



KBTC COLLEGE

KBTC School of IT

[NCC Global Assignment]	
Candidate Name:	Thar Linn Htet
Candidate ID Number:	202973
Center Name:	KBT001 - KBTC College (School of IT)
Unit:	Computing Project [CP]
Qualification:	L5DC
Date:	3/10/2025
Total Word Counts:	

Table of Contents

1	The Project Proposal.....	2
1.1	Project Background.....	2
1.2	The Aims and Objectives.....	2
1.3	Method of Solution / Architecture.....	3
1.4	Risk Assessment.....	4
1.5	Work Breakdown Structure and Gantt chart.....	5
1.6	Project Deliverables (artefacts).....	6
	References.....	7

KBTC COLLEGE
KBTC School of IT

1 The Project Proposal

1.1 Project Background

This computing project involves creating a Volunteer Management System (VMS) titled "Volunteer Connect". Since NGOs, hospitals and schools tend to rely on volunteers, they struggle to manage them because of decentralized systems of administration. When it comes to volunteer management, the current systems of emails, spreadsheets, and paperwork for recording hours, skill sets, and other important information involved are highly inefficient. The result is incomplete and inconsistent data.

This becomes a problem of data security and, more importantly, the handling of Background Checks/ID Verification data, compliance activities, and other personal information. The computing artefact being designed and built is a Volunteer Management System (VMS) which automates and centralizes the administrative functions directly related to volunteer activities and deployment. As a digital administration solution, organizations will manage the tasks of recruiting, onboarding, scheduling, and retaining volunteers in a cohesive system. The levels of reliability and efficiency offered in the systems provide an organization with the assurance of robust and dependable systems tracking volunteer efforts.

The reason for choosing this project comes from the growing need for digital tools that can make organizational tasks easier and more efficient. A lot of small and medium-sized organizations that depend on volunteers do not have the budget for expensive enterprise systems, but they still face the challenge of managing an increasing number of volunteers. Relying on spreadsheets or emails may work on a very small scale, but as numbers grow, these methods often cause problems such as scattered data, poor coordination and volunteers feeling unappreciated.

1.2 The Aims and Objectives

The aim of the project is to develop a reliable, efficient, and secure web-based Volunteer Management System that streamlines the volunteer lifecycle from initial registration and

onboarding through to assignment, time tracking, and event management, thereby optimizing resource deployment for the supporting organization.

To achieve the overall aim of developing the Volunteer Connect, the project will be broken down into a series of SMART objectives.

1. **Analysis and Requirements:** This phase focuses on gathering and prioritizing requirements using methods like MoSCoW to define both functional and non-functional needs.
2. **Design Specifications and Modeling:** The design stage aims to produce detailed specifications with structural and behavioral models using ERDs, Use Cases and UI design to guide the Volunteer management system's architecture.
3. **System Implementation:** This phase involves developing the core modules of the Volunteer Management System based on the approved design. Implementation will focus on secure, efficient coding practices and modular development to ensure scalability.
4. **Testing and QA:** The testing phase ensures that all modules meet functional and non-functional requirements meet the expectation.
5. **Deployment and Training:** Once tested, the system will be deployed in a controlled environment. Training sessions and user manuals will be provided to administrators and volunteers to ensure smooth adoption.

1.3 Method of Solution / Architecture

The problem will be solved by developing a web-based platform that centralizes volunteer management tasks into a single cohesive system. The development will follow an Agile methodology (Robinson, 2024), which allows iterative development, regular testing, and feedback from potential end-users. This approach is suitable because it ensures flexibility and alignment with the actual needs of NGOs and schools.

The proposed architecture will consist of the following:

- **Frontend:** Developed using Next.js (React) for its modular structure, reactivity, and community support. This ensures a responsive and intuitive user interface.
- **Backend:** Implemented using Node.js with Nest.js to provide scalability, asynchronous processing, and compatibility with RESTful APIs.
- **Database:** A relational database such as MySQL will be used to securely store volunteer data, event details, and time logs. MySQL was selected due to its robustness, security features, and wide adoption.
- **Authentication and Security:** User authentication will be implemented with JSON Web Tokens (JWT) and encrypted password storage using bcrypt. SSL/TLS encryption will ensure secure data transmission.
- **Hosting and Deployment:** The application will be deployed on a cloud service such as AWS or Azure to guarantee reliability, scalability, and availability. The choice of technologies reflects the need for open-source, cost-effective tools that small and medium-sized organizations can adopt without high financial burdens.

1.4 Risk Assessment

A risk assessment has been conducted to identify potential obstacles and evaluate their likelihood and impact. The following table summarizes the findings:

Risk	Likelihood	Impact	Exposure	Mitigation Strategy
Data security breach	Medium	High	High	Implement strong encryption, role-based access, and regular security audits.
Scope creep	High	Medium	High	Use MoSCoW prioritization and ensure strict adherence to requirements.
Limited stakeholder engagement	Medium	Medium	Medium	Schedule regular meetings with stakeholders for feedback.

Technical challenges in integration	Medium	Medium	Medium	Allocate time for research, prototyping, and use of open-source libraries.
Time overruns	Medium	High	High	Create realistic schedules, monitor progress, and adjust timelines.
System DownTime	Low	High	Medium	Medium Host on reliable cloud infrastructure, monitor uptime, and maintain a fallback server.

The risks with the highest exposure (data security, scope creep, and time overruns) are prioritized for active management.

1.5 Work Breakdown Structure and Gantt chart

For the successful completion of the **Volunteer Connect** system, the project will be organized into structured activities that follow a two-month (8-week) schedule. Each task is sequenced to ensure dependencies are respected and that progress can be tracked effectively. The Work Breakdown Structure (WBS) identifies the major deliverables and their respective time allocations.

The Gantt chart below illustrates the timeline, duration, and order of tasks (Martins, 2025):

Key Milestones (2-Month Plan):

- Requirements analysis completed
- System design finalized
- Frontend development completed (**Week 4**)
- Backend and database implementation completed (**Week 6**)
- Testing completed (**Week 7**)
- Final deployment and documentation (**Week 8**)

This condensed timeline ensures that all project phases are covered within two months while still maintaining logical sequencing. The Gantt chart serves as both a monitoring and planning tool, allowing early identification of delays and supporting efficient resource allocation.

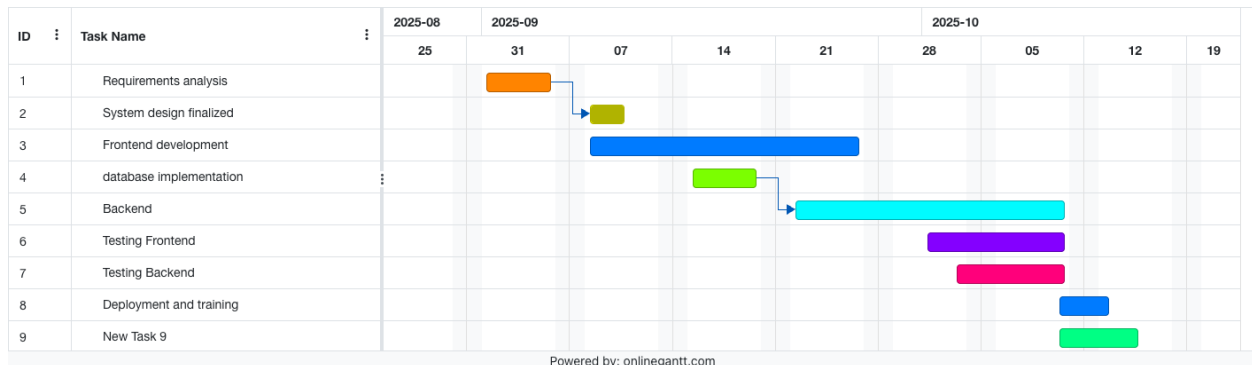


Figure 1 Project Timeline

1.6 Project Deliverables (artefacts)

The project will result in the following deliverables:

1. Software Artefacts:

- A fully functional Volunteer Management System (Volunteer Connect) accessible through a web interface.
- Core modules include user registration, volunteer onboarding, volunteer tasking, real time communication, role-based assignments, volunteer certifications and reporting.
- Secure authentication and data protection features.

2. Documentation:

- Requirements specification document.
- Design models including ERDs, Use Case diagrams, and UI wireframes.
- Test plans, test cases, and QA reports.
- User manuals and training materials for administrators and volunteers.

3. Project Report:

- a. A comprehensive academic report documenting the development process, methodologies used, challenges faced, and evaluation of outcomes.

References

Martins, J., 2025. *resources/gantt-chart-basics*. [Online]

Available at: <https://asana.com/>

[Accessed 3 Oct 2025].

Robinson, S., 2024. *searchsoftwarequality/definition/agile-software-development*. [Online]

Available at: <https://www.techtarget.com/>

[Accessed 1 Oct 2025].

