11/21/2024

Survey Dashboard For VMH Group

SUBMITTED BY:

DHRUBJYOTI MONDAL

ANKUR DHOL

ROCKY KUNDU

ABHIGAYAN MUKHERJEE

SAMPRITI BHOWMIK

<u>INDEX</u>

ABSTRACT

The VMH Group Database Dashboard is a web-based platform designed to facilitate efficient project management by connecting clients, users, and administrators in a streamlined workflow. The primary objective of the system is to provide a centralized solution for project assignment, user management, and status tracking. Built using MySQL for data storage and PHP for backend operations, the dashboard supports real-time updates, user slot management, and secure data handling. Key functionalities include project assignment by admin, status tracking with dynamic updates, and quota fulfilment notifications, providing transparency and optimizing resource utilization.

INTRODUCTION

The VMH Group Database Dashboard is an interactive web application that enables effective management of multiple clients and users within the VMH Group platform. The dashboard serves as a central system where clients can provide projects, users can be assigned to complete these projects, and an admin oversees the entire workflow. This system aims to provide efficient coordination of resources and transparency in project statuses.

The role of clients, users, and admin in the system includes clients providing projects, users working on assigned projects, and the admin managing user assignments and ensuring smooth project progress.

SYSTEM REQUIREMENTS

- Hardware Requirements:

     Hosting server specifications for the web application, client devices (desktop, tablet, mobile).

- Software Requirements:
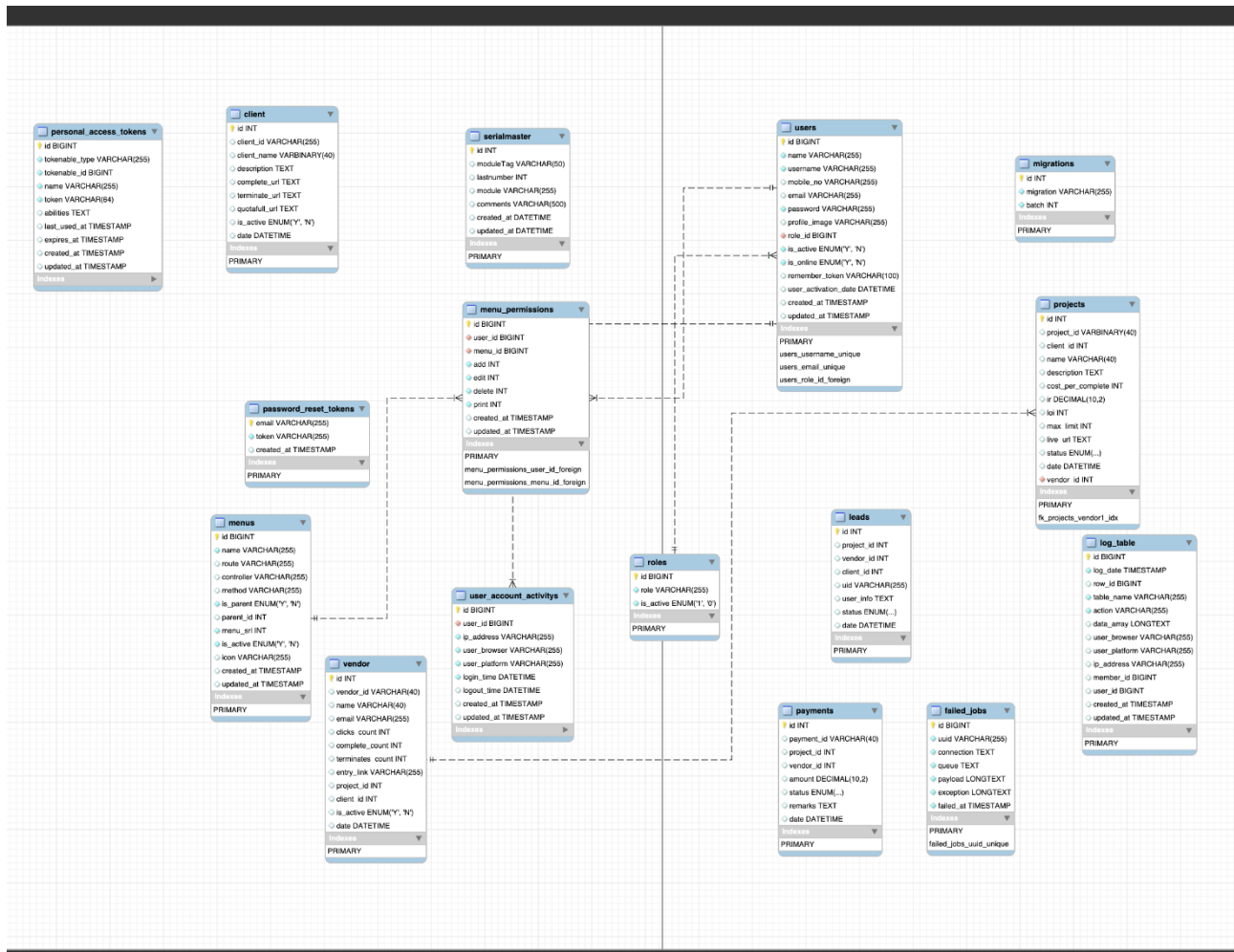
     PHP version 7.4 or above.

     MySQL version 5.7 or above.

     Apache or Nginx web server.

     Modern web browsers (Chrome, Firefox, Edge).

DATABASE DESIGN

ER Diagram: Illustrates entities such as Clients, Projects, Users, and their relationships.

Schema Design:

The database schema consists of several key tables to manage the project and user interactions efficiently:

1. Clients Table

  Purpose: Stores client details.
  Attributes:
    id (Primary Key): Unique identifier for each client.
    client_id: Unique client identifier.
    client_name: Client's name.
    description: Description of the client.
    complete_url, terminate_url, quotafull_url: URLs for respective project statuses.
    is_active: Indicates if the client is active (Y or N).
    date: Record creation date.

2. Projects Table

Purpose: Manages project-related data.
Attributes:
 id (Primary Key): Unique identifier for each project.
 project_id: Unique project identifier.
 client_id (Foreign Key): Links the project to a client.
 name: Project name.
 description: Project description.
 cost_per_complete: Cost associated with project completion.
 status: Indicates the project status (Live, Pause).
 date: Record creation date.

3. Users Table

Purpose: Maintains user information.
Attributes:
 id (Primary Key): Unique identifier for each user.
 name: User's full name.
 username: Unique username for login.
 email: User's email address.
 role_id (Foreign Key): Links to the roles table.
 is_active: Indicates if the user is active (Y or N).
 created_at, updated_at: Timestamps for record tracking.

4. Assignments Table

Purpose: Tracks user assignments to projects.
Attributes:
 id (Primary Key): Unique identifier for each assignment.
 project_id (Foreign Key): Links to the projects table.
 user_id (Foreign Key): Links to the users table.
 date: Assignment date.

5. Roles Table

Purpose: Defines roles for system users.
Attributes:
 id (Primary Key): Unique identifier for each role.
 role: Role name.
 is_active: Indicates if the role is active (1 or 0).

6. Leads Table

Purpose: Stores project leads.
Attributes:
  id (Primary Key): Unique identifier for each lead.
  project_id, client_id (Foreign Keys): Links to respective tables.
  status: Indicates lead status (Pending, Complete, etc.).
  date: Record creation date.

7. Menu Permissions Table

Purpose: Manages user permissions for system menus.
Attributes:
  id (Primary Key): Unique identifier.
  user_id, menu_id (Foreign Keys): Links to respective tables.
  add, edit, delete, print: Permissions flags (0 or 1).

8. Logs Table

Purpose: Records system actions for auditing.
Attributes:
  id (Primary Key): Unique log entry identifier.
  action: Description of the action performed.
  user_id: User who performed the action.
  created_at: Action timestamp.

FEATURES AND FUNCTIONALITIES

- **Real-Time Status Tracking**: Projects have three primary statuses: Pending, Complete, and Terminate. Additionally, the "Quota Fulfilled" status is automatically updated when the maximum number of users is assigned to a project, allowing admins and clients to see resource availability at a glance.

- **Project Assignment by Admin**: Admin can assign users to projects, managing user availability and ensuring that project requirements are met. Admins have full control over the allocation of users, which allows them to ensure that no project is under- or over-staffed, improving resource management.

- **User Slot Management**: The system ensures that projects do not exceed the specified user limit. Once a project reaches its assigned user capacity, it updates to "Quota Fulfilled" status, preventing further user assignments.

- **Customizable Reporting**: The dashboard allows clients and admins to generate tailored reports on project status, user assignments, and completion rates, making it easier to monitor progress and make informed decisions.

- **Multi-Device Compatibility**: The dashboard is accessible on desktop, tablet, and mobile devices, ensuring flexibility for users to manage and view project information from any device.

- **Data Visualization**: The platform incorporates charts, graphs, and other visualization tools to present project statuses, resource utilization, and completion rates clearly, enabling better understanding and insights for decision-making.

- **User-Friendly Interface**: Screenshots of the dashboard interface provide a visual representation of the client, user, and admin views, showcasing an intuitive and responsive design that improves overall usability.

- **Status Tracking**: Projects have three primary statuses: Pending, Complete, and Terminate. The "Quota Fulfilled" status is automatically updated when the maximum number of users is assigned to a project.

- **User Slot Management**: Ensures that projects do not exceed the specified user limit.

- **Screenshots of the Dashboard Interface**: Visual representation of client, user, and admin views within the dashboard shown as below.

a. The Admin Dashboard view & also the Home Page.
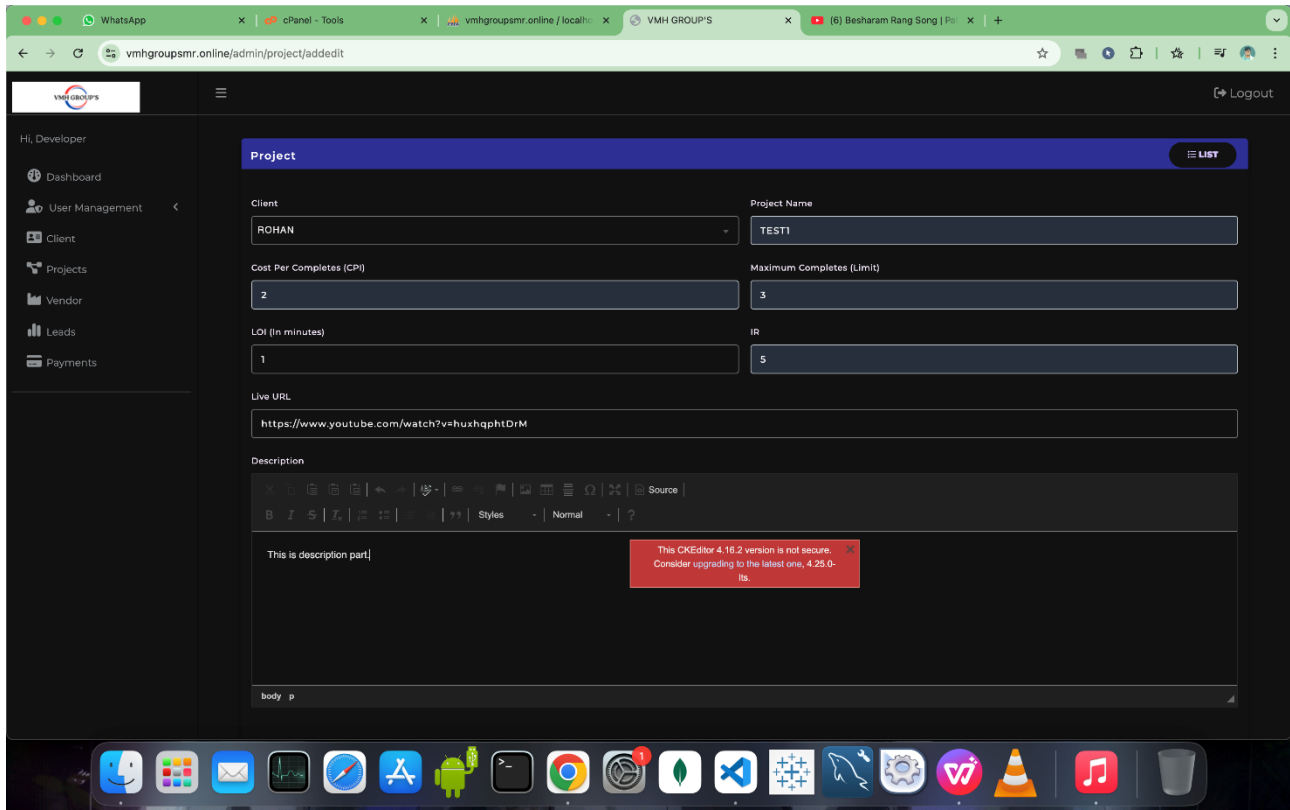
b. Visual Representation of Client Page.

IMPLEMENTATION

Assigning New Project & Vendor

Step1: Add details of new projects.
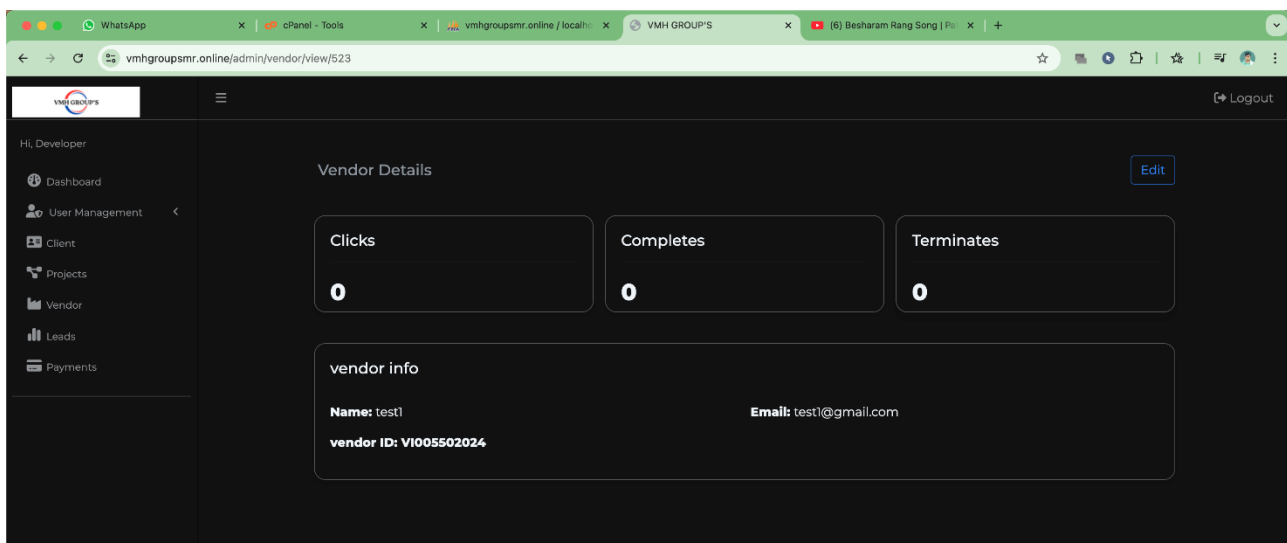

Step2: Project details successfully assigned.
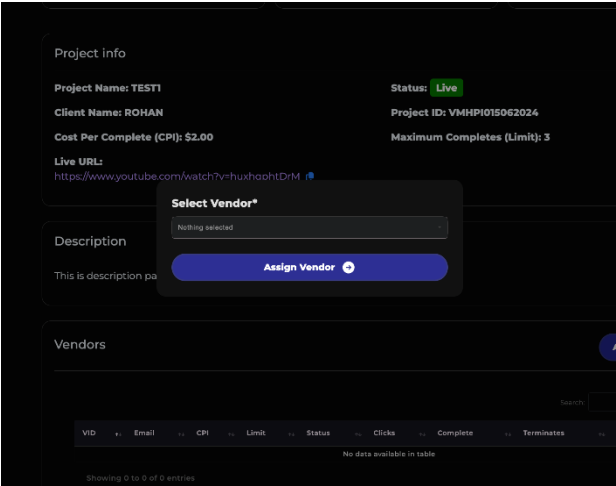
Step 3: Assigning the new vendor.
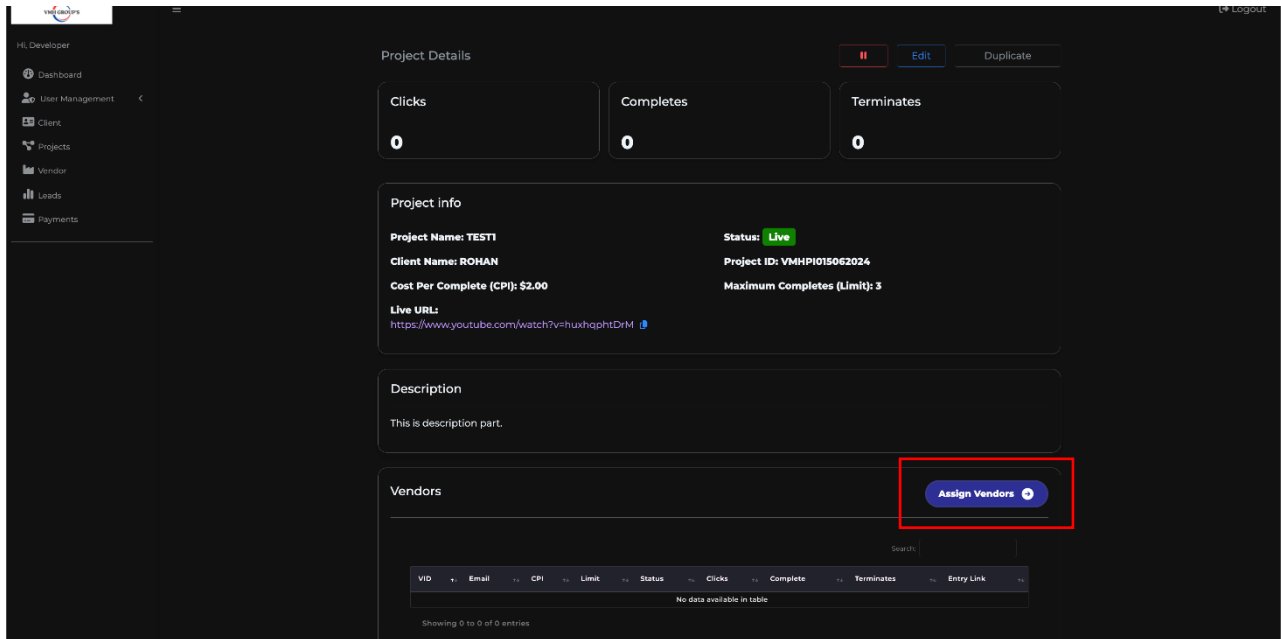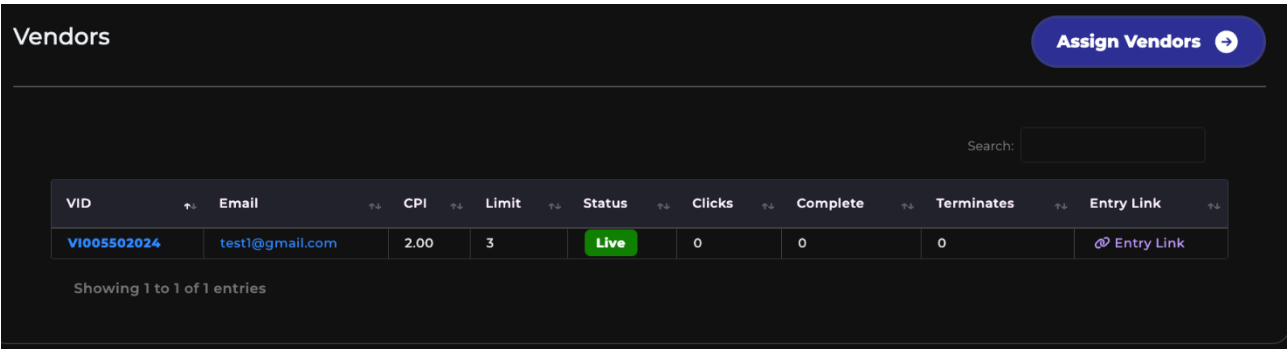

Step 4: Fill new vendors details.
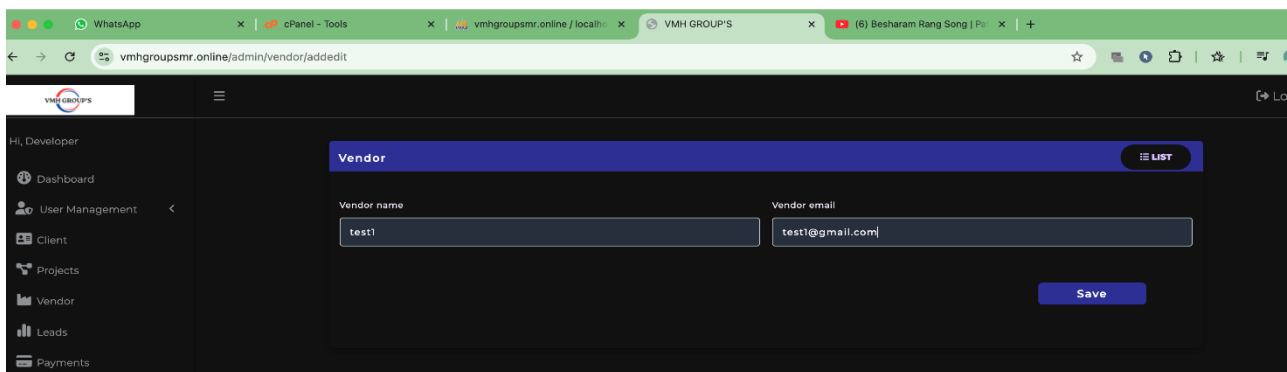
Step 5: Vendor Create successfully .

Step 6: Now search for new vendor in projects tab.
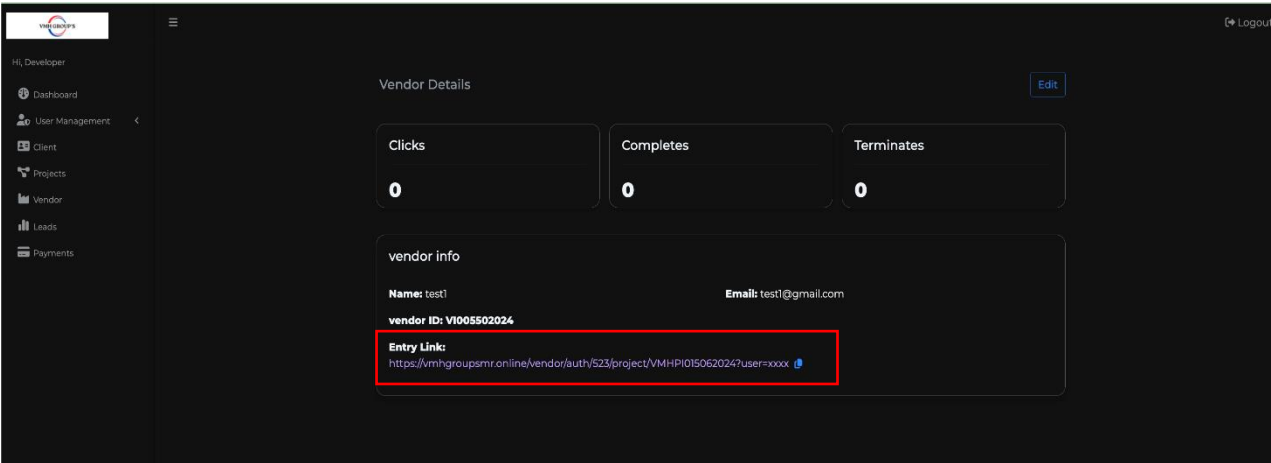

Step 7: Assigned the vendor in project.

Step 8: Vendor details with Entry link.

CHALLENGES AND SOLUTIONS

Challenges

- **Complex Relationships**: Managing relationships between clients, projects, users, and assignments was challenging due to the interconnected nature of the data.

- **Real-Time Updates**: Ensuring that the project statuses and user assignments were updated in real time required a robust backend logic and efficient data handling.

- **Scalability**: As the number of clients, projects, and users grows, ensuring that the system remains responsive posed scalability challenges.

- **Data Security**: Protecting sensitive client and project data from unauthorized access was a primary concern.

Solutions

- **Optimized SQL Queries**: Optimized SQL queries were implemented to improve data retrieval times and reduce the complexity of managing relationships between entities. Indexing was also used to speed up searches.

- **Database Indexing and Load Balancing**: To handle scalability, indexing was applied to frequently queried columns, and load balancing techniques were explored to distribute database queries effectively.

- **Data Encryption and Role-Based Access Control**: Sensitive data was encrypted using standard encryption methods, and role-based access control (RBAC) was implemented to restrict data access based on user roles, thereby enhancing security.

CONCLUSION

The VMH Group Database Dashboard successfully addresses the complexities of managing multiple clients, projects, and users in a unified platform. By providing features like real-time project status tracking, quota management, and customizable reporting, it ensures a transparent and efficient workflow. The use of MySQL and PHP for backend integration provides a robust foundation for data management, while AJAX enables a responsive user experience with real-time updates. Additionally, data security measures, including encryption and role-based access control, ensure that sensitive information is well-protected.

Moving forward, future enhancements will include integrating advanced analytics to track project success metrics, further improving data visualization tools for better insights, and ensuring that the platform is optimized for scalability as the number of users and projects grows. These improvements will help the system evolve to meet increasing demands and provide even greater value to clients and administrators.

APPENDICES

Project link: i. **Main project link (https://vmhgroupsmr.online/)**

ii. Presentation Link (https://sayanbanerjee-007.github.io/Dhrubo_Survey/)

Dashboard Screenshots: Visual representation of various views within the system, including user assignments, project statuses, and quota tracking.

1. Visual Representation of "User Management" tab.

2. Visual Representation of "projects Status" tab.

C . The "Payment" tab with proper Actions.

D . Visual Representation of "Vendor List".