FINAL PROJECT REPORT

Project Title: "Complete Progress Report Management System"

BAUN 6320 – Database Foundations for Business Analytics

Group Number: 3

Team Members:

Venkata Rama Anirudh Vikram Kesapragada
Megha Sai Sree Pinnaka
Neha Chowdary Uppala
Sowmya Yella
Pranav Bhanukumar
Bharath Dodda Lokesh

Student Name	Contribution
Pranav Bhanukumar (pxb220030)	Project Write-up, Tables, Data Insertion, Queries, Triggers, Stored Procedures, Functions
Bharath Dodda Lokesh (bxd220015)	Project Write-up, Tables, Queries, Triggers, Functions, Stored procedures, Video editing
Megha Sai Sree Pinnaka (mxp220072)	Project Write-up, Tables, Data Insertion into Tables, Functions, Video Script, Final Project Report
Venkata Rama Anirudh Vikram Kesapragada (vxk220066)	Project Write up, Tables, Data Insertion, Function, Query, Video Story line, Final Project Report
Neha Chowdary Uppala (nxu210006)	Project Write up, Tables, Video Script, Stored Procedure, Trigger, Final PPT
Sowmya Yella (sxy210058)	Project Write-up, Tables, Video, Final PPT

ACKNOWLEDGEMENT

We thank our respectable **Prof. Srikanth Kannan**, who provided us this challenging opportunity of learning which turned out to be a knowledgeable and in-depth learning experience for us regarding the Database Management Systems. During our journey, we came to learn all the nuances about mySQL and Databases in general. We hope to continue our journey with the strength and inspiration that you gave us.

Thank you.
Group 3

Advantages of Database Management System

There are many advantages of using a database management system (DBMS) to manage data. Here are some of the key advantages:

- ***Data Integrity**: A DBMS helps to ensure the accuracy and consistency of data by enforcing constraints, such as unique constraints and referential integrity constraints, and providing tools for data validation.*

- ***Data Security**: A DBMS provides a secure and controlled environment for storing and accessing data. Access to the data can be restricted by user roles and permissions, and sensitive data can be encrypted to protect it from unauthorized access.*

- ***Data Sharing**: A DBMS enables multiple users and applications to access and share the same data, without the risk of data conflicts or duplication. This makes it easier for organizations to collaborate and share data across different departments and teams.*

- ***Data Consistency**: A DBMS ensures that the data is consistent across different applications and systems, by providing tools for data integration and data transformation.*

- ***Data Scalability**: A DBMS can scale to handle large amounts of data and high levels of user traffic, without affecting the performance or reliability of the system.*

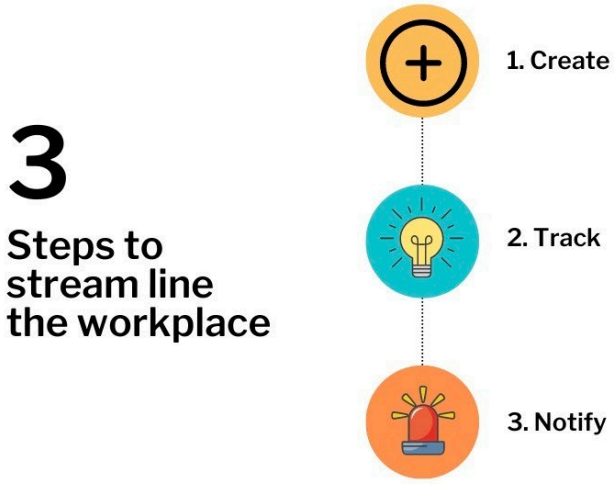
- ***Data Recovery**: A DBMS provides backup and recovery tools, to help restore data in case of system failures or disasters.*

Overall, a DBMS can help organizations to better manage their data, improve their decision-making processes, and increase their operational efficiency.

Business Overview

Many businesses and organizations struggle to efficiently manage their projects and stay on top of progress reporting. The lack of a centralized platform for creating, tracking, and managing progress reports can result in delays. To address this challenge, we offer **"Task Force"** a comprehensive software solution system that simplifies the process of progress reporting.

Our system streamlines progress reporting, improves communication among team members, and provides reporting options to better track progress on individual tasks. By using our platform, businesses and organizations can stay on track, identify potential issues, and hold team members accountable for their work. Ultimately increasing their chances of completing projects on time, within budget, and with high quality results.



Project Description

The Complete Progress Report Management System is a cutting-edge and all-encompassing solution created to give organizations a dependable and effective means to track, manage, and report on the progress of various projects inside an organization.

- Project managers can assess the status of their projects and spot any potential hazards or delays using the technology quickly and accurately.
- It gives stakeholders an up-to-date perspective of all project progress, assisting in ensuring that everyone agrees with the project's schedules and objectives.
- Manage project progress, monitor milestones, evaluate risks, and pinpoint areas for improvement, the project will establish a centralized platform.
- Enable stakeholders to readily convey project statuses in real-time, the system streamlines the process of creating and sharing reports. Due to its extensive feature set, the system aids companies in better project management and control, ensuring that goals are reached, and deadlines are met.

The Complete Progress Report Management System is designed to provide businesses with a cost-effective and efficient solution for ensuring timely project completion, while also keeping stakeholders well-informed and engaged. It will serve as a vital tool for project managers and stakeholders to track project progress, identify potential issues, and take necessary corrective actions.

Stakeholders

The Complete Progress Report Management System is a centralized database which can be accessible by an organization or business striving to achieve an efficient and comprehensive platform to manage their projects. This may include project managers, team leaders, external stakeholders, and other relevant personnel who are responsible for overseeing and monitoring the progress of a project.

Business Proposal

We are a company that offers services to individual users to manage their music data more precisely and allow them to get real-time updates about their songs, artists, and other active users as well. We, as a company, specialize in creating databases for personal users and allowing the user to manage them. With a driven team of problem solvers, we believe in giving the best services to our clients and enhancing their current business processes.

Background Information

Under the direction of our Professor Srikanth Kannan, six graduate students from The University of Texas in Dallas created this system. By showcasing our management system's competence in business development, strategic planning, and analytics, we hope to demonstrate its business acumen.

Project Deliverables

Project Deliverables for establishing a Complete Progress Report Management database which will be used for creating a centralized system for tracking projects are mentioned below:

- **Project Information:** Project details like the name, start date, budget, and description will be part of the database.
- **Time Tracking:** The time tracking functionality enables team members to record the time spent on projects or tasks, providing the means to bill clients, measure efficiency and evaluate performance.
- **Report Approval:** Reports submitted by team members can be approved or rejected by end users with appropriate permissions and access.
- **Task Management:** The task management functionality permits the creation of tasks, assignment to team members, setting of deadlines, and monitoring of progress. Additionally, team members can indicate when a task is complete and inform others about any changes to task status.

- **Automated Reminder:** Reminder will be sent to resources when reports are due or when the reports must be sent out to internal and external stakeholders.
- **Reporting:** Ability to generate performance, progress, budget, and other custom reports based on project data. End-users will be able to view the history of progress reports, including previous versions, revisions, and updates.
- **Collaboration:** Team members can collaborate and communicate with stakeholders effectively through messaging, email integration and discussion forums.
- **Access Control:** The access control functionality empowers project managers to manage access to project information and documents. The feature may comprise user roles and permissions, password protection, and two-factor authentication.
- **Analytics:** Insights into project progress, resource allocation and overall project performance.

Project Requirements

- **Data model:** Capturing all data elements, including projects, tasks, employees, progress reports, budgets, risk assessments, and issue logs.
- **Database Design:** Develop the database design, including tables, fields, and relationships between tables.
- **Data Migration:** Migrating existing data into the new database.
- **Development:** Develop the system using appropriate technologies and tools, such as SQL, Python, HTML, CSS, and JavaScript, and follow best practices for coding, testing, and documentation.
- **User Interface:** Development of an intuitive and user-friendly user interface which involves visual boards and graphs.
- **Budget:** Allocate sufficient budget for the development, implementation, and maintenance of the system, including hardware, software, and personnel costs.
- **System Deployment:** The system should be deployed on a secure and reliable platform, whether it's a dedicated server or cloud-based service and should be made accessible only to authorized users.
- **Testing:** Perform comprehensive testing of the system to ensure that it meets all requirements, is user-friendly, and is free of bugs and errors.
- **Training and Support:** Comprehensive training and support should be provided to users, which includes user manuals, online tutorials, and help desk services, to ensure their proficiency and satisfaction with the system.
- **Continuous Improvement:** The system's performance, usability, and effectiveness should be continuously monitored and evaluated.

Milestones

- This mechanism enables ease of tracking various projects in an organization for managers, clients and various members included in a project.
- It also grants required access to organization’s members depending on their role which ensures security.
- This allows ease of allocation of the projects to the eligible employee based on their profile.

- This also helps resolve various issues by other departments by creating a tracking request information.
- Our Complete Progress Report Management System is a centralized approach for project management in an organization.

Task Force vs Competitors

There are several competitors in the market for progress report management systems such as Asana, Trello, Microsoft Project, smart sheet, and Jira are a few well-known rivals in this market.

- Creating a more centralized system for data management was our focus to set the proposed system apart from existing solutions.
- With the help of this centralized platform, business organizations can stay on track, identify the problems they are having, and assign the team members tasks that will increase the likelihood that the projects will be finished effectively and on time.
- Hence, the difference between the current system and the one we are attempting to establish is that we are focusing on businesses or use cases. We are developing a progress report management system, for instance, that is intended especially for commercial enterprises.
- In the end, the secret to sticking out in a crowded market is to pinpoint customer wants and create a product that solves their problems. So, we made the decision to go with this centralized system that is focused on progress reporting, with features that enable easy creation, tracking, and management of progress reports. Additionally features of analytics, dashboard and reporting will be an added plus.

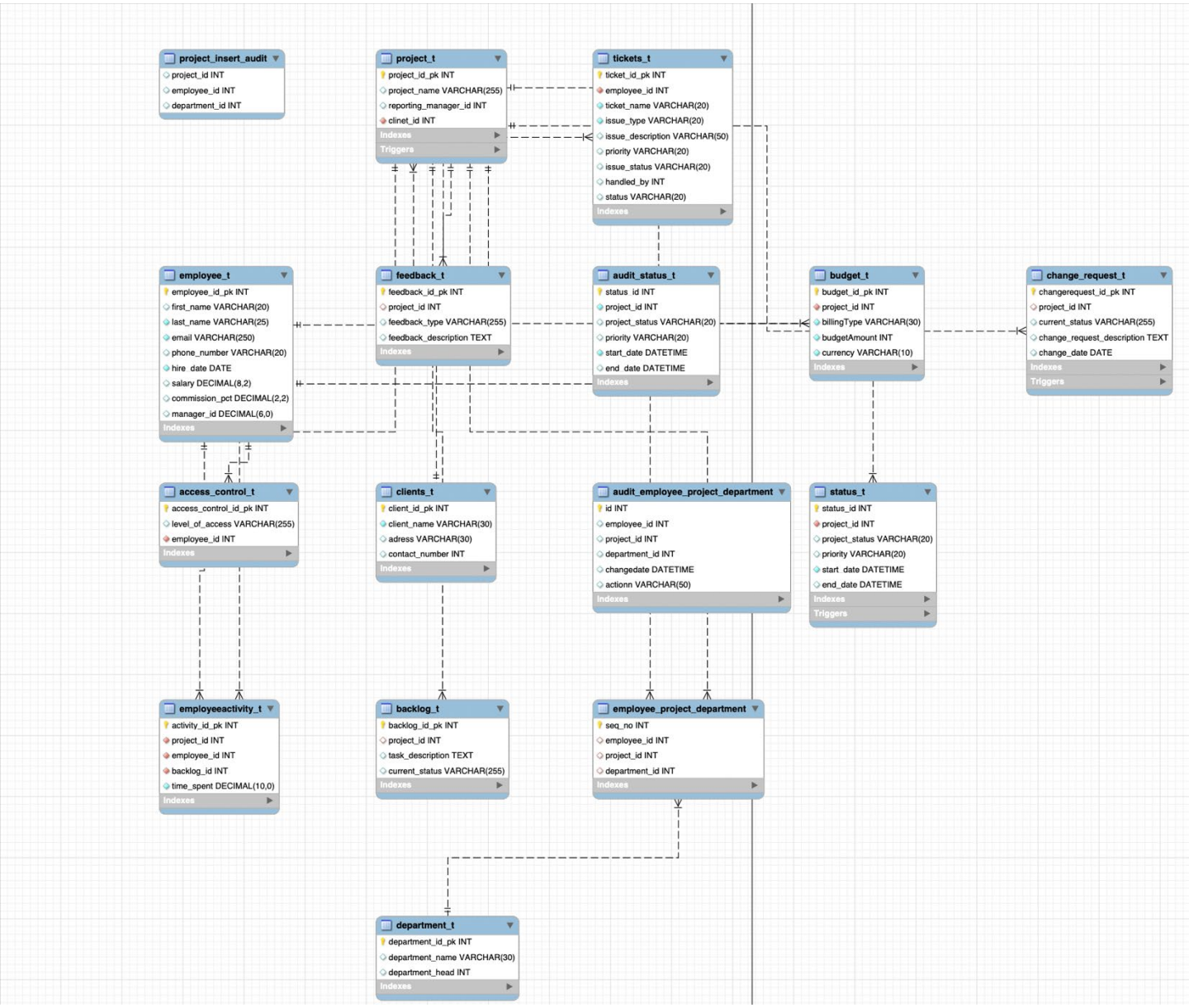
Product Features

If a user interface (UI) is built, the Complete Progress Report Management System can be offered as a product with the following features:

- **Analytics and Reporting:** The analytics and reporting module provides users with the ability to create different types of reports and visual aids for analyzing project data, monitoring progress, and recognizing patterns. This includes visualizations such as Gantt charts, pie charts, and bar graphs.
- **Integration:** To enhance efficiency and streamline workflows, it is recommended to integrate the system with existing tools and systems, including email clients, project management software, and productivity tools.
- **Dashboard:** The product may provide a personalized dashboard, which offers real-time updates and notifications, to give an overview of ongoing projects, task assignments, and progress reports.
- **Customer Support:** The product can provide customer support services such as user training, bug fixes, and upgrades.
- **Mobile-Friendly:** Users can access reports and data on their mobile devices through a mobile-friendly UI design.


Task Management
Access Control
Project Information
Report Approval
PROGRESS REPORT MANAGEMENT
Time Tracking
Collabration
Notification
Reporting

E R Diagram


project_insert_audit
project_id INT
employee_id INT
department_id INT
project_t
project_id_pk INT
project_name VARCHAR(255)
reporting_manager_id INT
clinet_id INT
Indexes
Triggers
tickets_t
ticket_id_pk INT
employee_id INT
ticket_name VARCHAR(20)
issue_type VARCHAR(20)
issue_description VARCHAR(50)
priority VARCHAR(20)
issue_status VARCHAR(20)
handled_by INT
status VARCHAR(20)
Indexes
employee_t
employee_id_pk INT
first_name VARCHAR(20)
last_name VARCHAR(25)
email VARCHAR(250)
phone_number VARCHAR(20)
hire_date DATE
salary DECIMAL(8,2)
commission_pct DECIMAL(2,2)
manager_id DECIMAL(6,0)
Indexes
feedback_t
feedback_id_pk INT
project_id INT
feedback_type VARCHAR(255)
feedback_description TEXT
Indexes
audit_status_t
status_id INT
project_id INT
project_status VARCHAR(20)
priority VARCHAR(20)
start_date DATETIME
end_date DATETIME
Indexes
budget_t
budget_id_pk INT
project_id INT
billingType VARCHAR(30)
budgetAmount INT
currency VARCHAR(10)
Indexes
change_request_t
changerequest_id_pk INT
project_id INT
current_status VARCHAR(255)
change_request_description TEXT
change_date DATE
Indexes
Triggers
access_control_t
access_control_id_pk INT
level_of_access VARCHAR(255)
employee_id INT
Indexes
clients_t
client_id_pk INT
client_name VARCHAR(30)
adress VARCHAR(30)
contact_number INT
Indexes
audit_employee_project_department
id INT
employee_id INT
project_id INT
department_id INT
changedate DATETIME
actionn VARCHAR(50)
Indexes
status_t
status_id INT
project_id INT
project_status VARCHAR(20)
priority VARCHAR(20)
start_date DATETIME
end_date DATETIME
Indexes
Triggers
employeeactivity_t
activity_id_pk INT
project_id INT
employee_id INT
backlog_id INT
time_spent DECIMAL(10,0)
Indexes
backlog_t
backlog_id_pk INT
project_id INT
task_description TEXT
current_status VARCHAR(255)
Indexes
employee_project_department
seq_no INT
employee_id INT
project_id INT
department_id INT
Indexes
department_t
department_id_pk INT
department_name VARCHAR(30)
department_head INT
Indexes

Tables

- ***feedback_t :*** feedback_id_pk, project_id, feedback_type, feedback_description
- ***change_request_t :*** changerequest_id_pk, project_id, current_status, change_request_description, change_date
- ***employeeactivity_t :*** activity_id_pk, project_id, employee_id, backlog_id, time_spent
- ***employee_project_department :*** seq_no, employee_id, project_id, department_id
- ***backlog_t :*** backlog_id_pk, project_id, task_description, current_status
- ***access_control_t :*** access_control_id_pk, level_of_access, employee_id
- ***tickets_t :*** ticket_id_pk, employee_id, ticket_name, issue_type, issue_description, priority, issue_status, handled_by, status
- ***employee_t :*** employee_id_pk, first_name, last_name, email, phone_number, hire_date, salary, commission_pct, manager_id
- ***department_t :*** department_id_pk, department_name, department_head
- ***status_t :*** status_id, project_id, project_status, priority, start_date, end_date
- ***budget_t :*** budget_id_pk, project_id, billingType, budgetAmount, currency
- ***project_t :*** project_id_pk, project_name, reporting_manager_id, clinet_id
- ***clients_t :*** client_id_pk, client_name, address, contact_number
- ***project_insert_audit:*** project_id, employee_id, department_id, changedate, action
- ***audit_employee_project_department:*** id, employee_id, project_id, department_id, changedate, action
- ***audit_status_t:*** status_id, project_id, project_status, priority, start_date, end_date, changedate, action
- ***audit_backlog_t:*** backlog_id_pk, project_id, task_description, current_status, changedate, actionn

Stored Procedures

get_details_for_project : To get the information about the number of tasks for a specific project, back logs, number of employees working on the project , no of change requests, revenue generated this stored procedure is created.

get_details_for_clients : To get all the number of projects, total budget, project status, backlogs for a client this stored procedure is used.

assign_ticket_to_employee_and_update_status: This stored procedure is used to assign ticket to employee and update the status.

Functions

get_no_emp_working_on_proj : Function to get number of employees working on a project.

get_backlog_count : Function to get backlog count for a project.

calculate_billing_amount : Function to calculate the billing amount.

get_employee_with_least_work : Function to get the number of employees who have done minimal or least work when compared to the other employees.

is_High_priority_projects: This function returns either 1 or 0. 1 indicating that is a high priority project 0 indicating that it is not a high priority project.

get_project_with_highest_backlogs: This function can be used to retrieve the project with highest number of backlogs.

Triggers

project_insert_trigger : Created this trigger for adding an employee when a new project is added. We need one function for getting the employee with the least amount of work.
The function created get_employee_with_least_work it needs 1 parameter to find the employee, with least work, we need to send the manager id.

free_employees : This trigger is created to free all employees once status of a project turns completed, which means that when project_status is updated to completed.

client_feedback_backlog_auto_add : To add the client feed back to backlogdb this trigger is used.

ticket_assignment_audit_trigger: This is a ticket assignment audit trigger used to record ticket assignments in the audit_employee_project_department table for auditing purposes.

ticket_assignment_and_status_audit_trigger: This trigger is used to record ticket assignments, as well as changes in ticket status, in the audit_employee_project_department table for auditing purposes.

Queries

```
/*
This query retrieves information about the projects that meet certain criteria
 for budget, number of employees, and total time spent.
*/
SELECT
  c.client_name AS Client_Name,
  p.project_name AS Project_Name,
  SUM(b.budgetAmount) AS Total_Budget,
  COUNT(DISTINCT e.employee_id_pk) AS Number_of_Employees,
  COUNT(DISTINCT t.ticket_id_pk) AS Number_of_Tickets,
  ROUND(SUM(ea.time_spent), 2) AS Total_Time_Spent
FROM clients_t c
JOIN project_t p ON p.clinet_id = c.client_id_pk
JOIN budget_t b ON b.project_id = p.project_id_pk
JOIN employee_project_department epd ON epd.project_id = p.project_id_pk
JOIN employee_t e ON e.employee_id_pk = epd.employee_id
JOIN tickets_t t ON t.employee_id = e.employee_id_pk
JOIN employeeactivity_t ea ON ea.employee_id = e.employee_id_pk AND ea.project_id =
p.project_id_pk
GROUP BY c.client_name, p.project_name
HAVING Total_Budget > 1000 AND Number_of_Employees > 1 AND Total_Time_Spent > 5
ORDER BY Total_Budget DESC;


/*Query to find all employees who have worked on multiple projects and their total hours worked.*/

SELECT CONCAT(employee_t.first_name, ' ', employee_t.last_name) AS employee_Full_Name,
sum(employeeactivity_t.time_spent)
AS No_of_Hours, Count(project_t.project_id_pk) AS No_of_Projects
from employee_t
INNER JOIN employeeactivity_t
ON employee_t.employee_id_pk = employeeactivity_t.employee_id
INNER JOIN project_t ON employeeactivity_t.project_id = project_t.project_id_pk
GROUP BY employee_Full_Name HAVING No_of_Projects > 1;

/*Query to find all tasks that have not been completed and the employees assigned to them*/

SELECT project_t.project_name, CONCAT(employee_t.first_name, ' ', employee_t.last_name) AS
employee_Full_Name
```

```
FROM project_t INNER JOIN employeeactivity_t ON project_t.project_id_pk =
employeeactivity_t.project_id
INNER JOIN status_t ON employeeactivity_t.project_id = status_t.project_id
INNER JOIN employee_t ON employeeactivity_t.employee_id = employee_t.employee_id_pk
WHERE project_status = 'In Progress' OR project_status = 'Pending';

/*To get a list of all projects, the employees assigned to those tasks, and the departments those
employees belong to */

SELECT project_t.project_name, CONCAT(employee_t.first_name, ' ', employee_t.last_name) AS
employee_Full_Name, department_t.department_name
FROM project_t INNER JOIN employee_project_department ON project_t.project_id_pk =
employee_project_department.project_id
INNER JOIN department_t ON employee_project_department.department_id =
department_t.department_id_pk
INNER JOIN employee_t ON employee_project_department.employee_id =
employee_t.employee_id_pk;

/*To get a list of all projects, the employees assigned to those projects and the departments they are
associated, but only for tasks that are currently active*/

SELECT project_t.project_name, CONCAT(employee_t.first_name, ' ', employee_t.last_name) AS
employee_Full_Name, department_t.department_name
FROM project_t INNER JOIN employee_project_department ON project_t.project_id_pk =
employee_project_department.project_id
INNER JOIN department_t ON employee_project_department.department_id =
department_t.department_id_pk
INNER JOIN employee_t ON employee_project_department.employee_id =
employee_t.employee_id_pk
INNER JOIN status_t ON project_t.project_id_pk = status_t.project_id
WHERE project_status = 'In Progress';

/*Subquery for Tickets_T table: This query retrieves all tickets belonging to the IT department */

SELECT ticket_name, Issue_type FROM tickets_t WHERE employee_id IN
(SELECT employee_id FROM employee_project_department WHERE department_id =
(SELECT department_id_pk FROM department_t WHERE department_name = 'IT'));

/*Subquery for Department_T table: This query retrieves all departments and their budgets where the
budget amount
is higher than the average budget amount for all departments*/

SELECT department_name, budget_t.budgetAmount FROM department_t
JOIN employee_project_department ON department_t.department_id_pk =
employee_project_department.department_id
JOIN budget_t ON employee_project_department.project_id = budget_t.project_id
WHERE budgetAmount > (SELECT AVG(budgetAmount) FROM budget_t);
```

```
/* Subquery for Clients_T table: This query retrieves all employees with their total number of tickets
and only includes
clients whose budget amount is higher than 20,000*/

SELECT clients_t.client_name, CONCAT(employee_t.first_name,' ', employee_t.last_name) AS
EMP_FULL_NAME, COUNT(tickets_t.ticket_id_pk)
AS No_of_tickets FROM employee_t
JOIN tickets_t ON employee_t.employee_id_pk = tickets_t.employee_id
JOIN employee_project_department ON tickets_t.employee_id =
employee_project_department.employee_id
JOIN project_t ON employee_project_department.project_id = project_t.project_id_pk
JOIN clients_t ON project_t.clinet_id = clients_t.client_id_pk
WHERE project_id IN (SELECT budgetAmount FROM budget_t WHERE budgetAmount > 12000) GROUP
BY clients_t.client_name, EMP_FULL_NAME;

/*Write a query retrieves all employee activities for a specific employee(identified by employee_id)
that took more than 8 hours,
along with the associated change request title and status */

SELECT project_t.project_name, change_request_t.change_request_description,
employeeactivity_t.backlog_id, employeeactivity_t.time_spent
FROM employeeactivity_t LEFT JOIN change_request_t ON employeeactivity_t.project_id =
change_request_t.project_id
LEFT JOIN project_t ON change_request_t.project_id = project_t.project_id_pk
WHERE (employeeactivity_t.employee_id = 4 AND employeeactivity_t.time_spent > 2);

/*Write a query that retrieves all feedback entries, along with the associated employee */

SELECT feedback_t.feedback_type, feedback_t.feedback_description,
CONCAT(employee_t.first_name,' ', employee_t.last_name) AS EMP_FULL_NAME FROM feedback_t
JOIN project_t ON feedback_t.project_id = project_t.project_id_pk
LEFT JOIN change_request_t ON project_t.project_id_pk = change_request_t.project_id
JOIN employee_project_department ON change_request_t.project_id =
employee_project_department.project_id
RIGHT JOIN employee_t ON employee_project_department.employee_id =
employee_t.employee_id_pk WHERE department_id = 10;

/*Write a query that retrieves all project names, empolyee names and their access control where the
project budget is greater than 5000 */
select project_t.project_name, CONCAT(employee_t.first_name,' ', employee_t.last_name) AS
EMP_FULL_NAME, access_control_t.level_of_access FROM project_t
JOIN employee_t ON project_t.reporting_manager_id = employee_t.manager_id
JOIN access_control_t ON employee_t.employee_id_pk
JOIN budget_t ON project_t.project_id_pk = budget_t.project_id
WHERE budget_t.budgetAmount > 5000;
```

```
/*
This query joins the clients_t, project_t, and backlog_t tables, and groups the results by the client name.
It uses the COUNT() function to count the number
of projects for each client, and the MAX() function to get the latest backlog_id_pk for each client.
*/
SELECT c.client_name, COUNT(p.project_id_pk) as num_projects, MAX(b.backlog_id_pk) as latest_backlog_id
FROM clients_t c
LEFT JOIN project_t p ON c.client_id_pk = p.clinet_id
LEFT JOIN backlog_t b ON b.project_id = p.project_id_pk
GROUP BY c.client_name
HAVING num_projects > 1 AND latest_backlog_id IS NOT NULL;
```

Limitations

- **Dependence on User Input:** Inaccurate or incomplete data can affect the validity of the reports and the effectiveness of the system.
- **Technical Expertise:** The system requires high end technical support to build, configure and maintain.
- **Integration and limited customization:** The system may face challenges in integrating with certain legacy systems that have limited compatibility with modern technologies. Although the system might have several functionalities, it may not be able to address each unique needs of a business.

Addressing the limitations

- **Regular Monitoring and Training:** Implementing data validation rules, providing user-friendly data entry interfaces, and conducting regular data audits to identify and correct errors.
- **Outsourcing:** Development and deploying of the system can be outsourced to an agency specializing in technical functionality.
- **Partnerships:** Partnerships with third-party providers or open-source solutions can help to reduce costs and increase the scalability of the system.

Conclusion

- Progress Report Management system covers various aspects of project management, including client and employee management, budget allocation, project tracking, and feedback.
- It also includes a ticketing system, audit trails, reporting, and analytics, making it a useful platform for businesses to manage the data associated with their projects.
- Effectively manipulates and analyzes data by including complex database functions like stored procedures, triggers, joins, and subqueries.

- It gives enterprises the ability to decide based on facts in order to maximize project performance.

Future Scope

- **Customization and personalization**: Customization and Personalization capabilities can be added to the system to let users adapt the progress reports to their requirements. This may include custom report formats, templates, and dashboards.
- **Integration with emerging technologies**: The progress report management system database can be integrated with developing technologies to improve its functionality and capabilities, such as blockchain, the Internet of Things (IoT), and augmented reality.