COLLEGE CODE : 8203

COLLEGE NAME : AVC COLLEGE OF ENGINEERING

DEPARTMENT : BE.CSE

STUDENT NM-ID: F836E05B6F1F228817443AEA06454328

ROLL NO : 820323104087

DATE : 08-09-2025

Completed the project named as

Phase__ TECHNOLOGY PROJECT

NAME : SHANMUGAPRIYA S

SUBMITTED BY,

NAME : SAHANA K

MOBILE NO : 7810095672

Phase 1 – Problem Understanding & Requirements

Problem Statement

In any organization, user feedback is critical for measuring satisfaction, identifying pain points, and improving services. However, most feedback is either collected manually or scattered across different platforms such as emails, social media, or spreadsheets. This lack of structure makes it difficult to analyze feedback effectively and respond in a timely manner.

The Feedback Collection System addresses this issue by providing a centralized platform where users can easily submit their feedback and organizations can store, organize, and review it. By integrating with a database like MongoDB and building REST APIs with Node.js and Express, the system ensures that data is securely stored, timestamped, and accessible through admin tools. Future enhancements like sentiment analysis and email notifications will help organizations gain deeper insights and maintain better engagement with users.

Users & Stakeholders

- End Users (Customers/Clients): Provide feedback through a simple web or mobile interface.
- Admins/Organization Staff: Require an easy way to

access all collected feedback, filter, and analyze.
- Developers/Technical Team: Responsible for system design, implementation, and maintenance.
- Business Stakeholders (Management/Owners): Seek actionable insights from feedback to improve services.

User Stories

1. As a user, I want to submit my feedback easily through a form so that my opinion is recorded without delays.
2. As a user, I want assurance that my feedback has been received so that I feel valued.
3. As an admin, I want to view a list of all submitted feedback along with timestamps so that I know when they were given.
4. As an admin, I want to filter and search feedback by parameters such as date, rating, or sentiment so that I can identify specific issues.
5. As a developer, I want to validate the input fields so that only structured and meaningful data is stored in MongoDB.
6. As an organization, I want to apply sentiment analysis to feedback so that I can measure overall satisfaction levels.
7. As a business stakeholder, I want to track long-term trends in customer sentiment so that I can improve decision-making.

MVP Features

- User Feedback Form: A simple interface for collecting feedback.
- Feedback Storage: Secure backend API that stores feedback in MongoDB with timestamps.
- Admin API: Provides access to all stored feedback with filter options.
- Schema Validation: Ensures structured input (valid email, numeric rating, required fields).
- Data Security: Basic authentication for admin access.
- Optional Enhancements: Email notifications and sentiment analysis.

Wireframes / API Endpoint List

Wireframes (Conceptual):

1. Feedback Form Page: Input fields (Name, Email, Rating, Comments) and Submit button.
2. Admin Dashboard: Table displaying feedback entries with filter and search options.

API Endpoint List:

- POST /api/feedback → Accepts and stores new feedback.
- GET /api/feedback → Returns all feedback entries.
- GET /api/feedback?rating=5&date=2025-09-20 → Returns filtered feedback.

- GET /api/feedback/:id → Retrieves a single feedback entry by ID.
- Optional: POST /api/notify → Sends an email notification.
- Optional: GET /api/sentiment → Runs sentiment analysis.

Acceptance Criteria

- The system must allow users to submit feedback successfully.
- All feedback must be stored in MongoDB with unique ID and timestamp.
- Input must pass schema validation before storage.
- Admins must be able to view and filter feedback records.
- System should ensure only admins can access feedback data.
- Enhancements like email notifications and sentiment analysis can be added later.