

CSE 414 Databases Project Report

Project Management System

Abdullah Çelik

171044002

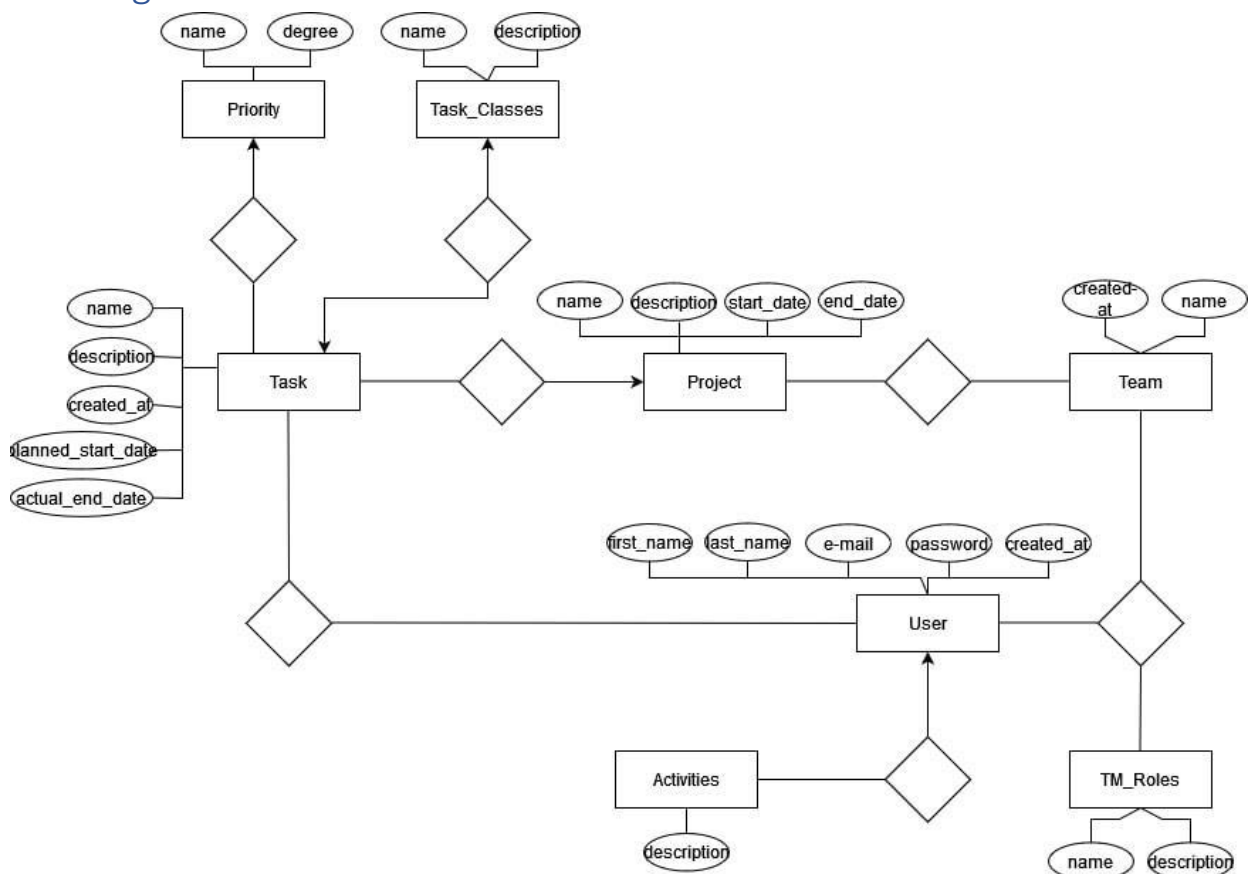
Table of Contents

User Requirements	2
E-R Diagram	2
Normalization	3
Functional Dependencies	3
1. Activies	3
2. Priority.....	3
3. TaskClasses	3
4. Project	3
5. User	3
6. TMRoles.....	3
7. Team	3
8. Task.....	3
Tables	3
Application.....	4
Join Queries	5
Triggers.....	5
View Queries.....	7

User Requirements

- Users can register to the system
- Users can login to the system
- Users can create a team
- Users can attend to a team
- Users can start a project
- Users can create new tasks for projects
- Task can have one priority
- Task can have a class
- Task can belong to a project
- Task can be assigned to users
- More than one user can work on a task
- A user may be participant for more than one team
- A team has more than one participant
- A Team member has a role for that team
- Activities are logged in a table

E-R Diagram



Normalization

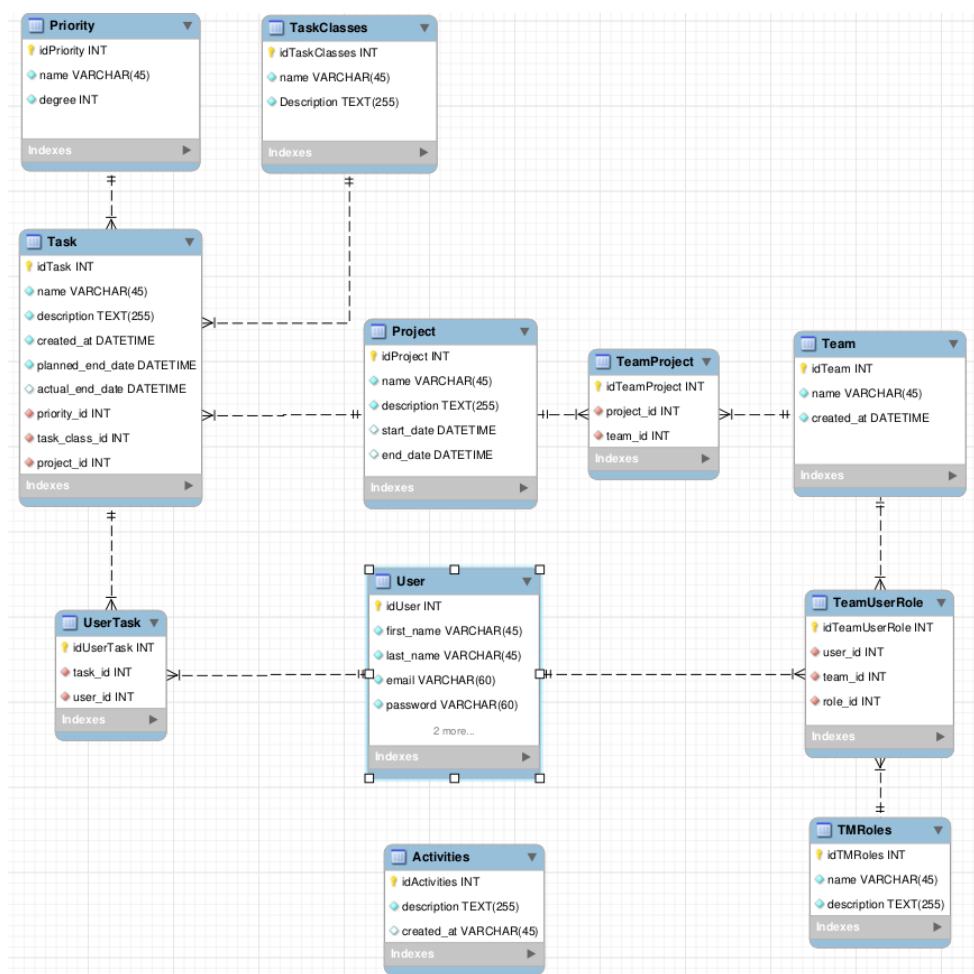
For all tables satisfied 3NF, Attribute values are atomic, each data field holds a single value, and not a composite value or multiple values. Partial dependencies among database relations are eliminated. There is no transitive dependency.

Functional Dependencies

1. Activities
idActivities ----> description, created_at
2. Priority
idPriority ----> name, degree
3. TaskClasses
idTaskClasses ----> name, Description
4. Project
idProject ----> name, description, start_date, end_date
5. User
idUser ----> first_name, last_name, email, password, created_at, company_email
6. TMRoles
idTMRoles ----> name, description
7. Team
idTeam ----> name, created_at
8. Task
idTask ----> name, description, created_at, planned_end_date, actual_end_date, priority_id, task_class_id, project_id

Tables

- Activities (idActivities, description, created_at)
- Priority (idPriority, name, degree)
- TaskClasses (idTaskClasses, name, Description)
- Project (idProject, name, description, start_date, end_date)
- User (idUser, first_name, last_name, email, password, created_at, company_email)
- TMRoles (idTMRoles, name, description)
- Team (idTeam, name, created_at)
- Task (idTask, name, description, created_at, planned_end_date, actual_end_date, priority_id, task_class_id, project_id)
- TeamProject (idTeamProject, project_id, team_id)
- UserTask (idUserTask, task_id, user_id)
- TeamUserRole (idTeamUserRole, user_id, team_id, role_id)



Application

I have used mysql and mysql workbench to create database. I have used Flask and Jinja for user interfaces.

	name	description	created_at
0	7 Pictures	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
1	How to Increase	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
2	21 Myths	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
3	Introducing Anonymous	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
4	Shape Shifter	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
5	Can Giraffes Dance	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
6	Giraffes Are	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
7	20 Tights	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
8	From Zero to Shape Shifter	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
9	YouTube Videos	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
10	10 Awesome	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	2022-06-18 01:11:19
11	POC	poc	2023-06-23 23:59:52

	first_name	last_name	created_at	company_email
0	Yurem	Barton	2022-08-17 23:59:59	YuremBarton@mycompany.com
1	Rayne	Foster	2022-08-17 23:59:59	RayneFoster@mycompany.com
2	Chace	Lee	2022-08-17 23:59:59	ChaceLee@mycompany.com
3	Kierra	Howell	2022-08-17 23:59:59	KierraHowell@mycompany.com
4	Chandler	Reeves	2022-08-17 23:59:59	ChandlerReeves@mycompany.com
5	Salvatore	Booth	2022-08-17 23:59:59	SalvatoreBooth@mycompany.com
6	Dangelo	Johns	2022-08-17 23:59:59	DangeloJohns@mycompany.com
7	Madeline	Thornton	2022-08-17 23:59:59	MadelineThornton@mycompany.com
8	Paisley	Sexton	2022-08-17 23:59:59	PaisleySexton@mycompany.com
9	Makena	Mendoza	2022-08-17 23:59:59	MakenaMendoza@mycompany.com

Join Queries

```
/* OUTER RIGHT JOIN */
```

```
SELECT * FROM Priority RIGHT OUTER JOIN Task ON Priority.idPriority = Task.priority_id;
```

```
/* OUTER LEFT JOIN */
```

```
SELECT * FROM TaskClasses LEFT OUTER JOIN Task  
ON TaskClasses.idTaskClasses = Task.priority_id;
```

```
/* FULL OUTER JOIN */
```

```
SELECT * FROM Project FULL JOIN Task;
```

Triggers

1)

```
USE `GTUProjectManagement`$$  
CREATE DEFINER = CURRENT_USER TRIGGER  
`GTUProjectManagement`.`Task_AFTER_INSERT` AFTER INSERT ON `Task` FOR EACH ROW  
BEGIN  
    INSERT INTO `GTUProjectManagement`.`Activities` (`description`) VALUES  
    (CONCAT(NEW.name, " is a new task"));  
END$$
```

2)

```
USE `GTUProjectManagement`$$  
CREATE DEFINER = CURRENT_USER TRIGGER  
`GTUProjectManagement`.`User_AFTER_INSERT` AFTER INSERT ON `User` FOR EACH ROW  
BEGIN  
    INSERT INTO `GTUProjectManagement`.`Activities` (`description`) VALUES  
    (CONCAT(NEW.first_name, " is a new user"));  
END$$
```

3)

```
USE `GTUProjectManagement`$$

CREATE DEFINER = CURRENT_USER TRIGGER
`GTUProjectManagement`.`Activities_BEFORE_INSERT` BEFORE INSERT ON `Activities`
FOR EACH ROW

BEGIN

    SET NEW.created_at = NOW();

END$$
```

4)

```
USE `GTUProjectManagement`$$

CREATE DEFINER = `root`@`localhost` TRIGGER `Project_BEFORE_INSERT` BEFORE INSERT
ON `Project` FOR EACH ROW

BEGIN

    DECLARE temp DATETIME;

    IF NEW.start_date IS NULL THEN

        SET NEW.start_date = NOW();

    END IF;

    IF NEW.end_date IS NOT NULL AND NEW.start_date > NEW.end_date THEN

        SET temp = NEW.start_date;

        SET NEW.start_date = NEW.end_date;

        SET NEW.end_date = temp;

    END IF;

END$$
```

5)

```
USE `GTUProjectManagement`$$

CREATE DEFINER = CURRENT_USER TRIGGER
`GTUProjectManagement`.`User_BEFORE_INSERT` BEFORE INSERT ON `User` FOR EACH
ROW

BEGIN
```

```

IF NEW.company_email IS NULL THEN

    IF NOT EXISTS(SELECT * FROM User WHERE first_name=NEW.first_name AND
last_name = NEW.last_name) THEN

        SET NEW.company_email = CONCAT(NEW.first_name, NEW.last_name,
"@mycompany.com");

    ELSE

        SET NEW.company_email = CONCAT(NEW.created_at, "@mycompany.com");

    END IF;

END IF;

END$$

```

View Queries

1)

```

CREATE VIEW current_tasks AS SELECT name, description, created_at FROM Task
HAVING NOW() > created_at;

```

2)

```

CREATE VIEW the_users AS SELECT first_name, last_name, created_at, company_email
FROM User;

```

3)

```

CREATE VIEW task_class_priority AS SELECT DISTINCT Task.name AS task_name,
TaskClasses.name AS class_name FROM TaskClasses LEFT OUTER JOIN Task ON
TaskClasses.idTaskClasses = Task.priority_id;

```

4)

```

CREATE VIEW task_priority AS SELECT Priority.name AS priority_name, Task.name AS
task_name FROM Priority RIGHT OUTER JOIN Task ON Priority.idPriority = Task.priority_id;

```

5)

```

CREATE VIEW task_project AS SELECT first_name, last_name, company_email, Team.name
AS team_name FROM User FULL JOIN Team;

```