UNIGLOBE COLLEGE

**(Affiliated to Pokhara University)**

New Baneswor, Kathmandu



**Third Semester Project Report**

**on**

**“AnonAeon: Anonymous Feedback Wall”**

**(CMP273)**

A Third Semester Project Report submitted in the partial fulfillment of the requirements for the degree of Bachelor of Computer System and Information Technology awarded by Pokhara University

**Under the supervision of**

**Bipin Maharjan**

**Lecturer/Supervisor**

**Submitted By:**

**Shulabh Shrestha (P.U. Registration No. 2024-2-08-0747)**

**Submitted To:**

**UNIGLOBE COLLEGE**

**Department of Computer Science and Information Technology**

**New Baneswor, Kathmandu, Nepal**

**November, 2025**

**“AnonAeon: Anonymous Feedback Wall”**

**(CMP273)**

A third semester project report submitted in partial fulfillment of the requirements for the degree of Bachelor of Computer System and Information Technology awarded by Pokhara University

**Submitted By**

Shulabh Shrestha (P.U. Registration No. 2024-2-08-0747)

**Submitted To**

**UNIGLOBE COLLEGE**

**Department of Computer Science and Information Technology**

**New Baneshwor, Kathmandu, Nepal**

**November, 2025**

# TABLE OF CONTENTS

Table of Contents

[TABLE OF CONTENTS 3](#_Toc215034042)

[ABBREVIATIONS 4](#_Toc215034043)

[CHAPTER 1: INTRODUCTION 1](#_Toc215034044)

[1.1 Background 1](#_Toc215034045)

[1.2 Problem Statement 1](#_Toc215034046)

[1.3 Objectives of the Project 1](#_Toc215034047)

[1.4 Scope of the Project 1](#_Toc215034048)

[1.5 Limitations of the Project 2](#_Toc215034049)

[1.6 Significance of the Study 2](#_Toc215034050)

[CHAPTER 2: LITERATURE REVIEW 3](#_Toc215034051)

[CHAPTER 3: SYSTEM ANALYSIS AND DESIGN 4](#_Toc215034052)

[3.1 Project Management Strategy and Development Tools 4](#_Toc215034053)

[3.1.1 Project Team 4](#_Toc215034054)

[3.1.2 Project Flow and Schedule 4](#_Toc215034055)

[3.1.3 Responsibilities 5](#_Toc215034056)

[3.1.4 Development Tools 6](#_Toc215034057)

[3.2 System Analysis 7](#_Toc215034058)

[3.2.1 Requirement Analysis 7](#_Toc215034059)

[3.2.2 Feasibility Analysis 8](#_Toc215034060)

[3.3 System Design 9](#_Toc215034061)

[3.3.1. System Flowchart 9](#_Toc215034062)

[3.3.2 Workflow 10](#_Toc215034063)

[3.3.3 Data Flow Diagram 11](#_Toc215034064)

[3.3.4 ER Diagram 12](#_Toc215034065)

[3.3.5 Use Case Diagram 13](#_Toc215034066)

[3.3.6 Schema of Database 13](#_Toc215034067)

[3.3.7 Sequence diagrams 14](#_Toc215034068)

[CHAPTER 4: SYSTEM IMPLEMENTATION 16](#_Toc215034069)

[4.1 Project Schedule 16](#_Toc215034070)

[4.1.1 Detailed Time Schedule 17](#_Toc215034071)

[4.1.2 Detailed Gantt Chart 17](#_Toc215034072)

[4.2 Implementation 18](#_Toc215034073)

[4.3 Testing 19](#_Toc215034074)

[4.3.1 Unit Testing 19](#_Toc215034075)

[4.3.2 Integration Testing 20](#_Toc215034076)

[4.3.3 System Testing 20](#_Toc215034077)

[4.4 Result Analysis 21](#_Toc215034078)

[4.4.1 Screenshots 21](#_Toc215034079)

[4.5 Critical Analysis 22](#_Toc215034080)

[CHAPTER 5: CONCLUSION AND FUTURE ENHANCEMENT 23](#_Toc215034081)

[5.1 Conclusion 23](#_Toc215034082)

[5.2 Future Enhancement 23](#_Toc215034084)

[REFERENCES 24](#_Toc215034085)

# ABBREVIATIONS

|  |  |
| --- | --- |
| CSS  HTML  JS  SQL  PHP  XAMPP | Cascading Style Sheet  Hyper Text Markup Language  JavaScript  Sequence Query Language  Hypertext Preprocessor  Cross-Platform (X), Apache (A), MariaDB (M), PHP (P), and Perl (P) |
|  |  |
|  |  |
|  |  |
|  |  |

# CHAPTER 1: INTRODUCTION

## Background

AnonAeon is an anonymous feedback system designed to allow users to post feedback either with their identity or anonymously. The platform enhances open communication in academic or organizational environments by providing a digital wall for feedback exchange. This project is made using PHP, HTML, CSS and JS for functionality.

## Problem Statement

Many students and employees hesitate to share honest feedback due to fear of judgment. AnonAeon solves this by allowing optional anonymity while maintaining accountability through login.

## 1.3 Objectives of the Project

The Objectives of the Project are as follows:

- To create a secure feedback system with optional anonymous posting.  
- To allow administrators to manage feedback and users effectively.

## 1.4 Scope of the Project

The system allows login, posting feedback, viewing the feedback wall, and admin features such as deleting feedback and users.

## 1.5 Limitations of the Project

The system does not include advanced analytics, sentiment analysis, or real-time notifications.

And since the identity is optionally hidden, the users might take advantage of this scope of this project and use it with ill intent.

## 1.6 Significance of the Study

The study benefits students and organizations by enabling open, honest, and safe communication channels.

# CHAPTER 2: LITERATURE REVIEW

Similar anonymous feedback systems exist globally, including tell-me apps, suggestion portals, and academic review tools. AnonAeon differentiates itself with a simple, optional anonymity feature and admin-managed environment.

This project is inspired by some sites such as X(Formally twitter) and 4chan(another anonymous website). It works similar to how twitter and 4chan works. Here,

* Users are needed to log in in order to post something
* Users are allowed to post publicly visible messages
* Users can read other’s post on a public wall
* It only differs as the visible posts are separated by group of people example BCSIT, BBA, etc. which group is selected during the registration process.
* It differs from twitter as it’s not a social media and moderation is done by administrator
* It differs from 4chan as it’s a feedback system where as 4chan is an Anonymous discussion forum

# CHAPTER 3: SYSTEM ANALYSIS AND DESIGN

## 3.1 Project Management Strategy and Development Tools

The project follows a structured development model using PHP, MySQL, HTML/CSS, and XAMPP.

### 3.1.1 Project Team

|  |  |
| --- | --- |
| **Team Resource** | **Role** |
| Bipin Maharjan | Supervisor |
| Shulabh Shrestha | Software Developer |

Table 2: Team Resource and Roles

The AnonAeon project by Shulabh Shrestha is supervised by Mr. Bipin Maharjan.

### 3.1.2 Project Flow and Schedule

* Team Size: 1
* Total Project Duration: 10 weeks
* Effort Required per person: 8 hours per week

### 3.1.3 Responsibilities

This section States the responsibilities of each members of the project

#### 3.1.3.1 Responsibilities of Supervisor

#### Provide guidance throughout the planning, analysis, design, and implementation phases of the project.

* Offer constructive feedback to improve system design, security, and functionality.
* Evaluate the final system, documentation, and presentation as part of academic assessment.

#### 3.1.3.2 Responsibilities of team member

* Conduct requirement analysis to understand the problem domain and define system features
* Design System Architecture, workflow diagrams, DFDs, ERDs, and use case for AnonAeon
* Develop full web application using PHP, HTML, CSS and JavaScript.
* Ensure timely completion of all project phases according to the schedule
* Prepare final report, demonstration materials, and contribute to the final evaluation process

### 3.1.4 Development Tools

Backend: PHP

Frontend: HTML/CSS/JavaScript

Database: MySQL

Server: Apache

Testing: Manual Testing

#### 3.1.4.1 Backend Tools

PHP is used for backend language as it is one of the most widely used language for server side development and it integrates seamlessly with MySQL making data handling efficient. PHP runs smoothly on apache servers providing stable environment.

#### Front End Tools

HTMLProvides the structure and layout of webpages.It is used to design components like forms, buttons, feedback cards, and navigation elements.

CSSControls the styling, appearance, and responsiveness of the website. Makes the interface visually appealing and user-friendly.It helps maintain consistency across pages such as login, feedback wall, and admin panel.

JavaScriptAdds interactivity and dynamic behavior to the web pages.JS is used for form validation, real-time UI updates, and enhancing user experience.It helps with asynchronous interactions and improves the responsiveness of the system.

#### Web Server

Apache is used as a web server because it is stable, open source and widely used in php applications.It’s fully compatible with XAMPP environments and commonly used by developers.

#### Testing Tools

Manual testing is chosen because the project is medium scale, making manual checks practical and efficient. Allows the developer to interact directly with the system identify UI/UX.

## System Analysis

The system is designed based on collected requirements for secure login, feedback posting, admin controls, and user experience

### 3.2.1 Requirement Analysis

The collected information is structured, conflicts are resolved and requirements are prioritized.

-- Functional Requirements(FR):

* FR1: The system must allow users to register and log in using a valid username and password
* FR2: The system must allow users to post feedback in the feedback wall either anonymously or with their username visible
* FR3: The system must display all the posted feedback by the logged in users, the wall must update with newly added feedback
* FR4: The admin must be able to log in using admin credentials, must be able to view all users and feedbacks. They must also be able to delete users or feedbacks from the system. They can post as **Admin** or **Anon**.
* FR5: The system must store user details, feedback entries, timestamps and anonymity settings

-- Non-Functional Requirements(NFR):

* NFR1: The user interface must be clean, simple and easy to navigate.
* NFR2: User passwords must be stored securely and sessions must be protected to prevent unauthorized access
* NFR3: The system should load page within 3s on standard devices. Feedback wall must handle all feedbacks without noticable delay.
* NFR4: The system should operate continuously without failure under normal conditions and system downtime should be minimized
* NFR5: The system should handle increasing numbers of users feedback entries.
* NFR6: The codebase should be modular and easy to update, documentation should be provided for future developers and system must support bug fixing without affecting data.

### 3.2.2 Feasibility Analysis

**1. Technical Feasibility:**

The system uses widely supported technologies(PHP, HTML, CSS, JS) which makes development practical. No advanced hardware and complex algorithms are needed.

**2. Operational Feasibility:**

Users can easily post messages by entering text and selecting department room. Optional anonymity encourages participation. Restricting posts to department-specific rooms keeps the app organized and reduces irrelevant messages.

**3. Economic Feasibility:**

Development tools like XAMPP, VS Code and MySQL are free to use and maintenance cost are low due to system’s simplicity.

**4. Time Feasibility:**

This project is simple enough to be completed within 1-2 months even by a single person. Tasks such as UI design, backend design, database setup, and testing are manageable during this time frame.

## 3.3 System Design

### 3.3.1. System Flowchart

A diagram of a software process

AI-generated content may be incorrect.

Fig 3.1 : System Flowchart

### 3.3.2 Workflow



Fig 3.2 : Workflow diagram

### 3.3.3 Data Flow Diagram

#### 3.3.3.1 Data Flow Diagram Level 0

A diagram of a user

AI-generated content may be incorrect.

Fig 3.3 : DFD Lvl 0

#### 3.3.3.2 Data Flow Diagram Level 1

A diagram of a flowchart

AI-generated content may be incorrect.

Fig 3.4 : DFD Lvl 1

#### 3.3.3.3 Data Flow Diagram Level 2

A diagram of a process

AI-generated content may be incorrect.

Fig 3.5 : DFD Lvl 2

### 3.3.4 ER Diagram

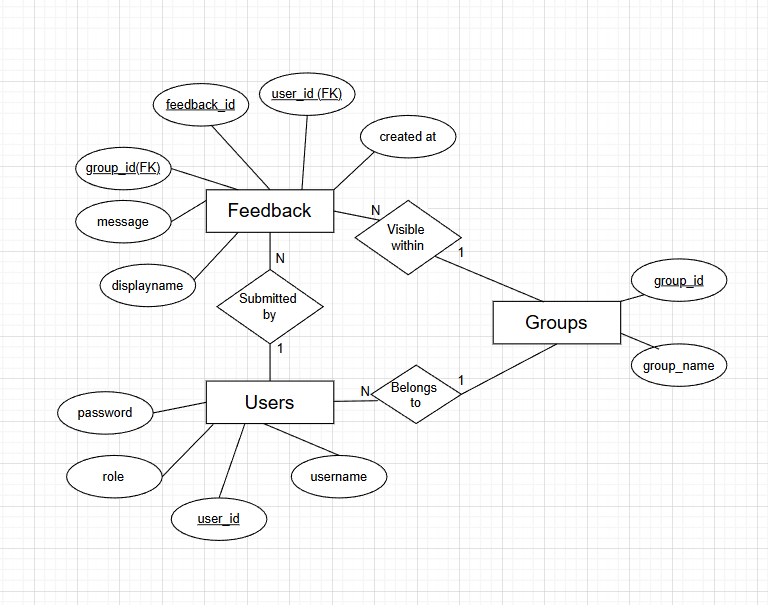


Fig 3.6: ER Diagram

### 3.3.5 Use Case Diagram

A diagram of a customer feedback

AI-generated content may be incorrect.

Fig 3.7 : Use case diagram

### 3.3.6 Schema of Database

Feedback schema

A screenshot of a computer

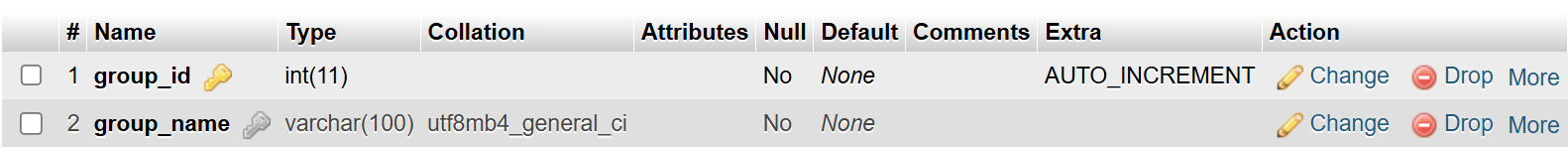
AI-generated content may be incorrect.

Users schema

A screenshot of a computer

AI-generated content may be incorrect.

Groups schema



### 3.3.7 Sequence diagrams

A diagram of a login sequence

AI-generated content may be incorrect.

A diagram of a program

AI-generated content may be incorrect.

A screenshot of a diagram

AI-generated content may be incorrect.

# CHAPTER 4: SYSTEM IMPLEMENTATION

## 4.1 Project Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task ID** | **Task Name** | **Start Date** | **End Date** | **Duration** |
| **1. Preliminary Work** |  |  |  |  |
| **1.1** | **Planning of project** | **11-Nov** | **12-Nov** | **2** |
| **1.2** | **Analysis of topics** | **12-Nov** | **14-Nov** | **3** |
| **1.3** | **Background Reading** | **14-Nov** | **15-Nov** | **2** |
| **1.4** | **Preliminary Research** | **15-Nov** | **15-Nov** | **1** |
| **2. Research Work** |  |  |  |  |
| **2.1** | **Research on Feedback handling** | **15-Nov** | **17-Nov** | **3** |
| **2.2** | **Research on Administrative actions** | **17-Nov** | **19-Nov** | **3** |
| **3. Design and Development** |  |  |  |  |
| **3.1** | **System Architecture Design** | **20-Nov** | **21-Nov** | **2** |
| **3.2** | **Context and DFD Diagram** | **21-Nov** | **21-Nov** | **1** |
| **3.3** | **Designing System Flowchart** | **21-Nov** | **21-Nov** | **1** |
| **3.4** | **Use case diagram** | **22-Nov** | **22-Nov** | **1** |
| **3.5** | **ER Diagram** | **22-Nov** | **22-Nov** | **1** |
| **3.6** | **Sequence Diagram** | **22-Nov** | **22-Nov** | **1** |
| **3.7** | **Schema of Database** | **22-Nov** | **22-Nov** | **1** |
| **4. Coding and Implementation** |  |  |  |  |
| **4.1** | **Code User Interface** | **22-Nov** | **26-Nov** | **5** |
| **4.2** | **Code User sign up and log in** | **26-Nov** | **30-Nov** | **5** |
| **4.3** | **Test User Signup and log in** | **30-Nov** | **1-Dec** | **2** |
| **4.4** | **Code User feedback submission** | **4-Dec** | **8-Dec** | **5** |
| **4.5** | **Test user feedback submission** | **8-Dec** | **11-Dec** | **4** |
| **4.6** | **Code Admin UI** | **11-Dec** | **16-Dec** | **6** |
| **4.7** | **Code Admin Feedback Deletion** | **16-Dec** | **21-Dec** | **6** |
| **4.8** | **Test Admin Feedback Deletion** | **21-Dec** | **24-Dec** | **4** |
| **4.9** | **Code Admin User management** | **24-Dec** | **29-Dec** | **6** |
| **4.11** | **Test Admin User management** | **29-Dec** | **29-Dec** | **1** |
| **4.12** | **Test System** | **29-Dec** | **2-Jan** | **5** |
| **4.13** | **Validate System** | **2-Jan** | **3-Jan** | **2** |
| **5. Dissertation** |  |  |  |  |
| **5.1** | **Critical Analysis** | **8-Jan** | **10-Jan** | **3** |
| **5.2** | **Draft Report Writing** | **10-Jan** | **11-Jan** | **2** |
| **5.3** | **Final Report Writing** | **11-Jan** | **13-Jan** | **3** |
| **5.4** | **Report Evaluation and Conclusion** | **13-Jan** | **15-Jan** | **3** |
| **5.5** | **Submission of draft copy report** | **15-Jan** | **15-Jan** | **1** |
| **5.6** | **Correction for final draft copy** | **15-Jan** | **16-Jan** | **2** |
| **6. Final Phase** |  |  |  |  |
| **6.1** | **Final Documentation Printing and Building** | **15-Jan** | **15-Jan** | **1** |
| **6.2** | **Submission of final report** | **16-Jan** | **16-Jan** | **1** |

### 4.1.1 Detailed Time Schedule

Table 3: Detailed Time Schedule

### 4.1.2 Detailed Gantt Chart

A graph of a graph with text

AI-generated content may be incorrect.

Fig 4.1 : Gantt Chart

## 

## 4.2 Implementation

1. Feedback submission module: Allows users to submit feedback anonymously through a simple form. No personal information is collected to maintain complete anonymity. Optionally, users can also choose to display their username while posting feedback,

2. Feedback Display module: Shows all submitted feedback in a clean and organized layout. Displays entries in list or card format for easy reading. Optional sorting for better usability.

3. Admin log in and authentication module: Secure login system for administrators only. Ensures that only authorized users can view or manage feedback. Uses PHP sessions for access control.

4. Database management module: Stores feedback entries safely in a MySQL database. Handles saving, retrieving and displaying data efficiently. Ensures structured storage with proper timestamps

5. UI/UX Frontend module: Built using HTML CSS and Javascript for a smooth user experience. Responsive design for laptops, tablets and mobile screens. Clean, minimal interface for easy navigation

## 4.3 Testing

### 4.3.1 Unit Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CaseID** | **Description** | **Input** | **Exp. Result** | **Actual Result** | **Status** |
| TC1 | Duplicate Email | Register with existing email | Error msg: “Email already used” | (Yet to be tested) | **\_\_\_\_\_** |
| TC2 | Wrong Password | Correct email + wrong password | “Invalid Credentials” message | It redirects to invalid credentials page but doesn’t display the message | **FAIL** |
| TC3 | User logs out | Click log out button | Session destroyed, redirect to login.php | User successfully redirects back to log in page | **PASS** |

**Table 4: Unit testing**

**Test Function Prototype**

**Result**

Screenshot and explanation

### 4.3.2 Integration Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TESTID | **Description** | **Input** | **Exp. Result** | **Actual Result** | **Status** |
| TC1 | Register with valid information | Username:"lumi", Email: "lumi@example.com", Password: "Test@123" | Account is created and user is redirected to login page | User successfully registers and is redirected to login | **PASS** |
| TC2 | Logging in valid credentials | Valid email + Password | Redirect to feedback submission page | User is successfully redirected to feedback submission page | **PASS** |

Table 5: Integration testing

### 4.3.3 System Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TESTID | **Description** | **Input** | **Exp. Result** | **Actual Result** | **Status** |
| TC1 | Redirecting to feedback wall | Click “Go to feedback wall ->” | User is redirected to feedback wall | User successfully redirects to feedback wall | **PASS** |
| TC2 | Admin can view all feedback and filter them | Admin enters feedback wall | Admin can see all feedback from all groups and can filter them according to groups | Admin is unable to see any feedback and unable to filter them | **FAIL** |

Table 6: System testing

## 4.4 Result Analysis

### 4.4.1 Screenshots

A screenshot of a computer screen

AI-generated content may be incorrect.A screenshot of a login screen

AI-generated content may be incorrect.

Fig 4.2: Login page + Responsive

A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a login form

AI-generated content may be incorrect.

Fig 4.3: Register page + Responsive

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.

Fig 4.4 : Submit feedback page + Responsive

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.

Fig 4.5 : Feedback wall(with test data to check responsive) + Responsive

## 4.5 Critical Analysis

Discuss system performance, limitations, and improvements.

# CHAPTER 5: CONCLUSION AND FUTURE ENHANCEMENT

## 5.1 Conclusion

## Overall, this project demonstrates a practical and effective approach to anonymous communication within educational or organizational settings. It enhances transparency, encourages honest input, and streamlines feedback management for both users and administrators. The system is scalable, extensible, and ready for further improvements such as analytics, multimedia attachments, or real-time updates.

## 5.2 Future Enhancement

Some one the future enhancement that can be added are:

1. Real-Time feedback updates: Integrate AJAX or WebSockets to allow feedback t appear instantly without requiring page reloads.
2. Reactions: Reactions(Such as like or emojis) to promote more interactions
3. File Attachments: Provide an option to upload screenshots or documents along with feedback.
4. Theme switch: Allow users to switch from dark mode to light mode.
5. Advanced filtering and search: Allow users and admins to filter feedback by: Date range, sentiment, keywords, user(if non anonymous)

# REFERENCES

[1] Shaheen, A., Azam, F., & Irshad, K. (2021, October). Anonymous Feedback: A real-time feedback to capture students’ engagement in learning process. In *International Conference on Medical Education (ICME 2021)* (pp. 65-68). Atlantis Press.

[2] Frampton, A., Fox, F., Hollowood, A., Northstone, K., Margelyte, R., Smith-Clarke, S., & Redwood, S. (2017). Using real-time, anonymous staff feedback to improve staff experience and engagement. *BMJ Quality Improvement Reports*, *6*(1).