# 22072226 Sujan Prasad Bajgain

Islington College,Nepal

## Document Details

Submission ID

trn:oid:::3618:84650167

Submission Date

Mar 6, 2025, 7:30 AM GMT+5:45

Download Date

Mar 6, 2025, 7:33 AM GMT+5:45

File Name

22072226 Sujan Prasad Bajgain

File Size

42.0 KB

68 Pages

5,849 Words

35,198 Characters

# 55% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Match Groups

**131**Not Cited or Quoted 54%
Matches with neither in-text citation nor quotation marks

1 Missing Quotations 0%
Matches that are still very similar to source material

4 Missing Citation 1%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

## Top Sources

| | |
|---|---|
| 33% | Internet sources |
| 9% | Publications |
| 54% | Submitted works (Student Papers) |

## Integrity Flags

**0 Integrity Flags for Review**

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Match Groups

**131**Not Cited or Quoted 54%
Matches with neither in-text citation nor quotation marks

**1** Missing Quotations 0%
Matches that are still very similar to source material

**4** Missing Citation 1%
Matches that have quotation marks, but no in-text citation

**0** Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

## Top Sources

| | |
|---|---|
| 33% | Internet sources |
| 9% | Publications |
| 54% | Submitted works (Student Papers) |

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

| | | | |
|---|---|---|---|
| 1 | Submitted works | islingtoncollege on 2024-12-30 | 19% |
| 2 | Submitted works | University of New South Wales on 2024-07-29 | 5% |
| 3 | Submitted works | University of Maryland, Global Campus on 2023-05-04 | 4% |
| 4 | Submitted works | University of Suffolk on 2021-01-04 | 4% |
| 5 | Submitted works | Harrisburg University of Science and Technology on 2015-11-02 | 3% |
| 6 | Internet | www.coursehero.com | 3% |
| 7 | Submitted works | University of West London on 2023-05-27 | 2% |
| 8 | Submitted works | unibuc on 2025-02-04 | 2% |
| 9 | Internet | www.cnblogs.com | 1% |
| 10 | Submitted works | RMIT University on 2024-08-18 | <1% |

| | | | |
|---|---|---|---|
| 11 | Internet | pastebin.com | <1% |
| 12 | Submitted works | unibuc on 2025-01-24 | <1% |
| 13 | Submitted works | Informatics Education Limited on 2010-11-06 | <1% |
| 14 | Submitted works | Staffordshire University on 2017-09-30 | <1% |
| 15 | Submitted works | Leeds Beckett University on 2024-03-18 | <1% |
| 16 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 17 | Submitted works | University of Northumbria at Newcastle on 2024-12-05 | <1% |
| 18 | Submitted works | Vilniaus kolegija on 2025-01-29 | <1% |
| 19 | Submitted works | Temple University on 2024-11-08 | <1% |
| 20 | Publication | de Lemos Almeida Gomes, Maria José. "Aplicação da Metodologia Scrum na Gestã... | <1% |
| 21 | Submitted works | Southampton Solent University on 2011-01-21 | <1% |
| 22 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 23 | Submitted works | Sheffield Hallam University on 2013-08-06 | <1% |
| 24 | Submitted works | University of Newcastle on 2019-11-25 | <1% |

| | | | |
|---|---|---|---|
| 25 | Submitted works | University of Maryland, Global Campus on 2023-11-08 | <1% |
| 26 | Internet | gist.github.com | <1% |
| 27 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 28 | Internet | myfik.unisza.edu.my | <1% |
| 29 | Submitted works | University of Greenwich on 2012-10-02 | <1% |
| 30 | Submitted works | University of Wolverhampton on 2020-06-05 | <1% |
| 31 | Submitted works | University of Essex on 2023-04-28 | <1% |
| 32 | Submitted works | Wright State University on 2018-04-27 | <1% |
| 33 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 34 | Submitted works | De Montfort University on 2024-11-14 | <1% |
| 35 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 36 | Submitted works | Informatics Education Limited on 2010-07-31 | <1% |
| 37 | Submitted works | University of Waikato on 2021-05-07 | <1% |
| 38 | Submitted works | islingtoncollege on 2024-12-30 | <1% |

| | | | |
|---|---|---|---|
| 39 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 40 | Submitted works | Manchester Metropolitan University on 2010-04-17 | <1% |
| 41 | Internet | thekeep.eiu.edu | <1% |
| 42 | Submitted works | Loughborough University on 2024-12-08 | <1% |
| 43 | Submitted works | Missouri University of Science and Technology on 2020-02-03 | <1% |
| 44 | Submitted works | Seabreeze High School on 2010-10-05 | <1% |
| 45 | Submitted works | The Hong Kong Polytechnic University on 2010-04-15 | <1% |
| 46 | Submitted works | University of Glamorgan on 2019-04-01 | <1% |
| 47 | Submitted works | Academy of Information Technology on 2019-02-17 | <1% |
| 48 | Submitted works | Southern New Hampshire University - Continuing Education on 2023-06-25 | <1% |
| 49 | Submitted works | University of London External System on 2020-07-22 | <1% |
| 50 | Submitted works | University of Wales Institute, Cardiff on 2017-09-12 | <1% |
| 51 | Submitted works | islingtoncollege on 2024-12-30 | <1% |
| 52 | Submitted works | unibuc on 2025-01-11 | <1% |

1

## Introduction

This coursework covers the strategy and implementation of a project management system for LS Corporation, a mid-sized technology company, to solve problems like poor communication, inefficiencies, difficulty tracking project progress, and managing multiple projects.

With the increase in corporate size, designing and implementing a robust and scalable system is important. The database designer and developer responsible for designing and implementing a robust system should ensure that the LS Corporation project management system is organized, and scalable and should be free from anomalies and redundancy.

1

## Aims and Objectives

The main aim is to design and implement a strong and scalable project management system that centralizes data, streamlines task assignment, and improves project visibility using the ASP.net webform, Oracle SQL Developer Data Modeler.

The objective of this coursework is to:

Developed and implemented logic for the project management system.

Centralize data.

Make it easy to track progress and task assignment

Improve visibility of projects.

Improved team communication in multiple projects.

Enhance skill in ASP.Net with C#

Use knowledge of SQL and database design, and implementation.

## Business Rule

The business rules to design and implement of project management system for LS Corporation are:

A user can be part of multiple projects.

A project will have multiple tasks.

A user can be assigned multiple tasks, and multiple users are associated with a single task.

Each task can have multiple sub-tasks, but each sub-task is linked with a single task.

51 Each project is broken down into multiple milestones.

1

Initial ERD

An Entity Relationship Diagram is a graphical and diagrammatic representation of system entities and their relationship, a model of data that represents the framework infrastructure of an entity. This is a simple model and easier to understand by the developer. (Rouse, 2017). The components of the ER model are:

Entity

An entity is a real-world object that exists and is a table in the database with each row representing an entity instance. A rectangle with an entity name inside is used to represent an entity. The two types of entities are tangible which exist in the real world physically. Intangible which does not exist physically but exists logically. (Rouse, 2017)

Figure 1: Entity (Thakur, 2016 )

Attributes

An attribute is a property that describes the entity. It is represented by an oval shape with attributes inside. (Rouse, 2017)

Figure 2: Attribute (Thakur, 2016 )

Relationship is the bond that link 2 or more that 2 entities. A diamond symbol represents a relationship. (Rouse, 2017)

Figure 3: Relationship (Thakur, 2016 )

Entity and Attributes

From the scenario and table, six main entities are identified before normalization.

User

The user is the tangible entity that contains the user information such as User_ID, User Name, User Email, User Contact, Role, etc.

Figure 4: User entity and its attributes.

Project

The project is the intangible entity that contains details about the project like Project_ID, Project Name, Project Start Date, Project Due Date, Project Status, etc.

Figure 5: Project entity and its attributes.

Task

The task is the intangible entity that contains information about the task like Task_ID, Task Name, Start Date, Due Date, Status, etc.

Figure 6: Task entity and its attributes.

Sub Task

The sub-task is the intangible entity that has the information of task work breakdown like STask_ID, STask Name, etc.

Figure 7: Sub Task entity and its attributes.

Milestone

The milestone is the entity that contains the information of the project work breakdown for each milestone.

Figure 8: Milestone entity and attributes.

Identification of Primary and Foreign Key

The primary key is the unique ID that holds the unique details and information of each row whereas the foreign key is linked with the primary key of the table to show the relationship between the entities.

Form the scenario and table the Primary and Foreign key:

Table 1: Primary key and Foreign key

Entities

Primary Key

Foreign Key

References

Users

User_ID

Project

Project_ID

User_ID

User

MileStone

Milestone_ID

Project_ID

Project

Task

Task_ID

Project_ID

Project

Sub_Task

STask_ID

Task_ID

Task

Entity Relationship Diagram

Figure 9: Initial ERD

In the given initial ERD, there exist many to many, and high redundancy chances which leads to data anomalies and data inconsistencies. The user information is likely to be repeated. This ERD might not be adjusted properly according to business rules which makes the database implementation and design ineffective for the system.

Normalization

Normalization is the process of database design that reduces data anomalies (insert, update, and delete) and redundancy. This is the multi-step process that helps to organize the data and maintains data integrity, and data consistency. It checked and removed dependency as well. (Kitthu, 2024)

UNF

This is the unnormalized form where all the repeated groups can be found.

User (User_ID, User Name, User Email, User Contact, Role,{Project_ID, Project Name, Project Start Date, Project Due Date, Project Status,{Task_ID, Task Name, Start Date, Due Date, Status}})

1 1NF

In the first normal form, the repeated groups are removed and placed separately as entities which contain unique record. (Allen, 2024)

The 1NF is:

From the UNF the repeated groups and separated in different entity with unique records.

User-1 (User_ID, User Name, User Email, User Contact, Role)

Project-1 (Project_ID, Project Name, Project Start Date, Project Due Date, Project Status, User_ID*)

Task-1 (Task_ID, Task Name, Start Date, Due Date, Status, Project_ID*, User_ID*)

1 2NF

1

In the second normal form, the partial dependency are removed and placed them in separate table. It requires all non-key attributes to be fully dependent on unique id. The composite key are also checked. (Allen, 2024)

The 2NF:

Here the partial dependency is checked and removed. Since, project have composite key. We check there exist partial dependency or not.

From Project-1

User_ID, Project_ID ->

Project_ID -> Project Name, Project Start Date, Project Due Date, Project Status

User_ID ->

User_Project (User_ID*, Project_ID*)

Project (Project_ID, Project Name, Project Start Date, Project Due Date, Project Status)

According to business rule number 3, there exist many to may relationship between User and Task. So, checking and removing partial dependency.

From Task-1

Task_ID, User_ID ->

Task_ID -> Task Name, Start Date, Due Date, Status, Project_ID*

User_ID ->

User_Task (Task_ID*, User_ID*)

Task (Task_ID, Task Name, Start Date, Due Date, Status, Project_ID*)

Final 2NF

User-2 (User_ID, User Name, User Email, User Contact, Role)

User_Project-2 (User_ID*, Project_ID*)

Project-2 (Project_ID, Project Name, Project Start Date, Project Due Date, Project Status)

User_Task-2 (Task_ID*, User_ID*)

Task-2 (Task_ID, Task Name, Start Date, Due Date, Status, Project_ID*)

1

3NF

In the third normal form, the transitive dependency and 1 to 1 relation are checked and removed. (Allen, 2024)

The 3NF:

Checking the transitive dependency and checking business rule to find 1 to 1 relationship between entity. Since there exist no any 1 to 1 relationship and transitive dependency then final 2NF is the final 3NF.

Final 3NF

User-3 (User_ID, User Name, User Email, User Contact, Role)

User_Project-3 (User_ID*, Project_ID*)

Project-3 (Project_ID, Project Name, Project Start Date, Project Due Date, Project Status)

User_Task-3 (Task_ID*, User_ID*)

Task-3 (Task_ID, Task Name, Start Date, Due Date, Status, Project_ID*)

## Final ERD

Figure 10: Final ERD

In the above diagram, the final erd of after normalization and the business rule points are aligned properly in order to reduce the data inconsistencies, and redundancy. After solving the steps of normalization, the bridge entity User_Task and User_Project are obtained as separate entities, similarly according to the scenario and requirements the Milestone and Sub_Task entities are included in the final erd which will be implemented properly.

## Database Implementation

### Connect System

Figure 11: System connect

### Creating Username

Figure 12: Creating Username

Grant Permission

Figure 13: Permission to Username

Grant required privileged

Figure 14: Privileged to Username

Connect username with sql developer

Figure 15: Connect in Sql developer

Generate script from data modeler

```
-- Generated by Oracle SQL Developer Data Modeler 24.3.0.240.1210
--   at:        2025-01-03 09:08:18 NPT
--   site:      Oracle Database 11g
--   type:      Oracle Database 11g



-- predefined type, no DDL - MDSYS.SDO_GEOMETRY
-- predefined type, no DDL - XMLTYPE

CREATE TABLE milestone (
    milestone_id      NUMBER NOT NULL,
    "Milestone Name" VARCHAR2(50),
    "Due Date"       DATE,
    project_id       NUMBER NOT NULL,
    user_id          NUMBER NOT NULL
);

ALTER TABLE milestone ADD CONSTRAINT milestone_pk PRIMARY KEY (
milestone_id );

CREATE TABLE project (
    project_id          NUMBER NOT NULL,
    "Project Name"       VARCHAR2(50),
    "Project Start Date" DATE,
    "Project Due Date"   DATE,
    "Project Status"     VARCHAR2(50)
);

ALTER TABLE project ADD CONSTRAINT project_pk PRIMARY KEY ( project_id );
```

```
CREATE TABLE sub_task (
   stask_id     NUMBER NOT NULL,
   "STask Name" VARCHAR2(50),
   task_id      NUMBER NOT NULL
);

ALTER TABLE sub_task ADD CONSTRAINT sub_task_pk PRIMARY KEY ( stask_id
);

CREATE TABLE task (
   task_id      NUMBER NOT NULL,
   "Task Name"  VARCHAR2(50),
   "Start Date" DATE,
   "Due Date"   DATE,
   status       VARCHAR2(30),
   project_id   NUMBER NOT NULL
);

ALTER TABLE task ADD CONSTRAINT task_pk PRIMARY KEY ( task_id );

CREATE TABLE user_project (
   user_id    NUMBER NOT NULL,
   project_id NUMBER NOT NULL
);

ALTER TABLE user_project ADD CONSTRAINT "User Project_PK" PRIMARY KEY (
user_id,
                                                   project_id );

CREATE TABLE user_task (
    task_id NUMBER NOT NULL,
    user_id NUMBER NOT NULL
);

ALTER TABLE user_task ADD CONSTRAINT "User Task_PK" PRIMARY KEY (
task_id,
                                                user_id );

CREATE TABLE users (
    user_id        NUMBER NOT NULL,
    "Use Name"     VARCHAR2(20),
    "User Email"   VARCHAR2(30),
    "User Contact" NUMBER,
    role           VARCHAR2(30)
);
```

```
ALTER TABLE users ADD CONSTRAINT user_pk PRIMARY KEY ( user_id );


ALTER TABLE users ADD CONSTRAINT user__un UNIQUE ( "User Email",

                                                  "User Contact" );


ALTER TABLE milestone

   ADD CONSTRAINT milestone_project_fk FOREIGN KEY ( project_id )

      REFERENCES project ( project_id );


ALTER TABLE milestone

   ADD CONSTRAINT milestone_users_fk FOREIGN KEY ( user_id )

      REFERENCES users ( user_id );


ALTER TABLE sub_task

   ADD CONSTRAINT sub_task_task_fk FOREIGN KEY ( task_id )

      REFERENCES task ( task_id );


ALTER TABLE task

   ADD CONSTRAINT task_project_fk FOREIGN KEY ( project_id )

      REFERENCES project ( project_id );


ALTER TABLE user_project

   ADD CONSTRAINT "User Project_Project_FK" FOREIGN KEY ( project_id )
```

```
        REFERENCES project ( project_id );

ALTER TABLE user_project
    ADD CONSTRAINT "User Project_User_FK" FOREIGN KEY ( user_id )
        REFERENCES users ( user_id );

ALTER TABLE user_task
    ADD CONSTRAINT "User Task_Task_FK" FOREIGN KEY ( task_id )
        REFERENCES task ( task_id );

ALTER TABLE user_task
    ADD CONSTRAINT "User Task_User_FK" FOREIGN KEY ( user_id )
        REFERENCES users ( user_id );



-- Oracle SQL Developer Data Modeler Summary Report:
--
-- CREATE TABLE                             7
-- CREATE INDEX                             0
-- ALTER TABLE                             16
-- CREATE VIEW                              0
-- ALTER VIEW                               0
-- CREATE PACKAGE                  0
-- CREATE PACKAGE BODY             0
-- CREATE PROCEDURE                0
-- CREATE FUNCTION                 0
-- CREATE TRIGGER                  0
-- ALTER TRIGGER                   0
-- CREATE COLLECTION TYPE          0
-- CREATE STRUCTURED TYPE          0
-- CREATE STRUCTURED TYPE BODY     0
-- CREATE CLUSTER                  0
-- CREATE CONTEXT                  0
-- CREATE DATABASE                 0
-- CREATE DIMENSION                0
-- CREATE DIRECTORY                0
-- CREATE DISK GROUP               0
-- CREATE ROLE                     0
-- CREATE ROLLBACK SEGMENT         0
-- CREATE SEQUENCE                 0
-- CREATE MATERIALIZED VIEW        0
-- CREATE MATERIALIZED VIEW LOG    0
-- CREATE SYNONYM                  0
-- CREATE TABLESPACE               0
-- CREATE USER                     0
```

```
--
-- DROP TABLESPACE             0
-- DROP DATABASE              0
--
-- REDACTION POLICY            0
--
-- ORDS DROP SCHEMA            0
-- ORDS ENABLE SCHEMA           0
-- ORDS ENABLE OBJECT           0
--
-- ERRORS                 0
-- WARNINGS                0
```

Create table from script

Figure 16: Creating tables from script in sql developer

Table Users

Figure 17: Users Table

Table Project

Figure 18: Project Table

Table User_Project

Figure 19: User_Project Table

Table Task

Figure 20: Task Table

Table User_Task

Figure 21: User_Task Table

Table Sub_Task

Figure 22: Sub_Task Table

Table Milestone

Figure 23: Milestone Table

Insert into Users.

```
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (1, 'Sujan Bajgain', 'sujanbajgain123@gmail.com', 9812345678, 'Developer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (2, 'Alisha Rijal', 'alisharijal123@gmail.com', 9854321076, 'Designer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (3, 'Sital Bajgain', 'sitalbajgain123@gmail.com', 9824567012, 'Project
Manager');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (4, 'Pujan Subedi', 'pujansubedi123@gmail.com', 9800045678, 'Developer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (5, 'Shreya Koirala', 'shreyakoirala123@gmail.com', 9812005710, 'Designer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (6, 'Kamal Thapa', 'kamalthapa123@gmail.com', 9876543210, 'Project
Manager');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (7, 'Utshab Thapa', 'utshabthapa123@gmail.com', 9812233445, 'Developer');
```

```
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (8, 'Subarna Guragain', 'subarnaguragain123@gmail.com', 9800556777,
'Designer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (9, 'Sushan Pokhrel', 'sushanpokhrel123@gmail.com', 9854432111, 'Project
Manager');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (10, 'Santosh Bajgain', 'santoshbajgain123@gmail.com', 9832112233,
'Developer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (11, 'Isha Bajgain', 'ishabajgain123@gmail.com', 9822112345, 'Designer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (12, 'Kritgyeta Rijal', 'kritgyetarial123@gmail.com', 9812348790, 'Project
Manager');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (13, 'Sarita Ojha', 'saritaojha123@gmail.com', 9800778888, 'Developer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (14, 'Amit Subedi', 'amitsubedi123@gmail.com', 9823344556, 'Designer');
INSERT INTO Users (user_id, "Use Name", "User Email", "User Contact", role)
VALUES (15, 'Keshab Bajgain', 'keshabbajgain123@gmail.com', 9800011222, 'Project
Manager');
```

Figure 24: Populate Users table

Insert into Project

```
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (1, 'Banking App', TO_DATE('2024-04-01', 'YYYY-MM-DD'), TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (2, 'Futshal Management', TO_DATE('2024-01-01', 'YYYY-MM-DD'), TO_DATE('2024-10-15', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (3, 'Tender Management', TO_DATE('2024-02-10', 'YYYY-MM-DD'), TO_DATE('2024-09-20', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (4, 'Auction Management', TO_DATE('2024-01-15', 'YYYY-MM-DD'), TO_DATE('2024-11-05', 'YYYY-MM-DD'), 'Completed');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (5, 'AI', TO_DATE('2024-06-01', 'YYYY-MM-DD'), TO_DATE('2024-12-25', 'YYYY-MM-DD'), 'Active');
```

```
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (6, 'Tesla', TO_DATE('2024-07-10', 'YYYY-MM-DD'), TO_DATE('2024-08-30', 'YYYY-MM-DD'), 'Inactive');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (7, 'Blood Bank', TO_DATE('2024-08-01', 'YYYY-MM-DD'), TO_DATE('2024-12-15', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (8, 'Content Management', TO_DATE('2024-08-05', 'YYYY-MM-DD'), TO_DATE('2024-11-20', 'YYYY-MM-DD'), 'Completed');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (9, 'IoT', TO_DATE('2024-09-10', 'YYYY-MM-DD'), TO_DATE('2024-10-30', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (10, 'Tourist Guide', TO_DATE('2024-10-01', 'YYYY-MM-DD'), TO_DATE('2024-11-30', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
```

```
VALUES (11, 'Hotel Booking', TO_DATE('2024-11-10', 'YYYY-MM-DD'), TO_DATE('2025-03-28', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (12, 'Ride share', TO_DATE('2024-12-01', 'YYYY-MM-DD'), TO_DATE('2025-06-20', 'YYYY-MM-DD'), 'Inactive');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (13, 'E-Commerce', TO_DATE('2024-01-10', 'YYYY-MM-DD'), TO_DATE('2024-04-30', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (14, 'Samsung', TO_DATE('2024-03-15', 'YYYY-MM-DD'), TO_DATE('2024-05-01', 'YYYY-MM-DD'), 'Active');
INSERT INTO project (project_id, "Project Name", "Project Start Date", "Project Due Date", "Project Status")
VALUES (15, 'Trinetra', TO_DATE('2024-05-20', 'YYYY-MM-DD'), TO_DATE('2024-10-15', 'YYYY-MM-DD'), 'Completed');
```

Figure 25: Populate Project table

Insert into Task

```
INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (1, 'Task 1', TO_DATE('2024-01-01', 'YYYY-MM-DD'), TO_DATE('2024-01-11', 'YYYY-MM-DD'), 'In Progress', 1);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (2, 'Task 2', TO_DATE('2024-02-05', 'YYYY-MM-DD'), TO_DATE('2024-01-19', 'YYYY-MM-DD'), 'Not Started', 2);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (3, 'Task 3', TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-02-20', 'YYYY-MM-DD'), 'Completed', 3);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (4, 'Task 4', TO_DATE('2024-04-10', 'YYYY-MM-DD'), TO_DATE('2024-02-20', 'YYYY-MM-DD'), 'In Progress', 4);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
```

```
VALUES (5, 'Task 5', TO_DATE('2024-05-01', 'YYYY-MM-DD'), TO_DATE('2024-03-28', 'YYYY-MM-DD'), 'Not Started', 5);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (6, 'Task 6', TO_DATE('2024-06-05', 'YYYY-MM-DD'), TO_DATE('2024-03-18', 'YYYY-MM-DD'), 'Completed', 6);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (7, 'Task 7', TO_DATE('2024-06-01', 'YYYY-MM-DD'), TO_DATE('2024-04-19', 'YYYY-MM-DD'), 'In Progress', 7);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (8, 'Task 8', TO_DATE('2024-08-05', 'YYYY-MM-DD'), TO_DATE('2024-04-20', 'YYYY-MM-DD'), 'Not Started', 8);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (9, 'Task 9', TO_DATE('2024-09-01', 'YYYY-MM-DD'), TO_DATE('2024-05-21', 'YYYY-MM-DD'), 'Completed', 9);
```

```
INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (10, 'Task 10', TO_DATE('2024-01-05', 'YYYY-MM-DD'), TO_DATE('2024-
05-22', 'YYYY-MM-DD'), 'In Progress', 10);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (11, 'Task 11', TO_DATE('2024-11-01', 'YYYY-MM-DD'), TO_DATE('2024-
06-23', 'YYYY-MM-DD'), 'Not Started', 11);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (12, 'Task 12', TO_DATE('2024-12-10', 'YYYY-MM-DD'), TO_DATE('2024-
06-24', 'YYYY-MM-DD'), 'Completed', 12);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (13, 'Task 13', TO_DATE('2024-11-01', 'YYYY-MM-DD'), TO_DATE('2024-
07-25', 'YYYY-MM-DD'), 'In Progress', 13);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
```

```
VALUES (14, 'Task 14', TO_DATE('2024-10-05', 'YYYY-MM-DD'), TO_DATE('2024-07-26', 'YYYY-MM-DD'), 'Not Started', 14);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (15, 'Task 15', TO_DATE('2024-09-01', 'YYYY-MM-DD'), TO_DATE('2024-08-27', 'YYYY-MM-DD'), 'Completed', 15);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (16, 'Task 16', TO_DATE('2024-08-05', 'YYYY-MM-DD'), TO_DATE('2024-08-28', 'YYYY-MM-DD'), 'In Progress', 1);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (17, 'Task 17', TO_DATE('2024-07-01', 'YYYY-MM-DD'), TO_DATE('2024-09-29', 'YYYY-MM-DD'), 'Not Started', 2);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (18, 'Task 18', TO_DATE('2024-06-10', 'YYYY-MM-DD'), TO_DATE('2024-09-30', 'YYYY-MM-DD'), 'Completed', 3);
```

```
INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (19, 'Task 19', TO_DATE('2024-05-01', 'YYYY-MM-DD'), TO_DATE('2024-
10-16', 'YYYY-MM-DD'), 'In Progress', 4);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (20, 'Task 20', TO_DATE('2024-04-05', 'YYYY-MM-DD'), TO_DATE('2024-
10-12', 'YYYY-MM-DD'), 'Not Started', 5);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (21, 'Task 21', TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-
11-13', 'YYYY-MM-DD'), 'Completed', 6);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (22, 'Task 22', TO_DATE('2024-02-05', 'YYYY-MM-DD'), TO_DATE('2024-
11-24', 'YYYY-MM-DD'), 'In Progress', 7);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
```

```
VALUES (23, 'Task 23', TO_DATE('2024-01-01', 'YYYY-MM-DD'), TO_DATE('2024-
12-23', 'YYYY-MM-DD'), 'Not Started', 8);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (24, 'Task 24', TO_DATE('2024-05-05', 'YYYY-MM-DD'), TO_DATE('2024-
12-15', 'YYYY-MM-DD'), 'Completed', 9);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (25, 'Task 25', TO_DATE('2024-12-10', 'YYYY-MM-DD'), TO_DATE('2024-
12-20', 'YYYY-MM-DD'), 'In Progress', 10);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (26, 'Task 26', TO_DATE('2024-12-15', 'YYYY-MM-DD'), TO_DATE('2024-
12-25', 'YYYY-MM-DD'), 'Not Started', 11);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS",
PROJECT_ID)
VALUES (27, 'Task 27', TO_DATE('2024-12-20', 'YYYY-MM-DD'), TO_DATE('2024-
12-30', 'YYYY-MM-DD'), 'Completed', 12);
```

```
INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (28, 'Task 28', TO_DATE('2024-12-25', 'YYYY-MM-DD'), TO_DATE('2025-01-05', 'YYYY-MM-DD'), 'In Progress', 13);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (29, 'Task 29', TO_DATE('2024-12-28', 'YYYY-MM-DD'), TO_DATE('2025-01-10', 'YYYY-MM-DD'), 'Not Started', 14);

INSERT INTO Task (TASK_ID, "Task Name", "Start Date", "Due Date", "STATUS", PROJECT_ID)
VALUES (30, 'Task 30', TO_DATE('2024-12-30', 'YYYY-MM-DD'), TO_DATE('2025-01-15', 'YYYY-MM-DD'), 'Completed', 15);
```

Figure 26: Populate Task table

Insert into User_Task

```
INSERT INTO User_Task VALUES (1, 1);
INSERT INTO User_Task VALUES (2, 2);
INSERT INTO User_Task VALUES (3, 3);
INSERT INTO User_Task VALUES (4, 4);
INSERT INTO User_Task VALUES (5, 5);
INSERT INTO User_Task VALUES (6, 1);
INSERT INTO User_Task VALUES (7, 2);
INSERT INTO User_Task VALUES (8, 3);
INSERT INTO User_Task VALUES (9, 4);
INSERT INTO User_Task VALUES (10, 5);
INSERT INTO User_Task VALUES (11, 1);
INSERT INTO User_Task VALUES (12, 2);
INSERT INTO User_Task VALUES (13, 3);
INSERT INTO User_Task VALUES (14, 4);
INSERT INTO User_Task VALUES (15, 5);
INSERT INTO User_Task VALUES (16, 1);
INSERT INTO User_Task VALUES (17, 2);
INSERT INTO User_Task VALUES (18, 3);
INSERT INTO User_Task VALUES (19, 4);
INSERT INTO User_Task VALUES (20, 5);
INSERT INTO User_Task VALUES (21, 1);
INSERT INTO User_Task VALUES (22, 2);
INSERT INTO User_Task VALUES (23, 3);
INSERT INTO User_Task VALUES (24, 4);
INSERT INTO User_Task VALUES (25, 5);
INSERT INTO User_Task VALUES (26, 1);
INSERT INTO User_Task VALUES (27, 2);
INSERT INTO User_Task VALUES (28, 3);

INSERT INTO User_Task VALUES (29, 4);

INSERT INTO User_Task VALUES (30, 5);
```

Figure 27: Populate User_Task

Insert into User_Project

```
INSERT INTO User_Project VALUES (1, 1);

INSERT INTO User_Project VALUES (2, 2);

INSERT INTO User_Project VALUES (3, 3);

INSERT INTO User_Project VALUES (4, 4);

INSERT INTO User_Project VALUES (5, 5);

INSERT INTO User_Project VALUES (6, 2);

INSERT INTO User_Project VALUES (7, 3);

INSERT INTO User_Project VALUES (8, 4);

INSERT INTO User_Project VALUES (9, 5);

INSERT INTO User_Project VALUES (10, 1);

INSERT INTO User_Project VALUES (11, 3);

INSERT INTO User_Project VALUES (12, 4);

INSERT INTO User_Project VALUES (13, 5);

INSERT INTO User_Project VALUES (14, 1);
INSERT INTO User_Project VALUES (15, 2);

INSERT INTO User_Project VALUES (1, 4);

INSERT INTO User_Project VALUES (2, 5);

INSERT INTO User_Project VALUES (3, 1);

INSERT INTO User_Project VALUES (4, 2);

INSERT INTO User_Project VALUES (5, 3);

INSERT INTO User_Project VALUES (6, 5);

INSERT INTO User_Project VALUES (7, 1);

INSERT INTO User_Project VALUES (8, 2);

INSERT INTO User_Project VALUES (9, 3);

INSERT INTO User_Project VALUES (10, 4);

INSERT INTO User_Project VALUES (11, 2);

INSERT INTO User_Project VALUES (12, 3);

INSERT INTO User_Project VALUES (13, 4);

INSERT INTO User_Project VALUES (14, 5);

INSERT INTO User_Project VALUES (15, 1);
```

Figure 28: Populate User_Project table

32 Insert into Sub_Task

17 INSERT INTO Sub_Task  VALUES (1, 'Sub Task 1', 1);

```
INSERT INTO Sub_Task  VALUES (2, 'Sub Task 2', 1);

INSERT INTO Sub_Task  VALUES (3, 'Sub Task 3', 2);

INSERT INTO Sub_Task  VALUES (4, 'Sub Task 4', 2);

INSERT INTO Sub_Task  VALUES (5, 'Sub Task 5', 3);

INSERT INTO Sub_Task  VALUES (6, 'Sub Task 6', 3);

INSERT INTO Sub_Task  VALUES (7, 'Sub Task 7', 4);

INSERT INTO Sub_Task  VALUES (8, 'Sub Task 8', 4);

INSERT INTO Sub_Task  VALUES (9, 'Sub Task 9', 5);

INSERT INTO Sub_Task  VALUES (10, 'Sub Task 10', 5);

INSERT INTO Sub_Task  VALUES (11, 'Sub Task 11', 6);

INSERT INTO Sub_Task  VALUES (12, 'Sub Task 12', 6);

INSERT INTO Sub_Task  VALUES (13, 'Sub Task 13', 7);

INSERT INTO Sub_Task  VALUES (14, 'Sub Task 14', 7);

INSERT INTO Sub_Task  VALUES (15, 'Sub Task 15', 8);

INSERT INTO Sub_Task  VALUES (16, 'Sub Task 16', 8);

INSERT INTO Sub_Task  VALUES (17, 'Sub Task 17', 9);

INSERT INTO Sub_Task  VALUES (18, 'Sub Task 18', 9);

INSERT INTO Sub_Task  VALUES (19, 'Sub Task 19', 10);

INSERT INTO Sub_Task  VALUES (20, 'Sub Task 20', 10);

INSERT INTO Sub_Task  VALUES (21, 'Sub Task 21', 11);

INSERT INTO Sub_Task  VALUES (22, 'Sub Task 22', 11);

INSERT INTO Sub_Task  VALUES (23, 'Sub Task 23', 12);

INSERT INTO Sub_Task  VALUES (24, 'Sub Task 24', 12);
INSERT INTO Sub_Task  VALUES (25, 'Sub Task 25', 13);

INSERT INTO Sub_Task  VALUES (26, 'Sub Task 26', 13);

INSERT INTO Sub_Task  VALUES (27, 'Sub Task 27', 14);

INSERT INTO Sub_Task  VALUES (28, 'Sub Task 28', 14);

INSERT INTO Sub_Task  VALUES (29, 'Sub Task 29', 15);

INSERT INTO Sub_Task  VALUES (30, 'Sub Task 30', 15);
```

Figure 29: Populate Sub_Task table

Insert into Milestone

```
INSERT INTO milestone  VALUES (1, 'Initial Planning', TO_DATE('2024-01-01', 'YYYY-MM-DD'), 1, 1);

INSERT INTO milestone  VALUES (2, 'UI Design', TO_DATE('2024-02-10', 'YYYY-MM-DD'), 2, 2);

INSERT INTO milestone  VALUES (3, 'SRS', TO_DATE('2024-04-01', 'YYYY-MM-DD'), 3, 3);

INSERT INTO milestone  VALUES (4, 'Frontend', TO_DATE('2024-02-05', 'YYYY-MM-DD'), 4, 4);
INSERT INTO milestone  VALUES (5, 'Database Design', TO_DATE('2024-02-01', 'YYYY-MM-DD'), 5, 5);

INSERT INTO milestone  VALUES (6, 'Backend', TO_DATE('2024-04-16', 'YYYY-MM-DD'), 6, 6);

INSERT INTO milestone  VALUES (7, 'API Testing', TO_DATE('2024-05-05', 'YYYY-MM-DD'), 7, 7);

INSERT INTO milestone  VALUES (8, 'Integration', TO_DATE('2024-10-30', 'YYYY-MM-DD'), 8, 8);

INSERT INTO milestone  VALUES (9, 'Sprint Review', TO_DATE('2024-03-05', 'YYYY-MM-DD'), 9, 9);

INSERT INTO milestone  VALUES (10, 'System Testing', TO_DATE('2024-06-15', 'YYYY-MM-DD'), 10, 10);

INSERT INTO milestone  VALUES (11, 'User Testing', TO_DATE('2024-09-29', 'YYYY-MM-DD'), 11, 11);
INSERT INTO milestone  VALUES (12, 'Implementation', TO_DATE('2024-11-29', 'YYYY-MM-DD'), 12, 12);

INSERT INTO milestone  VALUES (13, 'System Design', TO_DATE('2024-12-15', 'YYYY-MM-DD'), 13, 13);

INSERT INTO milestone  VALUES (14, 'Feasibility Study', TO_DATE('2024-04-01', 'YYYY-MM-DD'), 14, 13);

INSERT INTO milestone  VALUES (15, 'Final Presentation', TO_DATE('2024-12-01', 'YYYY-MM-DD'), 15, 15);
```

Figure 30: Populate Milestone table

## Web-based Database Application

### Home Page

Figure 31: Home Page

## Basic Webforms

### User Details

Figure 32: User Details page

### Project Details

Figure 33: Project Details page

Task Details

Figure 34: Task Details page

Sub Task Details

Figure 35: Sub Task Details page

Milestone Details

Figure 36: Milestone Details Page

Complex webforms

User Project

Figure 37: User Project Page

Project Milestone

Figure 38: Project Milestone Page

Top Performers

Figure 39: Top Performers page

Testing

Test 1: Test to add the new user.

Table 2: Test to add the new user.

Test

Add the new user

Action

Add new user:

Username = Ram Adhikari

Email = ramad123@gmail.com

Contact= 9812342423

Role = Developer

Expected Result

The new user should be added with the required username, email, contact, and role.

Actual Result

The new user is added with the input username, email, contact, and role.

Conclusion

Test Successful

Figure 40: Adding User with the required input.

Figure 41: User Added successfully.

Figure 42: New User in Database

Test 2: Test to update the user.

Table 3:Test to update the user.

Test

Update the user

Action

Current user details:

Username = Ram Adhikari

Email = ramad123@gmail.com

Contact= 9812342423

Role = Developer

Updated user details:

Username = Ram Adhikari

Email = ramad123@gmail.com

Contact= 9812342423

Role = Designer

36

Expected Result

The existing user should be updated.

Actual Result

The user is updated with the input credentials.

Conclusion

Test Successful

Figure 43: Current User Details

Figure 44: Updating the User role

Figure 45: User role updated successfully

Figure 46: Updated user in database

Test 3: Test to delete the user.

Table 4:Test to delete the user.

Test

Delete the user

Action

Current user details:

Username = Ram Adhikari

Email = ramad123@gmail.com

Contact= 9812342423

Role = Designer

Delete user details of current user with user id 21.

Expected Result

The selected user should be able to delete.

Actual Result

The user is deleted.

Conclusion

Test Successful

Figure 47: Current User details in the user list with user id 21.

Figure 48: Deleting the user

Figure 49: User with user id 21 deleted successfully.

Figure 50: Database after deleting the user

Test 4: Test to add the new project.

Table 5: Test to add the new project

Test

Add the new project

Action

Add new project:

Project Name = FYP

Start Date = January 1, 2025

Due Date= April 30, 2025

Status = Active

Expected Result

The new project should be added with the input credentials.

Actual Result

The new project is added with the input credentials.

Conclusion

Test Successful

Figure 51: Adding new project

Figure 52: Project Added Successfully

Figure 53: Added project in the database

Test 5: Test to update the project.

Table 6: Test to update the project

Test

Update the  project

Action

Current project:

Project Name = FYP

Start Date = January 1, 2025

Due Date= April 30, 2025

Status = Active

Updated project:

Project Name = FYP

Start Date = January 1, 2025

Due Date= April 30, 2025

Status = Completed

Expected Result

The existing project should be updated with the input credentials.

Actual Result

The selected project is updated with the input credentials.

Conclusion

Test Successful

Figure 54: Selected Project Details

Figure 55: Updating the project details

Figure 56: Project updated successfully

Figure 57: Updated Project details in the database

Test 6: Test to delete the project.

Table 7: Test to delete the project

Test

Delete the project

Action

Current project:

Project Name = FYP

Start Date = January 1, 2025

Due Date = April 30, 2025

Status = Active

Delete project with project ID 19.

Expected Result

The selected project should be deleted.

Actual Result

The selected project is deleted.

Conclusion

Test Successful

Figure 58: Current Project details with project id 19

Figure 59: Deleting the project

Figure 60: Selected project delete successfully.

Figure 61: Database after deleting the project with project id 19

Test 7: Test to add the new task.

Table 8: Test to add the new task

Test

Add the new task

Action

Add new task:

Task Name = New Task

Start Date = March 5, 2025

Due Date= March 7, 2025

Status = Not Started

Project = Banking App

Expected Result

The new task should be added with the input credentials.

Actual Result

The new task is added with the input credentials.

Conclusion

Test Successful

Figure 62: Adding new task

Figure 63: Task added successfully

Figure 64: Added task in the database

Test 8: Test to update the task.

Table 9: Test to update the task

Test

Update the task

Action

Current task:

Task Name = New Task

Start Date = March 5, 2025

Due Date = March 7, 2025

Status = Not Started

Project = Banking App

Updated task:

Task Name = New Task

Start Date = March 5, 2025

Due Date = March 7, 2025

Status = In Progress

Project = Banking App

Expected Result

The selected task should be updated with the input credentials.

Actual Result

The selected task is updated with the input credentials.

Conclusion

Test Successful

Figure 65: Selected Task

Figure 66: Updating the task

Figure 67: Task Updated successfully

Figure 68: Updated task in the database

Test 9: Test to delete the task.

Table 10: Test to delete the task

Test

Delete the task

Action

Current task:

Task Name = New Task

Start Date = March 5, 2025

Due Date = March 7, 2025

Status = In Progress

Project = Banking App

Delete task with task id 36.

Expected Result

The selected task should be deleted.

Actual Result

The selected task is deleted.

Conclusion

Test Successful

Figure 69: Before deleting task with task id 36

Figure 70: Deleting the selected task

Figure 71: Task delete successful

Figure 72: Database after deleting task with id 36

Test 10: Test to add the new sub-task.

Table 11: Test to add the new sub-task

Test

Add the new sub-task

Action

Add new sub-task:

Sub Task Name = New Sub Task

Task = Task 1

Expected Result

The new sub-task should be added with the input credentials.

Actual Result

The new sub-task is added with the input credentials.

Conclusion

Test Successful

Figure 73: Adding new sub-task

Figure 74: New sub-task added successfully

Figure 75: Added new sub task in database with id 32

Test 11: Test to update the sub task.

Table 12: Test to update the sub task

Test

Update the sub-task

Action

Current sub-task:

Sub Task Name = New Sub Task

Task = Task 1

Updated sub-task:

Sub Task Name = Modify Sub Task

Task = Task 1

Expected Result

The selected sub-task should be updated with the input credentials.

Actual Result

The selected sub-task is updated with the input credentials.

Conclusion

Test Successful

Figure 76: Selected sub-task

Figure 77: Updating sub-task

Figure 78: Sub-task updated successfully

Figure 79: Database after updating the sub-task

Test 12: Test to delete the sub task.

Table 13: Test to delete the sub task

Test

Delete the sub-task

Action

Current sub-task:

Sub Task Name = New Sub Task

Task = Task 1

Delete sub-task with id 32

Expected Result

The selected sub-task should be deleted.

Actual Result

The selected sub-task is deleted.

Conclusion

Test Successful

Figure 80: Before deleting the sub-task with id 32

Figure 81: Deleting sub-task

Figure 82: Sub-task deleted successfully

Figure 83: Database after deleting the sub-task with id 32

Test 13: Test to add the new milestone.

Table 14: Test to add the new milestone

Test

Add the new milestone

Action

Add new milestone:

Milestone Name = milestone 3

Due Date = March 6, 2025

Project = Project Beta

Assigned User = Sujan Bajgain

14 Expected Result

The new milestone should be added with the input credentials.

Actual Result

The new milestone is added with the input credentials.

Conclusion

Test Successful

Figure 84: Adding new milestone

Figure 85: Milestone added successfully

Figure 86: Database after adding the new milestone with id 16

Test 14: Test to update the milestone.

Table 15: Test to update the milestone

Test

Update the milestone

Action

Current milestone:

Milestone Name = milestone 3

Due Date = March 6, 2025

Project = Project Beta

Assigned User = Sujan Bajgain

Updated milestone:

Milestone Name = milestone 3

Due Date = March 6, 2025

Project = Project Beta

Assigned User = Sushan Pokhrel

Expected Result

The milestone should be updated with the input credentials.

Actual Result

The milestone is updated with the input credentials.

Conclusion

Test Successful

Figure 87: Before updating milestone

Figure 88: Updating milestone

Figure 89: Milestone updated successfully

Figure 90: database after updating milestone

Test 15: Test to delete the milestone.

Table 16: Test to delete the milestone

Test

Update the milestone

Action

Current milestone:

Milestone Name = milestone 3

Due Date = March 6, 2025

Project = Project Beta

Assigned User = Sushan Pokhrel

Delete milestone with id 16

13

Expected Result

The selected milestone should be deleted.

Actual Result

The selected milestone is deleted.

Conclusion

Test Successful

Figure 91: Before deleting the milestone with  id 16

Figure 92: Deleting the milestone

Figure 93: Milestone deleted successfully

Figure 94: Database after deleting the milestone with id 16

28 Test 16: Test to view the user details and project details for the selected user.

28 Table 17: Test to view the user details and project details

Test

View user details and project details of selected users

Action

Selected user:

Select User: Sujan Bajgain

Expected Result

The selected user working in the project should be able to display.

Actual Result

The selected user working in the project is display.

Conclusion

Test Successful

Figure 95: Before selecting the user

Figure 96: After selecting the user

Test 17: Test to view the project details with its milestones.

Table 18: Test to view the project details with its milestones

Test

View project details with its milestones

Action

Selected project:

Select Project: Banking App

Expected Result

The selected project with its milestone should be display.

Actual Result

The selected project with its milestone is display

Conclusion

Test Successful

Figure 97: Before selecting project

Figure 98: After selecting project and showing its milestone

Test 18: Test to view the top 3 performers of the project.

Table 19: Test to view the top 3 performers of the project

Test

View top 3 performers of the project

Action

Selected project:

Select Project: Banking App

Expected Result

The selected project with its top 3 performers should be displayed.

Actual Result

The selected project with its top 3 performers is displayed.

Conclusion

Test Successful

Figure 99: Before selecting the project

Figure 100: After selecting project and showing top 3 performers

## References

1 Allen, M., 2024. DBMS Normalization: 1NF, 2NF, 3NF Database Example. [Online] Available at: https://www.guru99.com/database-normalization.html [Accessed 25 December 2024].

Kitthu, H. A., 2024. Normalization in DBMS: A Detailed Guide. [Online] Available at: https://www.simplilearn.com/tutorials/sql-tutorial/what-is-normalization-in-sql [Accessed 25 December 2024].

Rouse, M., 2017. Entity-Relationship Diagram. [Online] Available at: https://www.techopedia.com/definition/1200/entity-relationship-diagram-erd [Accessed 25 December 2024].

Thakur, S., 2016 . E-R Diagrams in DBMS: Components, Symbols, And Notations. [Online]

Available at: https://whatisdbms.com/e-r-diagrams-in-dbms-components-symbols-and-notations/

[Accessed 25 December 2024].