22068011-Kritika Ghimire 5.docx

Islington College,Nepal

Document Details

Submission ID
trn:oid:::3618:87809234

Submission Date
Mar 25, 2025, 11:39 PM GMT+5:45

Download Date
Mar 25, 2025, 11:40 PM GMT+5:45

File Name
22068011-Kritika Ghimire 5.docx

File Size
31.9 KB

43 Pages

4,074 Words

27,811 Characters

36% Overall Similarity

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

Match Groups

- **158** Not Cited or Quoted 36%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  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

- 9% Internet sources
- 0% Publications
- 36% 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

- **158** Not Cited or Quoted 36%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  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

- 9% Internet sources
- 0% Publications
- 36% 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	University of Northampton on 2021-01-17 — 7%
2	Submitted works	Queen Mary and Westfield College on 2023-12-01 — 6%
3	Submitted works	unibuc on 2025-02-04 — 5%
4	Submitted works	islingtoncollege on 2024-12-30 — 4%
5	Submitted works	University of Portsmouth on 2023-01-30 — 3%
6	Submitted works	Oklahoma State University on 2024-05-04 — 2%
7	Submitted works	Curtin University of Technology on 2023-10-20 — 1%
8	Submitted works	University of Maryland, University College on 2022-04-04 — <1%
9	Submitted works	Leeds Beckett University on 2024-03-18 — <1%
10	Submitted works	University of Greenwich on 2016-05-12 — <1%

11 Submitted works
islingtoncollege on 2024-12-30 <1%

12 Submitted works
University of Maryland, University College on 2020-12-09 <1%

13 Submitted works
islingtoncollege on 2024-12-30 <1%

14 Submitted works
islingtoncollege on 2024-12-30 <1%

15 Submitted works
islingtoncollege on 2024-12-30 <1%

16 Submitted works
islingtoncollege on 2024-12-31 <1%

17 Submitted works
unibuc on 2025-01-11 <1%

18 Submitted works
ULACIT Universidad Latinoamericana de Ciencia y Tecnología on 2024-04-25 <1%

19 Submitted works
University of East London on 2024-12-13 <1%

20 Submitted works
islingtoncollege on 2024-12-30 <1%

21 Submitted works
University of Colorado, Denver on 2025-03-21 <1%

22 Submitted works
islingtoncollege on 2024-12-30 <1%

23 Submitted works
islingtoncollege on 2024-12-30 <1%

24 Submitted works
CSU, Los Angeles on 2023-11-14 <1%

25 Submitted works
islingtoncollege on 2024-12-30 <1%

26 Submitted works
Napier University on 2021-12-03 <1%

27 Submitted works
islingtoncollege on 2024-12-30 <1%

28 Submitted works
islingtoncollege on 2024-12-30 <1%

29 Submitted works
Webster University on 2025-03-04 <1%

30 Submitted works
islingtoncollege on 2024-12-30 <1%

31 Submitted works
Kingston University on 2022-04-11 <1%

32 Submitted works
Leeds Beckett University on 2025-03-12 <1%

33 Submitted works
QA Learning on 2024-06-25 <1%

34 Submitted works
University of Greenwich on 2017-04-25 <1%

35 Submitted works
islingtoncollege on 2024-12-30 <1%

Introduction

with C# for the web interface and Oracle SQL Developer for database management. In this particular system users can be a part of multiple project and each of the project consists of tasks and the tasks also consists multiple subtasks. Users can comment. Project needs it's resources and tasks also contains milestones. First, this project has to make initial erd of the overall project. Then normalization is done and final er-diagram is drawn. Final erd is drawn in data modeler and select, insert is done oracle sql developer. Then implementation of web-based database application is done using Asp.Net with C#. The project should include webforms which include basic webfroms and complex webforms.

This project focuses on solving many problems like difficulties of assigning and tracking tasks. Here, a user can be assigned with many tasks and there will be other users as well that is assigned with a single task. To have a proper communication and to get proper feedback there is option of comment in this project. This project will help in improve team collaboration, ensure of smooth project implementation, and maintain project-related data.

14

Aims and Objectives

AIM

The main aim of this project is to develop a user-friendly and a affective management system that will help our LS Corporation to track project progress, manage milestones, assign tasks and sub-tasks, and have proper communication through comments.

Objective

To manage users and allow enable users to be a part of multiple projects and tasks.

Allows users to add, edit, delete users, projects, tasks, milestones, subtasks, and resources.

Helps in allowing having different resources assigned to them.

To store and manage all the data using Oracle SQL Developer.

To test and validate all the features of the system, ensuring proper functionality, security, and usability.

To provide a dashboard with real time update on the progress of the task completion, milestones, and projects.

To enable effective communication among team members through the comment section.

To provide a user friendly interface making system easy to run.

Textual Analysis

.

User and comment

ER-Diagram

Figure 8 Inital ERD

3. Normalization

. Here is the normalization of the scenario given:

UNF

In an unnormalized form, data is in its raw and unstructured form. In this form, repeating groups and repeating data are included. Here, is the unnormalized form of this scenario

1NF

In this 1NF UNF is transformed into a structured table. Here, separating data into

different tables.

2NF

Should remove partial dependencies.

For the User Table:

User-2 (User_ID, User_name, User_email, User_contact)

For the Project Table:

Project_id à Project_name, Project_StartDate, Project_DueDate, Project_Status

User_ID à X

X

For The Task Table:

Task_id à Task_name, Task_StartDate, Task_DueDate, Task_Status

Here, one more new table can be formed where it resolves many to many relationship between task, project, and user.

So, here are the final tables for 2NF

Final 2NF tables:

3NF

There should be removal of transitive dependencies.

There are no indirect relation between non-key attributes.

For the Project Table:

PF or the ProjectUser Table:

This ProjectUser table is used as bridge table. This table associates each user with a

project, representing a unique combination of user and project. The ProjectUser table have no non-key values so, there is no transitive dependencies.

For Task Table:

Task_id à Task_name à X

Task_id à Task_StartDate à X

Task_id à Task_DueDate à X

Task_id à Task_Status à X

For TaskProjectUser Table:

This TaskProjectUser table is used as bridge table. This table associates each user with a project and with different tasks in the project, representing a unique combination of user project, and task. The TaskProjectUser table have no non-key values so, there is no transitive dependencies.

So, here are the final tables after 3NF

Final Tables:

User (User_ID, User_name, User_email, User_contact)

Project (Project_id, Project_name, Project_StartDate, Project_DueDate, Project_Status

ProjectUser (User_ID*, Project_id*)

4. Integration and Assumptions

d. Tasks can have multiple sub tasks.

e. Each and every project includes milestone and resources.

f. Users can comment one or multiple comments.

5. Final ERD

This is the creation of final Er-diagram.

Figure 9 Final ERD

6. Data Dictionary

User table

Table 5 Milestone table

7. Data Creation

Table Creation of All the tables:

Figure 10 Creation of table

8. Insertion of data

Here, is insertion of value of all the tables.

Figure 11 Insert user table

Figure 12 Select Users

select *From users;

Figure 13 Insert Project

Figure 14 Select Project

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date, ProjectEnd_Date)
VALUES (100, 'Student Admission System', TO_DATE('2024-12-20', 'YYYY-MM-DD'), TO_DATE('2025-04-30', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,

```
ProjectEnd_Date)
VALUES (108, 'Smart Home Service', TO_DATE('2025-04-07', 'YYYY-MM-DD'),
TO_DATE('2025-07-01', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (109, 'Supply Chain System', TO_DATE('2025-04-30', 'YYYY-MM-DD'),
TO_DATE('2025-09-30', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (110, 'MY Social App', TO_DATE('2025-05-15', 'YYYY-MM-DD'),
TO_DATE('2025-09-15', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (111, 'Thank You', TO_DATE('2025-04-20', 'YYYY-MM-DD'),
TO_DATE('2025-08-29', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (112, 'Student Management System', TO_DATE('2025-06-18', 'YYYY-MM-
DD'), TO_DATE('2025-10-11', 'YYYY-MM-DD'));
```

```
INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (113, 'Ecommerce Solutions', TO_DATE('2025-07-12', 'YYYY-MM-DD'),
TO_DATE('2025-12-12', 'YYYY-MM-DD'));

INSERT INTO Projects (Project_id, Project_name, ProjectDue_Date,
ProjectEnd_Date)
VALUES (114, 'Data Analysis Platform', TO_DATE('2025-03-10', 'YYYY-MM-DD'),
TO_DATE('2025-09-15', 'YYYY-MM-DD'));

Select * from Projects;
```

Figure 15 Insert Tasks

Figure 16 Select Tasks

```
INSERT INTO Tasks (Task_id, Task_name, Task_duedate, Task_enddate)
VALUES (1, 'task1', TO_DATE('2025-01-15', 'YYYY-MM-DD'), TO_DATE('2025-01-20',
'YYYY-MM-DD'));

INSERT INTO Tasks (Task_id, Task_name, Task_duedate, Task_enddate)
VALUES (2, 'task2', TO_DATE('2025-02-01', 'YYYY-MM-DD'), TO_DATE('2025-02-05',
```

'YYYY-MM-DD'));

INSERT INTO Tasks (Task_id, Task_name, Task_duedate, Task_enddate)

17 VALUES (3, 'task3', TO_DATE('2025-02-10', 'YYYY-MM-DD'), TO_DATE('2025-02-20',

'YYYY-MM-DD'));

INSERT INTO Tasks (Task_id, Task_name, Task_duedate, Task_enddate)

18 VALUES (4, 'task4', TO_DATE('2025-02-25', 'YYYY-MM-DD'), TO_DATE('2025-03-05',

'YYYY-MM-DD'));

Select * from tasks;

Figure 17 Insert SubTasks

Figure 18 Select subtasks

4 INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2001,

'Create ERD', 1);

4 INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2002,

'Normalize Tables', 1);

4 INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2003,

'Setup Environment', 2);

4 INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2004,

'Build UI Components', 2);

```
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2005,
'Setup API Endpoints', 3);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2006,
'Create Database Schema', 3);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2007,
'Implement Business Logic', 4);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2008,
'Connect Database to API', 4);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2009,
'Create Testing Plan', 5);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2010,
'Test UI Components', 5);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2011,
'Integrate Frontend with Backend', 6);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2012,
'Build Login Screen', 6);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2013,
'Create Payment System', 7);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2014,
'Implement User Roles', 7);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2015,
'Configure Web Server', 8);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2016,
```

'Test User Roles', 8);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2017, 'Create Data Entry Forms', 9);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2018, 'Build Invoice Generator', 9);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2019, 'Setup Email Notifications', 10);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2020, 'Create Admin Dashboard', 10);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2021, 'Test API Endpoints', 11);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2022, 'Implement Data Validation', 11);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2023, 'Build Report Generation', 12);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2024, 'Set Up Continuous Integration', 12);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2025, 'Test Report Generation', 13);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2026, 'Create User Manual', 13);

INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2027, 'Monitor Server Performance', 14);

```
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2028,
'Create API Documentation', 14);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2029,
'Deploy to Production', 15);
INSERT INTO SubTasks (SubTask_id, SubTask_name, Task_id) VALUES (2030,
'Post-Deployment Testing', 15);

select *  from SubTasks;
```

Figure 19 Insert Comment

Figure 20 Select comment

```
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1001, 'Started working on the task.', SYSDATE, 1);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1002, 'Task is in progress.', SYSDATE, 2);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1003, 'Need to review the database schema.', SYSDATE, 3);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1004, 'Frontend development underway.', SYSDATE, 4);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1005, 'Working on backend API integration.', SYSDATE, 5);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
```

```
VALUES (1006, 'Initial testing completed.', SYSDATE, 6);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1007, 'Code refactored for better performance.', SYSDATE, 7);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1008, 'Bug fixes in progress.', SYSDATE, 8);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1009, 'Reviewing the task for final approval.', SYSDATE, 9);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1010, 'Unit tests passed successfully.', SYSDATE, 10);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1011, 'Final report being prepared.', SYSDATE, 11);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1012, 'Awaiting feedback from the team.', SYSDATE, 12);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1013, 'Design phase completed.', SYSDATE, 13);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1014, 'Environment setup finished.', SYSDATE, 14);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1015, 'API documentation updated.', SYSDATE, 15);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1016, 'Checking for any broken links.', SYSDATE, 1);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1017, 'UI components look good.', SYSDATE, 2);
```

```
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1018, 'Database connection verified.', SYSDATE, 3);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1019, 'Frontend UI is responsive now.', SYSDATE, 4);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1020, 'User testing completed.', SYSDATE, 5);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1021, 'Code merged successfully.', SYSDATE, 6);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1022, 'Added email notifications.', SYSDATE, 7);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1023, 'Security patches applied.', SYSDATE, 8);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1024, 'Performance optimization done.', SYSDATE, 9);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1025, 'Added pagination functionality.', SYSDATE, 10);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1026, 'Session timeout issue resolved.', SYSDATE, 11);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1027, 'Deployed to staging environment.', SYSDATE, 12);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1028, 'Reviewed performance on mobile.', SYSDATE, 13);
INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
```

```
VALUES (1029, 'Compatibility check completed.', SYSDATE, 14);

INSERT INTO Comments (Comment_id, Comment_content, comment_date, Task_id)
VALUES (1030, 'Payment integration tested.', SYSDATE, 15);

select * from Comments;
```

Figure 21 Insert Milestone

Figure 22 Select milestone

```
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (201, 'Sprint1', TO_DATE('2024-02-15', 'YYYY-MM-DD'), 101);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (202, 'Sprint2', TO_DATE('2024-01-25', 'YYYY-MM-DD'), 102);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (203, 'Sprint3', TO_DATE('2024-03-10', 'YYYY-MM-DD'), 103);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (204, 'Sprint4', TO_DATE('2024-04-15', 'YYYY-MM-DD'), 104);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (205, 'Sprint5', TO_DATE('2024-04-25', 'YYYY-MM-DD'), 105);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (206, 'Sprint6', TO_DATE('2024-05-13', 'YYYY-MM-DD'), 101);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
```

Project_id) VALUES (207, 'Sprint7', TO_DATE('2025-05-25', 'YYYY-MM-DD'), 102);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (208, 'Sprint8', TO_DATE('2024-08-02', 'YYYY-MM-DD'), 103);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (209, 'Sprint9', TO_DATE('2024-09-03', 'YYYY-MM-DD'), 104);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (210, 'Sprint10', TO_DATE('2024-10-29', 'YYYY-MM-DD'), 105);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (211, 'Sprint11', TO_DATE('2024-11-17', 'YYYY-MM-DD'), 101);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (212, 'Sprint12', TO_DATE('2024-12-26', 'YYYY-MM-DD'), 102);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (213, 'Sprint13', TO_DATE('2025-01-24', 'YYYY-MM-DD'), 103);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (214, 'Sprint14', TO_DATE('2025-02-28', 'YYYY-MM-DD'), 104);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (215, 'Sprint15', TO_DATE('2024-03-10', 'YYYY-MM-DD'), 105);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (216, 'Sprint16', TO_DATE('2025-04-18', 'YYYY-MM-DD'), 101);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (217, 'Sprint17', TO_DATE('2025-05-11', 'YYYY-MM-DD'), 102);

INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,

Project_id) VALUES (218, 'Sprint18', TO_DATE('2025-06-15', 'YYYY-MM-DD'), 103);

```
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (219, 'Sprint19', TO_DATE('2025-07-15', 'YYYY-MM-DD'), 104);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (220, 'Sprint20', TO_DATE('2025-08-15', 'YYYY-MM-DD'), 105);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (221, 'Sprint21', TO_DATE('2025-09-15', 'YYYY-MM-DD'), 101);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (222, 'Sprint22', TO_DATE('2025-10-19', 'YYYY-MM-DD'), 102);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (223, 'Sprint23', TO_DATE('2024-11-17', 'YYYY-MM-DD'), 103);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (224, 'Sprint24', TO_DATE('2025-12-18', 'YYYY-MM-DD'), 104);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (225, 'Sprint25', TO_DATE('2025-01-05', 'YYYY-MM-DD'), 105);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (226, 'Sprint26', TO_DATE('2025-02-14', 'YYYY-MM-DD'), 101);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (227, 'Sprint27', TO_DATE('2025-03-23', 'YYYY-MM-DD'), 102);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (228, 'Sprint28', TO_DATE('2025-05-17', 'YYYY-MM-DD'), 103);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
Project_id) VALUES (229, 'Sprint29', TO_DATE('2025-05-30', 'YYYY-MM-DD'), 104);
INSERT INTO Milestones (Milestone_id, Milestone_name, Milestone_duedate,
```

```
Project_id) VALUES (230, 'Sprint30', TO_DATE('2025-06-23', 'YYYY-MM-DD'), 105);

Select * from Milestones;
```

Figure 23 Insert Resources

Figure 24 Select resources

```
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (301,
'AWS Server', 1);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (302,
'MySQL Database', 2);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (303,
'React Framework', 3);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (304,
'Postman Tool', 4);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (305,
'Django Framework', 5);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (306,
'Figma', 5);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (307,
'AWS Server', 6);
```

```
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (308,
'MySQL Database', 7);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (309,
'React Framework', 8);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (310,
'Postman Tool', 9);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (311,
'Django Framework', 10);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (312,
'Figma', 10);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (313,
'AWS Server', 11);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (314,
'MySQL Database', 12);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (315,
'React Framework', 13);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (316,
'Postman Tool', 14);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (317,
'Django Framework', 15);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (318,
'Figma', 15);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (319,
```

```
'AWS Server', 1);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (320,
'MySQL Database', 2);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (321,
'React Framework', 3);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (322,
'Postman Tool', 4);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (323,
'Django Framework', 5);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (324,
'Figma', 5);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (325,
'AWS Server', 6);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (326,
'MySQL Database', 7);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (327,
'React Framework', 8);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (328,
'Postman Tool', 9);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (329,
'Django Framework', 10);
INSERT INTO Resources (Resource_id, Resource_name, Task_id) VALUES (330,
'Figma', 10);
```

select * from Resources;

5 INSERT INTO UserProject (User_Id, Project_id) VALUES (1, 100);

INSERT INTO UserProject (User_Id, Project_id) VALUES (2, 101);

INSERT INTO UserProject (User_Id, Project_id) VALUES (3, 102);

INSERT INTO UserProject (User_Id, Project_id) VALUES (4, 103);

INSERT INTO UserProject (User_Id, Project_id) VALUES (5, 104);

INSERT INTO UserProject (User_Id, Project_id) VALUES (6, 105);

INSERT INTO UserProject (User_Id, Project_id) VALUES (7, 106);

INSERT INTO UserProject (User_Id, Project_id) VALUES (8, 107);

INSERT INTO UserProject (User_Id, Project_id) VALUES (9, 108);

INSERT INTO UserProject (User_Id, Project_id) VALUES (10, 109);

INSERT INTO UserProject (User_Id, Project_id) VALUES (11, 110);

INSERT INTO UserProject (User_Id, Project_id) VALUES (12, 111);

INSERT INTO UserProject (User_Id, Project_id) VALUES (13, 112);

INSERT INTO UserProject (User_Id, Project_id) VALUES (14, 113);

INSERT INTO UserProject (User_Id, Project_id) VALUES (15, 114);

INSERT INTO UserProject (User_Id, Project_id) VALUES (1, 101);

INSERT INTO UserProject (User_Id, Project_id) VALUES (2, 102);

INSERT INTO UserProject (User_Id, Project_id) VALUES (3, 103);

INSERT INTO UserProject (User_Id, Project_id) VALUES (4, 104);

INSERT INTO UserProject (User_Id, Project_id) VALUES (5, 105);

```
INSERT INTO UserProject (User_Id, Project_id) VALUES (6, 106);
INSERT INTO UserProject (User_Id, Project_id) VALUES (7, 107);
INSERT INTO UserProject (User_Id, Project_id) VALUES (8, 108);
INSERT INTO UserProject (User_Id, Project_id) VALUES (9, 109);
INSERT INTO UserProject (User_Id, Project_id) VALUES (10, 110);
INSERT INTO UserProject (User_Id, Project_id) VALUES (11, 111);
INSERT INTO UserProject (User_Id, Project_id) VALUES (12, 112);
INSERT INTO UserProject (User_Id, Project_id) VALUES (13, 113);
INSERT INTO UserProject (User_Id, Project_id) VALUES (14, 114);
INSERT INTO UserProject (User_Id, Project_id) VALUES (15, 100);
```

```
select * from UserProject;
```

User Manual

Dashboard

Simple forms

Users

It is the user table. User table includes all the buttons needed.

Here, we added or inserted a user.

Here, new user added.

Here, user name is edit.

Here, user name is updated.

Here, user mayur is deleted.

Project

This is the project table.

Here, new project is added.

Here, book management project is edited to book service management .

Here, book management is deleted.

Here, new task is inserted.

New task, ask17 is already added.

Here, task id 16 is updated to mine tasks.

Task id 16 is deleted.

Sub Tasks

Sub tasks mini-party is being added.

Here, mini-party is added.

A column has been added. Post-deployment testing is updated to deployment testing.

Here, sub-tasks deployment testing is delete.

Milestone

New milestone is being added.

New milestone is already added.

Here, milestone id 224 is edit and updated to a lot to do.

Here, A lot to do is delete.

Complex form

UserProject

Top Users

10. Testing

10.1. Testing of Basic form

Test-1 Testing of Insert or add of a column in all basic form.

Objective

To test the addition of new columns in the form.

Actions

Firstly, add button was clicked the required filed was filled and then clicking the insert button and new column was added.

Expected outcomes

The new column should be added.

Actual Results

New column was added in the basic.

Conclusions

New column was added successfully.

Table 6 Add or insertion

Figure 25 User insert

Figure 26 User Add

Here, new user is added.

Figure 27 Project Insert

Figure 28 Project Add

Here, new project is added.

Figure 29 Task insert

Figure 30 task add

Here, new task task17 is added.

Figure 31 subtask insert

Figure 32 subtask add

Here, new sub tasks is added.

Figure 33 Milestone insert

Figure 34 milestone add

Here, new milestone is added.

Test-2 Testing of edit of a column in all basic form.

Objective

To test the edit of columns in the form.

Actions

Firstly, click edit button the required filed was filled and then clicking the update button and now column is edited.

Expected outcomes

Column should be edited successfully.

Actual Results

Column was edited in the basic form.

Conclusions

Column was edited successfully.

Table 7 Edit of the column

Figure 35 User edit

Here, edit is successful in user table.

Figure 36 Project edit

Here, project is successfully edited.

Figure 37 Task edit

Figure 38 Sub-task edit

Sub tasks successfully edited.

Figure 39 Milestone edit

Here, milestone is edited.

Test-3 Testing of delete of a column in all basic form.

Objective

To test the delete of column in the form.

Actions

Firstly, click delete button the column which is meant to be delete is deleted.

Expected outcomes

Column should be deleted successfully.

Actual Results

Column was deleted in the basic form.

Conclusions

Column was edited successfully.

Table 8 Delete of column

Figure 40 User delete

A column with user id 16 is delete.

Figure 41 Project delete

A column with project id 116 is deleted.

Figure 42 Task delete

Task id 16 is deleted.

31

Figure 43 Sub task delete

31 Sub task with sub task id 2030 is deleted.

Figure 44 Milestone delete

Here, milestone id 224 is deleted.

10.2. Testing of Complex Form.

Test-1 Testing of Top 3 Performer.

Objective

To test the top 3 performers will show or not.

Actions

Connecting User, project, and task table and finding top 3 performers from there.

Expected outcomes

Top 3 performers should be shown.

Actual Results

The actual results meet the expected outcomes.

Conclusions

Top 3 performer was shown successfully.

Table 9 Top 3 performer.

Figure 45 Top performer

Here, top 3 performer was shown successfully.

Test-2 Testing of User Project.

Objective

To test the project related to a particular user.

Actions

Selecting users from dropdown.

Expected outcomes

A particular user should be involved in the projects.

Actual Results

Users are involved in different projects.

Conclusions

A particular user is involved in the projects.

Figure 46 UserProject

Test-3 Testing of Project Milestone.

Objective

To test the Milestone of a project.

Actions

Selecting projects from dropdown.

Expected outcomes

A particular project with it’s milestones should be displayed.

Actual Results

Project with milestone is displayed.

Conclusions

Project with milestone was displayed successfully.

Table 10 Project milestone

10.3. Error testing

Objective

To resolve the server error encountered while building the dashboard in the application.

Actions

1.Checked the error logs to identify the issue.

2. Reviewed the dashboard code for syntax errors.

3. Implemented error handling for failures.

4. rewrote the mistake code with correct code.

Expected outcomes

Dashboard loads without any server errors.

Application runs smoothly without crashes.

Actual Results

The error was fixed, and the dashboard successfully loaded with the proper code.

Conclusions

The error or issue was fixed.

Table 11 Error testing

Further Discussion

In this project, from designing the database developing the UI was done. One of the goal was to make system easy to use, efficient, and scalable so that the users could manage tasks, assign resources, and track progress without any confusion. Making sure that the data was accurate and secure were also a big priority since the project details need to be well-organized and protected. Testing was also one of the most important part there were some bugs which were able to fix with the help of thorough testing. In the end, this project taught us about learning how the technologies work together and helps in building a project which is very useful.

11.1. Tools and Technologies

The tools and technologies that helped in creating this project are:

Oracle Sql developer: This tool was used in managing database. It helped in creating,

inserting, and managing all the tables that are needed for this system. Without the proper management of database, handling project related information would have been difficult.

Data Modeler: This specific tool was used for designing the final ERD of the system. The ERD helped in understanding the relationships between different entities like users, project, tasks, resources, tasks, subtasks and so on. ERD ensured that database was well structured and managed.

Visual Studio: The tool visual studio was used for the overall development of the system. It helped by providing good environment and structure for the coding, debugging, and testing the application. This web development tool helped in building web forms, and integrate database properly.

Word: This tool was used for the documentation of the project. All the reports, planning documents, system designs, and final project reports were all prepared using the Microsoft Word. Proper Documentation is key for the proper system as it helps users to understand the system.

12. Conclusion

This particular project was designed to solve LS Corporation issue of managing multiple projects in a system. The system allows users to be part of multiple projects, manage tasks and subtasks, track milestones, and communicate easily through comments. This system helped in minimizing problems as users can easily manage projects, tasks, subtasks, monitor milestones, and communicate through the comments. The technologies used in this project played a major role in ensuring its

success.

28 By using Oracle for data storage and ASP.NET with C# for backend development, we were able to create a structured and efficient solution. The system also ensures smooth task assignments and improves project tracking, which will help the company work more efficiently. These tools ensured that the system was stable fast, and easy to operate. The project successfully meets its goal of improving project management by making it easier to track progress, assign tasks, and improve communication among team members.

This system can improve project management by making it more structured and efficient. The introduction ot project management, tasks handling, proper milestone tracking, proper resource allocation and well communication through comment has helped the LS Corporation work more easily and productively. Many people can be inspired for making this system and using it for the management of tasks, projects. This project helped in meeting all the needs and goals and enhanced the way LS Corporation handles their project.