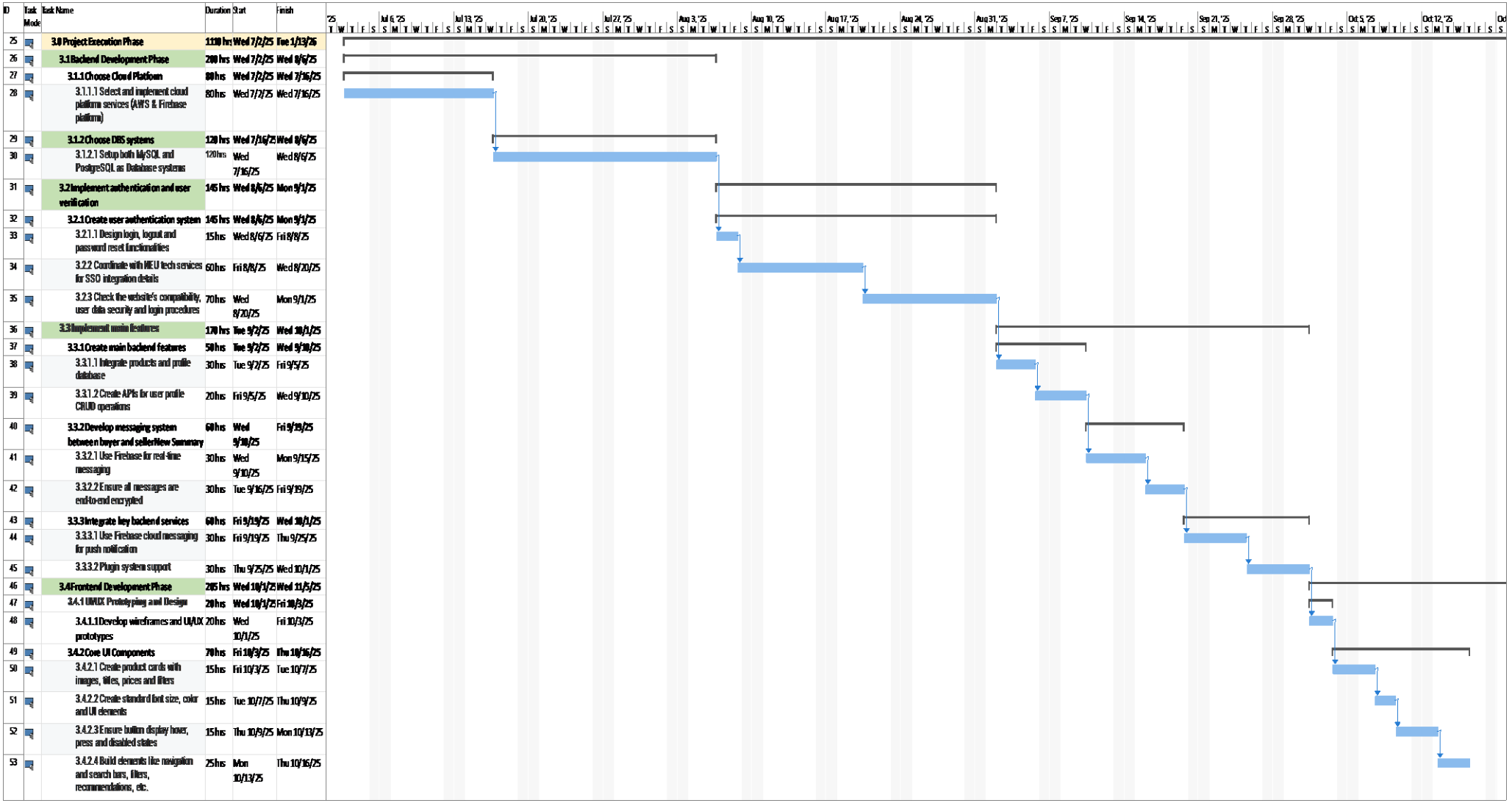


Fig. 3.1 Gantt Chart

ID	Task Name	Duration	Start	Finish
54	3.4.3 Building user profile interface	50 hrs	Thu 10/16/25	Fri 10/24/25
55	3.4.3.1 Profile Page layout (Buyer/Seller view)	10 hrs	Thu 10/16/25	Fri 10/17/25
56	3.4.3.2 View Seller Profiles (Product page -> Seller info)	15 hrs	Fri 10/17/25	Tue 10/21/25
57	3.4.3.3 View Buyer Profiles (Order -> Buyer info)	15 hrs	Tue 10/21/25	Thu 10/23/25
58	3.4.3.4 Display ratings and reviews	10 hrs	Thu 10/23/25	Fri 10/24/25
59	3.4.4 Identify required third-party APIs	65 hrs	Fri 10/24/25	Wed 11/5/25
60	3.4.4.1 Payment processing (Apple Pay, Pay Pal, Credit Card, Debit Card, Net Banking)	30 hrs	Fri 10/24/25	Thu 10/30/25
61	3.4.4.2 Evaluate ease of API integration, secure gateways and connections	35 hrs	Thu 10/30/25	Wed 11/5/25
62	3.5 Integration of API and Testing Phase	165 hrs	Wed 11/5/25	Thu 12/4/25
63	3.5.1 Integrate Frontend and Backend functionalities	100 hrs	Wed 11/5/25	Mon 11/24/25
64	3.5.1.1 Implementing and connecting frontend elements to backend services through APIs	20 hrs	Wed 11/5/25	Mon 11/10/25
65	3.5.1.1.1 Test API functionalities	30 hrs	Mon 11/10/2	Fri 11/14/25
66	3.5.1.1.2 Testing API functionalities for CRUD operations in user profiles	30 hrs	Fri 11/14/25	Wed 11/19/25
67	3.5.1.1.3 Verification of login, logout and token refresh APIs	20 hrs	Wed 11/19/25	Mon 11/24/25
68	3.5.2 Maintain data flow and functionalities	20 hrs	Mon 11/24/25	Wed 11/26/25
69	3.5.2.1 Ensure data flows and matches backend database in frontend	20 hrs	Mon 11/24/25	Wed 11/26/25
70	3.5.3 Conduct user testing and gather feedback	20 hrs	Wed 11/26/25	Mon 12/1/25
71	3.5.3.1 Ensure API response times are responsive	10 hrs	Wed 11/26/25	Fri 11/28/25
72	3.5.3.2 Verify each component's feature and function	10 hrs	Fri 11/28/25	Mon 12/1/25
73	3.5.4 Refine and finalize UI/UX based on testing	25 hrs	Mon 12/1/25	Thu 12/4/25
74	3.5.4.1 Ensure smooth data transfer from user sign up to profile creation	25 hrs	Mon 12/1/25	Thu 12/4/25
75	3.6 Test Third-party Integrations	70 hrs	Thu 12/4/25	Wed 12/17/25
76	3.6.1 Secure external API integration	70 hrs	Thu 12/4/25	Wed 12/17/25
77	3.7 Security and Fraud Prevention	80 hrs	Wed 12/17/2	Wed 12/31/25
78	3.7.1 Implement secure transaction mechanisms	40 hrs	Wed 12/17/25	Wed 12/24/25
79	3.7.2 Payment gateway integration and plan transaction	25 hrs	Wed 12/24/25	Mon 12/29/25
80	3.7.3 Develop admin panel for moderation	15 hrs	Mon 12/29/2	Wed 12/31/25

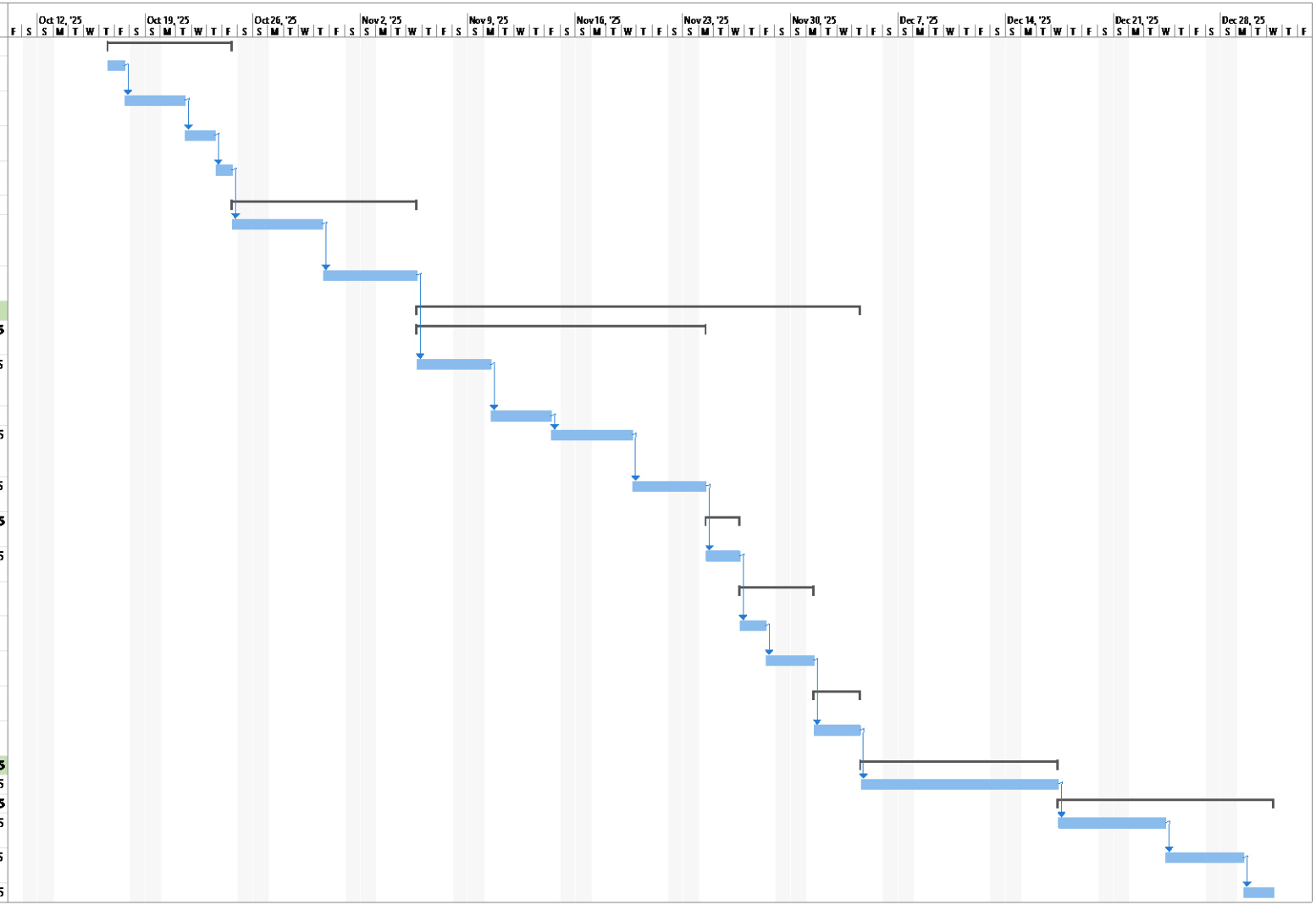
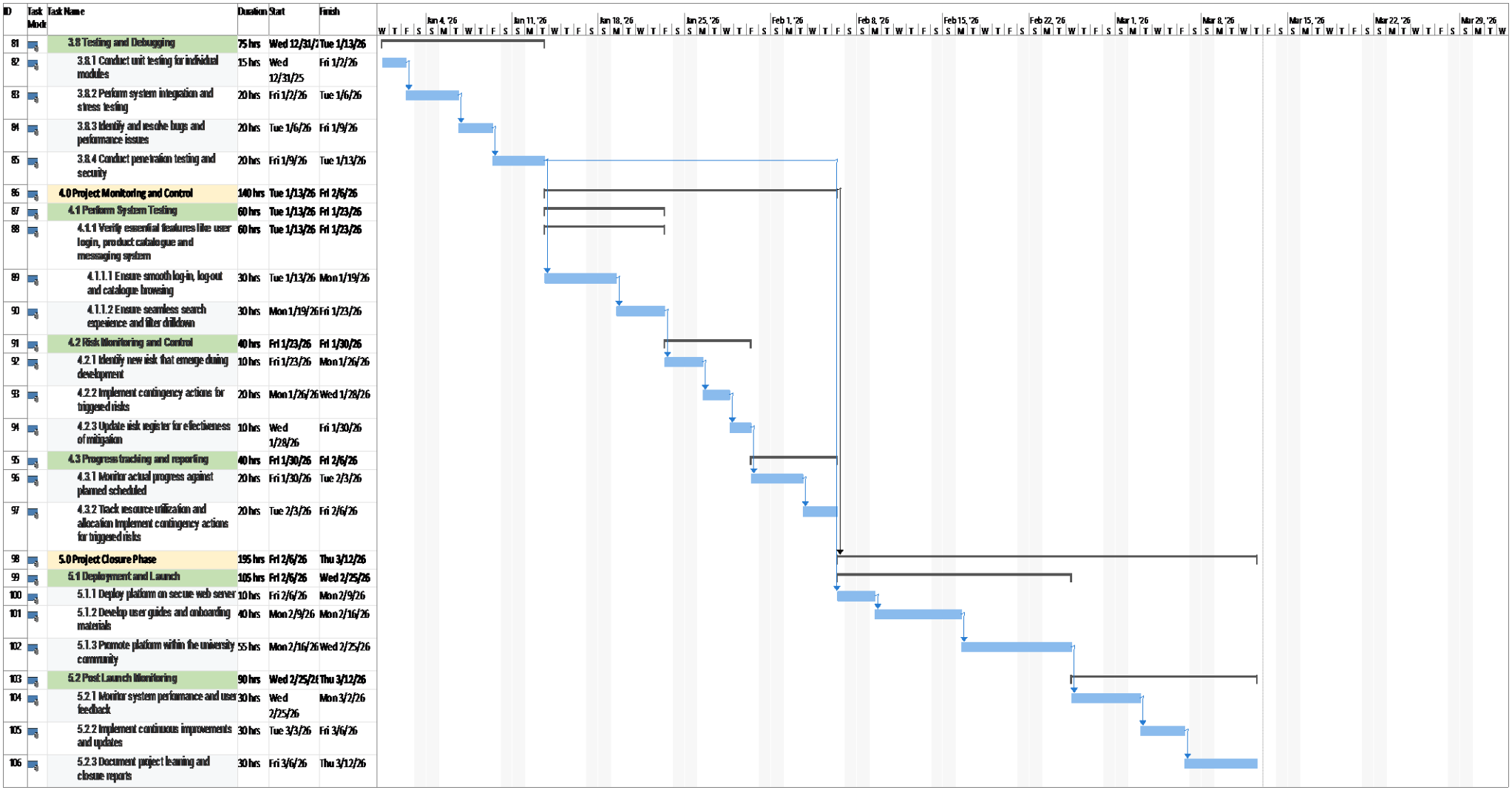


Fig. 3.2 Gantt Chart



ID	Task Mode	Task Name	Duration	Start	Finish
81		3.8 Testing and Debugging	75 hrs	Wed 12/31/2	Tue 1/13/26
82		3.8.1 Conduct unit testing for individual modules	15 hrs	Wed 12/31/25	Fri 1/2/26
83		3.8.2 Perform system integration and stress testing	20 hrs	Fri 1/2/26	Tue 1/6/26
84		3.8.3 Identify and resolve bugs and performance issues	20 hrs	Tue 1/6/26	Fri 1/9/26
85		3.8.4 Conduct penetration testing and security	20 hrs	Fri 1/9/26	Tue 1/13/26
86		4.0 Project Monitoring and Control	140 hrs	Tue 1/13/26	Fri 2/6/26
87		4.1 Perform System Testing	60 hrs	Tue 1/13/26	Fri 1/23/26
88		4.1.1 Verify essential features like user login, product catalogue and messaging system	60 hrs	Tue 1/13/26	Fri 1/23/26
89		4.1.1.1 Ensure smooth log-in, log-out and catalogue browsing	30 hrs	Tue 1/13/26	Mon 1/19/26
90		4.1.1.2 Ensure seamless search experience and filter drildown	30 hrs	Mon 1/19/26	Fri 1/23/26
91		4.2 Risk Monitoring and Control	40 hrs	Fri 1/23/26	Fri 1/30/26
92		4.2.1 Identify new risk that emerge during development	10 hrs	Fri 1/23/26	Mon 1/26/26
93		4.2.2 Implement contingency actions for triggered risks	20 hrs	Mon 1/26/26	Wed 1/28/26
94		4.2.3 Update risk register for effectiveness of mitigation	10 hrs	Wed 1/28/26	Fri 1/30/26
95		4.3 Progress tracking and reporting	40 hrs	Fri 1/30/26	Fri 2/6/26
96		4.3.1 Monitor actual progress against planned scheduled	20 hrs	Fri 1/30/26	Tue 2/3/26
97		4.3.2 Track resource utilization and allocation Implement contingency actions for triggered risks	20 hrs	Tue 2/3/26	Fri 2/6/26
98		5.0 Project Closure Phase	195 hrs	Fri 2/6/26	Thu 3/12/26
99		5.1 Deployment and Launch	105 hrs	Fri 2/6/26	Wed 2/25/26
100		5.1.1 Deploy platform on secure web server	10 hrs	Fri 2/6/26	Mon 2/9/26
101		5.1.2 Develop user guides and onboarding materials	40 hrs	Mon 2/9/26	Mon 2/16/26
102		5.1.3 Promote platform within the university community	55 hrs	Mon 2/16/26	Wed 2/25/26
103		5.2 Post Launch Monitoring	90 hrs	Wed 2/25/2	Thu 3/12/26
104		5.2.1 Monitor system performance and user feedback	30 hrs	Wed 2/25/26	Mon 3/2/26
105		5.2.2 Implement continuous improvements and updates	30 hrs	Tue 3/3/26	Fri 3/6/26
106		5.2.3 Document project learning and closure reports	30 hrs	Fri 3/6/26	Thu 3/12/26

Fig. 3.3 Gantt Chart

APPENDIX C: RACI MATRIX

ID	Task	Time Taken (in Hours)	Project Manager	Product Manager (PM)	Business Analyst	UI/UX Designer	Frontend Developer	Backend Developer	Database Engineer	Cybersecurity Specialist	QA Tester	Marketing/Community Manager
1	**1.0 Project Initiation Phase**	**168**										
2	1.1 Conduct Project Kick-off Meeting											
3	1.1.1 Define Objectives, Scope, and Constraints	20	R, A	C	C							I
4	1.1.2 Establish Project Timeline and Milestones	25	R	A	C	I	I	I	I	I	I	I
5	1.1.3 Identify Key Stakeholders	8	R	A	C							C
6	1.1.4 Draft Budget for the Project	30	R, A	C	C							
7	1.2 Define Web Objectives											
8	1.2.1 Establishing well-defined UI/design features	20		A		R	C					
9	1.2.2 Focus on core features such as database management, payment gateways, etc.	20		I				A	R	C		

10	1.3 Risk Management and Compliance											
11	1.3.1 Identify potential risk and mitigation strategies	15	A	R	C			I		C		
12	1.3.2 Ensure Compliance with Data Protection Regulations	10		A				R	R	R		
13	1.3.3 Incorporate Cyber-security compliance and risk measures aligning with industry standards	20		A					C	R		
14	**2.0 Project Planning Phase**	**215**										
15	2.1 Conduct Market Research											
16	2.1.1 Analyze existing platforms used by students	20		A	R	C	I	I				C
17	2.1.2 Identify key challenges and areas of improvement	40	C	A	R	C	C	C				C
18	2.2 Defining Requirements											
19	2.2.1 Define functional and non-functional requirements	35		A	R	C	C	C				

20	2.2.2 Define system architecture and workflow	40		A	C		C	R	R	C		
21	2.3 Tech Stacks Selection											
22	2.3.1 Research and finalize frontend and backend technologies	20		C			R	R		C		
23	2.3.2 Define hosting and cloud infrastructure requirements	30		A				R	R	C		
24	2.3.3 Identify security framework and API requirements	30		A						R		
25	**3.0 Project Execution Phase**	**1110**										
26	3.1 Backend Development Phase											
27	3.1.1 Choose Cloud Platform											
28	3.1.1.1 Select and implement cloud platform services (AWS & Firebase platform)	80		C				A	R	C		
29	3.1.2 Choose DBS systems											

30	3.1.2.1 Setup both MySQL and PostgreSQL as Database systems	120		C				A	R	C		
31	3.2 Implement authentication and user verification											
32	3.2.1 Create user authentication system											
33	3.2.1.1 Design login, logout and password reset functionalities	15		I		C	R	R		C		
34	3.2.2 Coordinate with NEU tech services for SSO integration details	60	R	I				C	A	C		
35	3.2.3 Check the website's compatibility, user data security and login procedures	70		I,A			C	C		R	R	
36	3.3 Implement main features											
37	3.3.1 Create main backend features											
38	3.3.1.1 Integrate products and profile database	30		I			C	R, A	C			
39	3.3.1.2 Create APIs for user profile CRUD operations	20		I				R, A		C		

40	3.3.2 Develop messaging system between buyer and seller											
41	3.3.2.1 Use Firebase for real-time messaging	30		I				A	A	R		
42	3.3.2.2 Ensure all messages are end-to-end encrypted	30		I				A				
43	3.3.3 Integrate key backend services											
44	3.3.3.1 Use Firebase cloud messaging for push notification	30		I				C	A			
45	3.3.3.2 Plugin system support	30		I				R				
46	3.4 Frontend Development Phase											
47	3.4.1 UI/UX Prototyping and Design											
48	3.4.1.1 Develop wireframes and UI/UX prototypes	20		A	C	R	C	I				
49	3.4.2 Core UI Components											
50	3.4.2.1 Create product cards with images, titles, prices, and filters	15		A		C	R					

51	3.4.2.2 Create standard font size, color, and UI elements	15		A		C	R					
52	3.4.2.3 Ensure button displays hover, press and disabled states	15		A			R					
53	3.4.2.4 Build elements like navigation and search bars, filters, recommendations, etc.	25		A		C	R					
54	3.4.3 Building user profile interface											
55	3.4.3.1 Profile Page layout (Buyer/Seller view)	10		A			R					
56	3.4.3.2 View Seller Profiles (Product page -> Seller info)	15		A		C	R					
57	3.4.3.3 View Buyer Profiles (Order -> Buyer info)	15		A		C	R					
58	3.4.3.4 Display ratings and reviews	10		A			R					
59	3.4.4 Identify required third-party APIs											

60	3.4.4.1 Payment processing (Apple Pay, PayPal, Credit Card, Debit Card, Net Banking)	30		I			R	R		A		
61	3.4.4.2 Evaluate ease of API integration, secure gateways, and connections	35		A			R	R		C		
62	3.5 Integration of API and Testing Phase											
63	3.5.1 Integrate Frontend and Backend functionalities											
64	3.5.1.1 Implementing and connecting frontend elements to backend services through APIs	20		C			R	R			A	
65	3.5.1.1.1 Test API functionalities	30		A			C	C			R	
66	3.5.1.1.2 Testing API functionalities for CRUD operations in user profiles	30		A	C	R	C				R	
67	3.5.1.1.3 Verifcation of login, logout, and token refresh APIs	20		A				R			A	
68	3.5.2 Maintain data flow and functionalities											

69	3.5.2.1 Ensure data flows and matches backend database in frontend	20		I			A	A	C		R	
70	3.5.3 Conduct user testing and gather feedback											
71	3.5.3.1 Ensure API response times are responsive	10		I			C	A			R	
72	3.5.3.2 Verify each component's feature and function	10		I			A	C			R	
73	3.5.4 Refine and finalize UI/UX based on testing											
74	3.5.4.1 Ensure smooth data transfer from user sign up to profile creation	25				R	C	R	C		A	
75	3.6 Test Third-party Integrations											
76	3.6.1 Secure external API integration	70		C			C	R, A				
77	3.7 Security and Fraud Prevention											
78	3.7.1 Implement secure transaction mechanisms	40		I				A	C	R	C	
79	3.7.2 Payment gateway integration and plan transaction	25		C				A		R	C	

80	3.7.3 Develop admin panel for moderation	15		I			R	A		R		
81	3.8 Testing and Debugging											
82	3.8.1 Conduct unit testing for individual modules	15					C	C			R, A	
83	3.8.2 Perform system integration and stress testing	20		A			C	C		C	R	
84	3.8.3 Identify and resolve bugs and performance issues	20		C			R	R	C		A	
85	3.8.4 Conduct penetration testing and security	20		I						R, A	R	
86	**4.0 Project Monitoring and Control**	**140**										
87	4.1 Perform System Testing											
88	4.1.1 Verify essential features like user login, product catalogue and messaging system											
89	4.1.1.1 Ensure smooth log-in, log-out and catalogue browsing	30					A	A			R	I

90	4.1.1.2 Ensure seamless search experience and filter drilldown	30					A	A			R	I
91	4.2 Risk Monitoring and Control											
92	4.2.1 Identify new risk that emerge during development	10	R	A	C					C		
93	4.2.2 Implement contingency actions for triggered risks	20	R	A	C					C		
94	4.2.3 Update risk register for effectiveness of mitigation	10	R		A					C		
95	4.3 Progress tracking and reporting											
96	4.3.1 Monitor actual progress against planned scheduled	20	R	A	C		I	I	I	I	I	I
97	4.3.2 Track resource utilization and allocation Implement contingency actions for triggered risks	20	R	A	C		I	I	I	I	I	I
98	**5.0 Project Closure Phase**	**195**										
99	5.1 Deployment and Launch											
100	5.1.1 Deploy platform on secure web server	10	R	A				C	C	C		I

101	5.1.2 Develop user guides and onboarding materials	40		C	R							A
102	5.1.3 Promote platform within the university community	55	C									R,A
103	5.2 Post Launch Monitoring											
104	5.2.1 Monitor system performance and user feedback	30		A			I	C	I	C	R	C
105	5.2.2 Implement continuous improvements and updates	30		C			A	A	A	C		I
106	5.2.3 Document project learning and closure reports	30	R		C							

Table 6.0 RACI Matrix Summary

APPENDIX D: BUDGET JUSTIFICATION

Task	Number of people	Total Working Hours	Cost
1.0 Project Initiation Phase			
1.1 Conduct Project Kick-off Meeting			
1.1.1 Define Objectives, Scope, and Constraints	4	33.00	1857.22
1.1.2 Establish Project Timeline and Milestones	10	45.00	2535.65
1.1.3 Identify Key Stakeholders	4	12.80	714.63
1.1.4 Draft Budget for the Project	2	48.00	2710.33
1.2 Define Web Objectives			
1.2.1 Establishing well-defined UI/design features	3	29.00	1655.90
1.2.2 Focus on core features such as database management, payment gateways, etc.	4	30.00	1713.00
1.3 Risk Management and Compliance			
1.3.1 Identify potential risks and mitigation strategies	5	24.75	1397.99
1.3.2 Ensure Compliance with Data Protection Regulations	4	33.00	1884.30
1.3.3 Incorporate cybersecurity compliance and risk measures aligning with industry standards	3	29.00	1655.90
2.0 Project Planning Phase			
2.1 Conduct Market Research			
2.1.1 Analyze existing platforms used by students	6	34.00	1785.69
2.1.2 Identify key challenges and areas of improvement	7	82.00	4370.78
2.2 Defining Requirements			
2.2.1 Define functional and non-functional requirements	5	61.25	3260.42
2.2.2 Define system architecture and workflow	6	110.00	6240.38
2.3 Tech Stacks Selection			
2.3.1 Research and finalize frontend and backend technologies	4	46.00	2626.60
2.3.2 Define hosting and cloud infrastructure requirements	4	73.50	4196.85
2.3.3 Identify security framework and API requirements	2	39.00	2226.90
3.0 Project Execution Phase			

3.1 Backend Development Phase			
3.1.1 Choose Cloud Platform			
3.1.1.1 Select and implement cloud platform services (AWS & Firebase platform)	4	128.00	7308.80
3.1.2 Choose DBS systems			
3.1.2.1 Setup both MySQL and PostgreSQL as Database systems	4	192.00	10963.20
3.2 Implement authentication and user verification			
3.2.1 Create user authentication system			
3.2.1.1 Design login, logout and password reset functionalities	5	35.25	2012.77
3.2.2 Coordinate with NEU tech services for SSO integration details	5	99.00	5652.90
3.2.3 Check the website's compatibility, user data security and login procedures	5	164.50	9392.95
3.3 Implement main features			
3.3.1 Create main backend features			
3.3.1.1 Integrate products and profile database	4	49.50	2826.45
3.3.1.2 Create APIs for user profile CRUD operations	3	30.00	1713.00
3.3.2 Develop a messaging system between buyer and seller			
3.3.2.1 Use Firebase for real-time messaging	4	49.50	2826.45
3.3.2.2 Ensure all messages are end-to-end encrypted	2	10.50	599.55
3.3.3 Integrate key backend services			
3.3.3.1 Use Firebase Cloud Messaging for push notification	3	15.00	856.50
3.3.3.2 Plugin system support	2	31.50	1798.65
3.4 Frontend Development Phase			
3.4.1 UI/UX Prototyping and Design			
3.4.1.1 Develop wireframes and UI/UX prototypes	5	33.00	1863.99
3.4.2 Core UI Components			
3.4.2.1 Create product cards with images, titles, prices, and filters	3	21.75	1241.92
3.4.2.2 Create standard font size, color, and UI elements	3	21.75	1241.92
3.4.2.3 Ensure button display hover, press and disabled states	2	19.50	1113.45
3.4.2.4 Build elements like navigation and search bars, filters, recommendations, etc.	3	36.25	2069.87

3.4.3 Building user profile interface			
3.4.3.1 Profile Page layout (Buyer/Seller view)	2	13.00	742.30
3.4.3.2 View Seller Profiles (Product page -> Seller info)	3	21.75	1241.92
3.4.3.3 View Buyer Profiles (Order -> Buyer info)	3	21.75	1241.92
3.4.3.4 Display ratings and reviews	2	13	742.30
3.4.4 Identify required third-party APIs			
3.4.4.1 Payment processing (Apple Pay, Pay Pal, Credit Card, Debit Card, Net Banking)	4	70.50	4025.55
3.4.4.2 Evaluate ease of API integration, secure gateways, and connections	4	85.75	4896.32
3.5 Integration of API and Testing Phase			
3.5.1 Integrate Frontend and Backend functionalities			
3.5.1.1 Implementing and connecting frontend elements to backend services through APIs	4	49.00	2797.90
3.5.1.1.1 Test API functionalities	4	48.00	2740.80
3.5.1.1.2 Testing API functionalities for CRUD operations in user profiles	5	78.00	4423.33
3.5.1.1.3 Verifcation of login, logout, and token refresh APIs	3	32.00	1827.20
3.5.2 Maintain data flow and functionalities			
3.5.2.1 Ensure data flows and matches backend database in frontend	5	36.00	2055.60
3.5.3 Conduct user testing and gather feedback			
3.5.3.1 Ensure API response times are responsive	4	15.00	856.50
3.5.3.2 Verify each component’s feature and function	4	15.00	856.50
3.5.4 Refine and finalize UI/UX based on testing			
3.5.4.1 Ensure smooth data transfer from user sign up to profile creation	4	65.00	3711.50
3.6 Test Third-party Integrations			
3.6.1 Secure external API integration	3	21.00	1199.10
3.7 Security and Fraud Prevention			
3.7.1 Implement secure transaction mechanisms	5	66.00	3768.60
3.7.2 Payment gateway integration and plan transaction	4	40.00	2284.00
3.7.3 Develop admin panel for moderation	4	35.25	2012.77
3.8 Testing and Debugging			

3.8.1 Conduct unit testing for individual modules	3	24.00	1370.40
3.8.2 Perform system integration and stress testing	5	35.00	1998.50
3.8.3 Identify and resolve bugs and performance issues	5	52.00	2969.20
3.8.4 Conduct penetration testing and security	3	47.00	2683.70
4.0 Project Monitoring and Control			
4.1 Perform System Testing			
4.1.1 Verify essential features like user login, product catalogue and messaging system			
4.1.1.1 Ensure smooth log-in, log-out and catalogue browsing	4	49.50	2816.29
4.1.1.2 Ensure seamless search experience and filter drilldown	4	49.50	2816.29
4.2 Risk Monitoring and Control			
4.2.1 Identify new risk that emerge during development	4	16.00	903.44
4.2.2 Implement contingency actions for triggered risks	4	32.00	1806.89
4.2.3 Update risk register for effectiveness of mitigation	3	14.50	807.64
4.3 Progress tracking and reporting			
4.3.1 Monitor actual progress against planned schedule	9	35.00	1971.42
4.3.2 Track resource utilization and allocation. Implement contingency actions for triggered risks	9	35.00	1971.42
5.0 Project Closure Phase			
5.1 Deployment and Launch			
5.1.1 Deploy platform on secure web server	6	18.00	1024.42
5.1.2 Develop user guides and onboarding materials	3	58.00	2959.76
5.1.3 Promote platform within the university community	2	60.25	3088.24
5.2 Post Launch Monitoring			
5.2.1 Monitor system performance and user feedback	7	55.50	3138.59
5.2.2 Implement continuous improvements and updates	6	37.50	2131.09
5.2.3 Document project learning and closure reports	2	34.50	1939.49

Table 7.0 Budget Summary

APPENDIX E: RESOURCE ALLOCATION

Sr. no.	Role	Total Hours in Project for Each Team Member	Hourly Wage allocation as per Bureau of Labor Statistics (in $)	Total Wage per Each Team Member	% Allocation
1	Project Manager	296.75	57.10	16944.43	16.23
2	Product Manager (PM)	309.15	57.10	17652.47	16.91
3	Business Analyst	181.20	50.33	9119.80	9.91
4	UI/UX Designer	125.50	57.10	7166.05	6.87
5	Frontend Developer	385.00	57.10	21983.50	21.06
6	Backend Developer	562.50	57.10	32118.75	30.77
7	Database Engineer	374.50	57.10	21383.95	20.49
8	Cybersecurity Specialist	366.00	57.10	20898.60	20.02
9	QA Tester	358.00	57.10	20441.80	19.58
10	Marketing/Community Manager	87.95	50.33	4426.53	4.81
Total Labor Cost				**$172,135.86**	

Table 8.0 Resource Allocation Summary

REFERENCES:

[1]	U.S Bureau of Labour Statistics: https://www.bls.gov/
[2]	Northeastern University Information Technology Services: https://its.northeastern.edu/
[3]	Tech Knowledge Base - Northeastern University: https://service.northeastern.edu/tech?id=kb_view2