

CMPE 321 - Assignment III

Project Management System

Enis Simsar - 2014400219

Spring Term, 2018

Contents

1	Introduction	3
2	Interface	4
2.1	Guest User or Employee	4
2.2	Admin	7
2.3	Project Manager	11
3	Database	16
3.1	ER Model	16
3.2	Stored Procedures	18
3.3	Triggers	19
4	Conclusions & Assessment	21

1 Introduction

This application is about Project Management System. We have two types of logged in users: Admin and Project Manager. Admin can CRUD (create-read-update-delete) operations of Project Managers, Projects and Employees. In addition to this, admin can assign a project to a project manager. On the other hand, Project Manager can CRUD (create-read-update-delete) operations of Tasks. In addition to this, Project Manager can assign a task to a project and a task to an employee. You can register as an Admin, but you cannot register as a Project Manager. A project manager can be only created by an Admin.

I implemented this assignment with PHP Laravel Framework and MySQL. Also, I used some useful packages for frontend ui such as Admin LTE and table. For easy deployment, I used Docker. For using Docker, I wrote some configuration files and etc. Now, if you run '\$ bash deploy.sh', you can reach the system in your localhost with the port 8000. My project has a lot of directories, but with this design we can separate each module easily. I used generally ORM (Object Relational Model) in my project. However, in some parts like stored procedures, triggers and some additional parts, I used pure SQL queries. Laravel has SQL injection protection but I did not use this built-in functions, and customized login function. Then, I added my SQL injection protection to the project.

2 Interface

This project has three parts: Front End (for regular users or employees), Admin Panel and Project Manager Panel. I added screen shots for each part.

2.1 Guest User or Employee

This part of the project for guest, regular users or employees. This screens are only read-only. No one can change any entity in these pages.

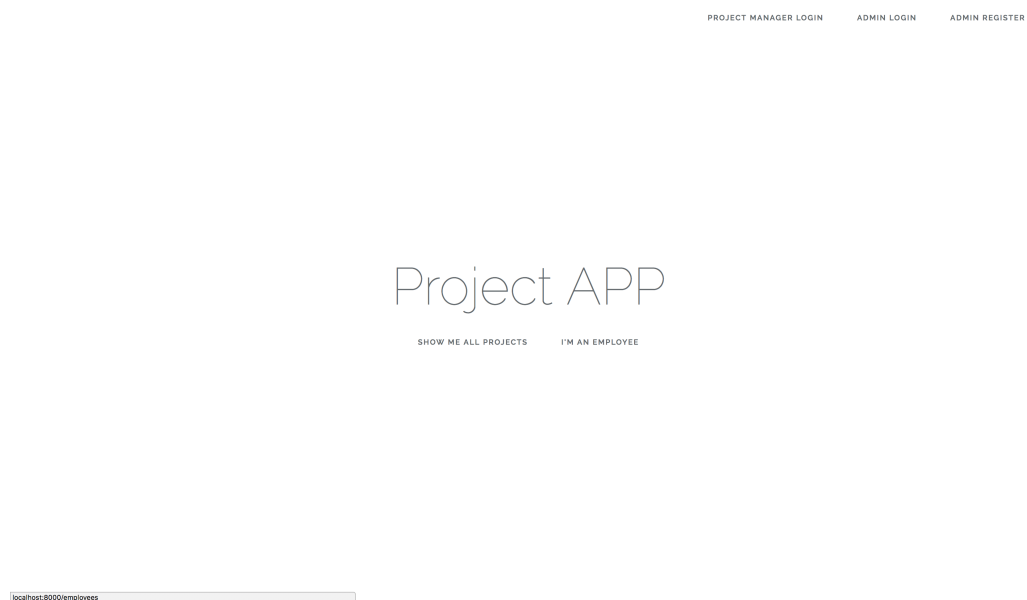


Figure 1: Home Page

Project App

Project Manager LoginAdmin LoginAdmin Register

Projects

Id ^	Name	Started At	Description	Completed	Created At	Project Link
1	Legendary Project	01.05.2018	This is a project about labor day.	No	06.05.2018	Go
2	Ordinary Project	10.05.2018	Hello! This is an ordinary project.	Yes	06.05.2018	Go
3	Ordinary Project 2	13.05.2018	Again. It is an ordinary project.	No	06.05.2018	Go
5	Ordinary Project 4	30.05.2018	Extra ordinary..	No	06.05.2018	Go

Figure 2: Projects Page

Project App

Project Manager LoginAdmin LoginAdmin Register

Project Managers

Id ^	Name	Email
1	P. Manager 1	pm_1@gmail.com
3	P. Manager 3	pm_3@gmail.com

Tasks

Id ^	Project Id	Name	Description	Started At	Duration	Completed
1	1	Task 1	Task 1 !! Yupp!!	01.05.2018	10 Days	No
2	1	Task 2	Task 2...	02.05.2018	3 Days	No
3	1	Task3	Not in the period!!	13.05.2018	2 Days	No

Figure 3: Project Page





Project App		Project Manager Login	Admin Login	Admin Register
Employees				
Id ^	Name	Created At	Employee Link	
1	Employee 1	06.05.2018		
3	Employee 3	06.05.2018		
4	Employee 4	06.05.2018		
5	Employee 5	06.05.2018		

Figure 4: Employees Page

Project App

Project Manager LoginAdmin LoginAdmin Register

Tasks

Id ^	Project Id	Name	Description	Started At	Duration	Completed
1	1	Task 1	Task 1 !! Yuppil	01.05.2018	10 Days	No
3	1	Task3	Not in the period!!	13.05.2018	2 Days	No

Figure 5: Employee Page

2.2 Admin

This part of the project for admin users. This screens can only be reached by a logged in admin.

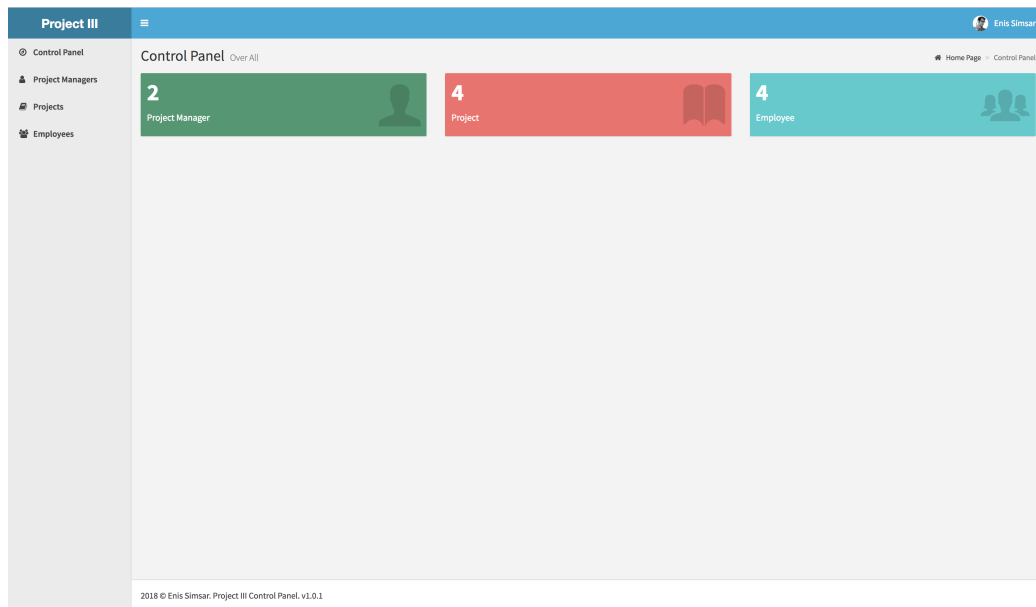


Figure 6: Admin Dashboard Page

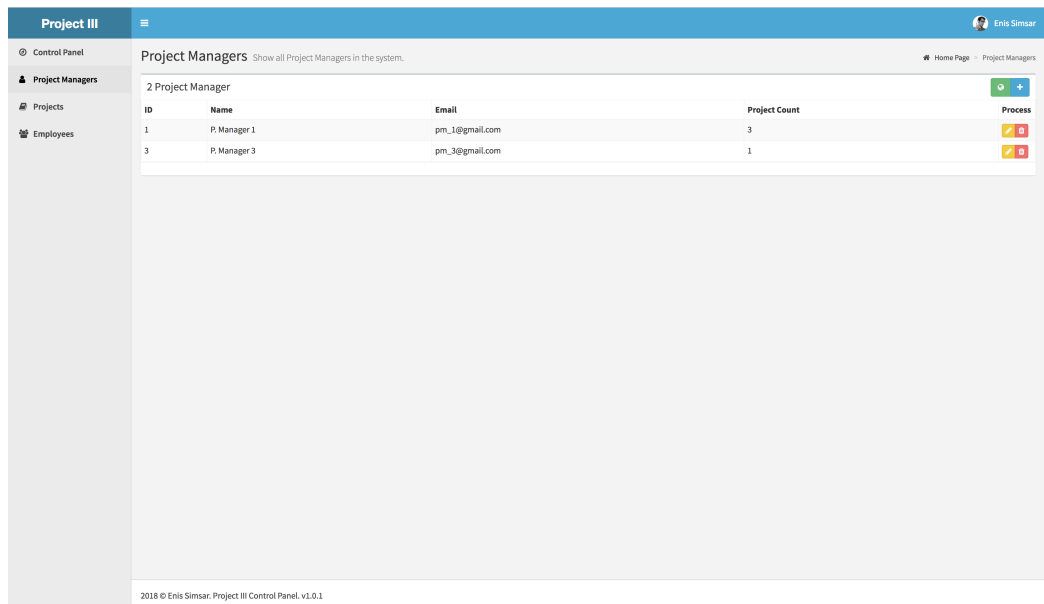


Figure 7: Project Manager Index Page

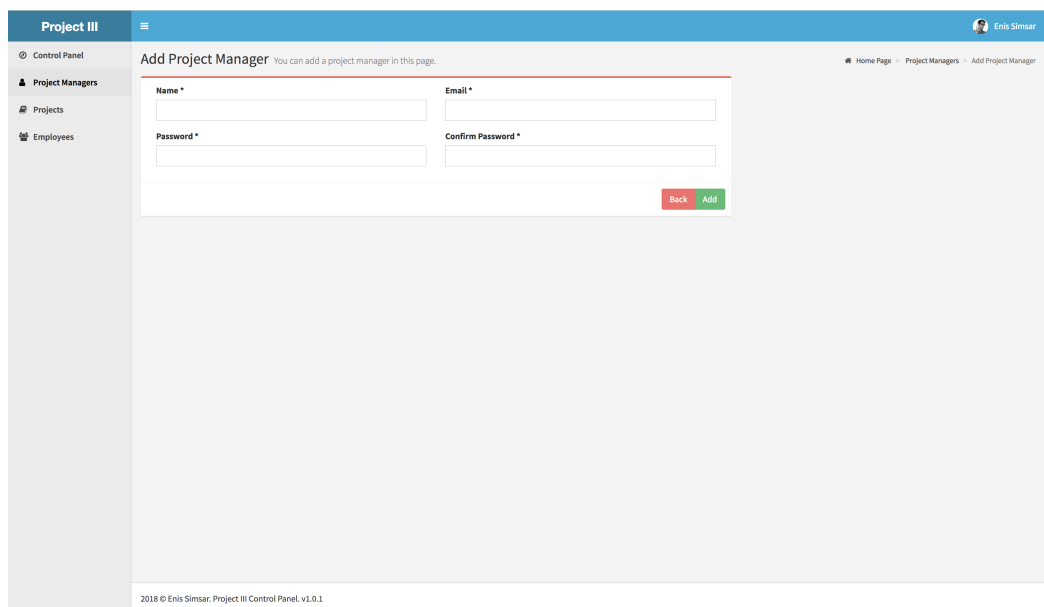


Figure 8: Project Manager Create Page

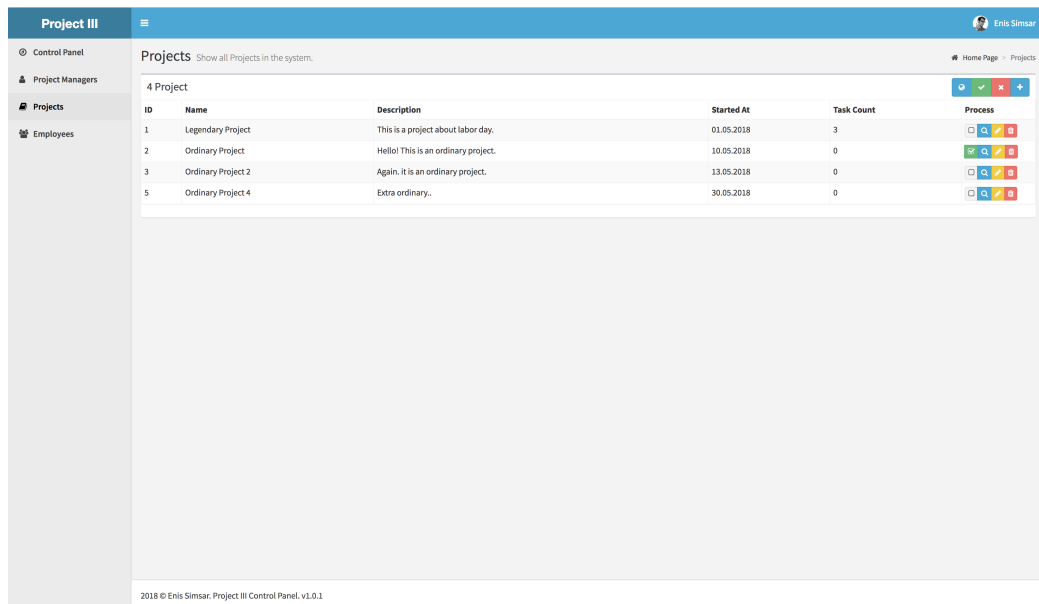


Figure 9: Project Index Page

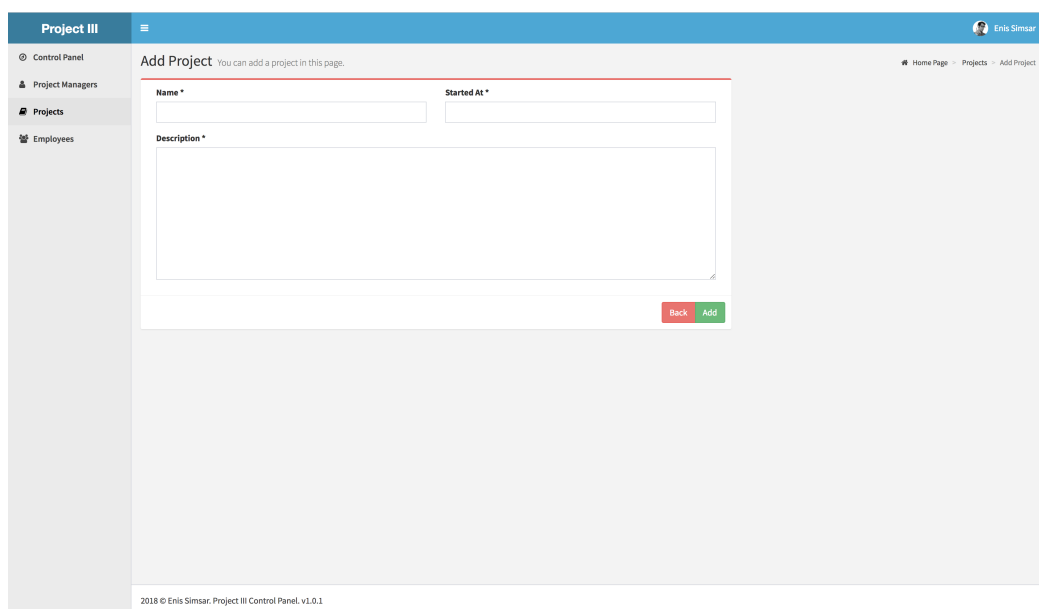


Figure 10: Project Create Page

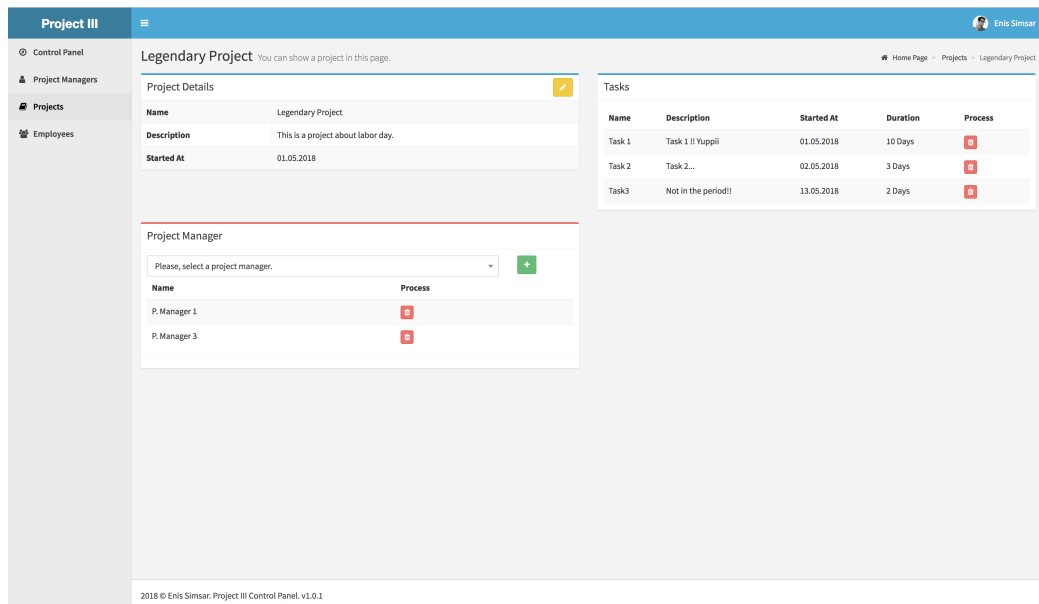


Figure 11: Project Manager Show Page

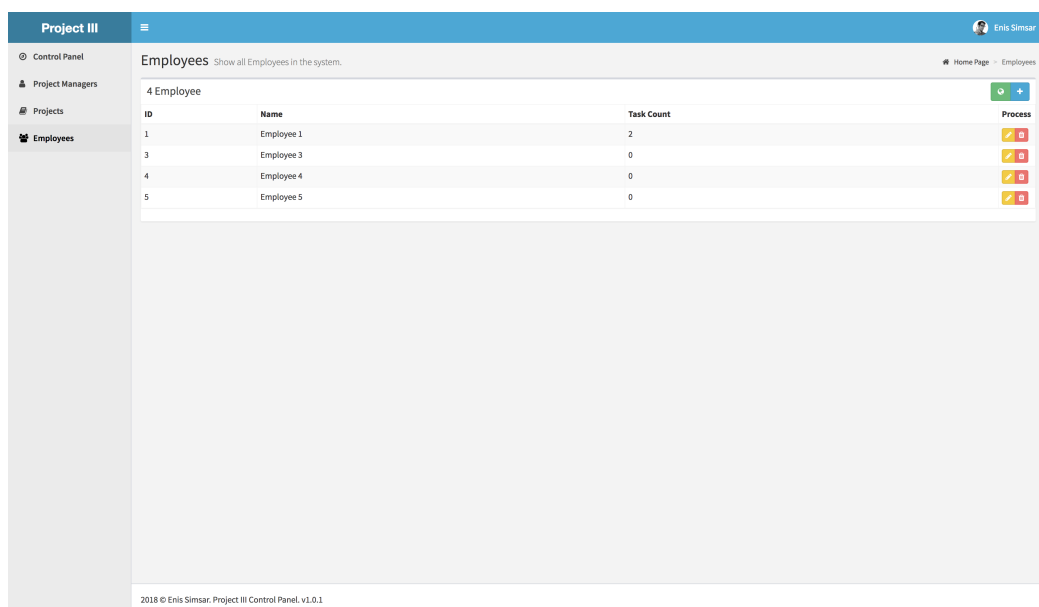


Figure 12: Employee Index Page

2.3 Project Manager

This part of the project for project manager users. This screens can only be reached by a logged in project manager.

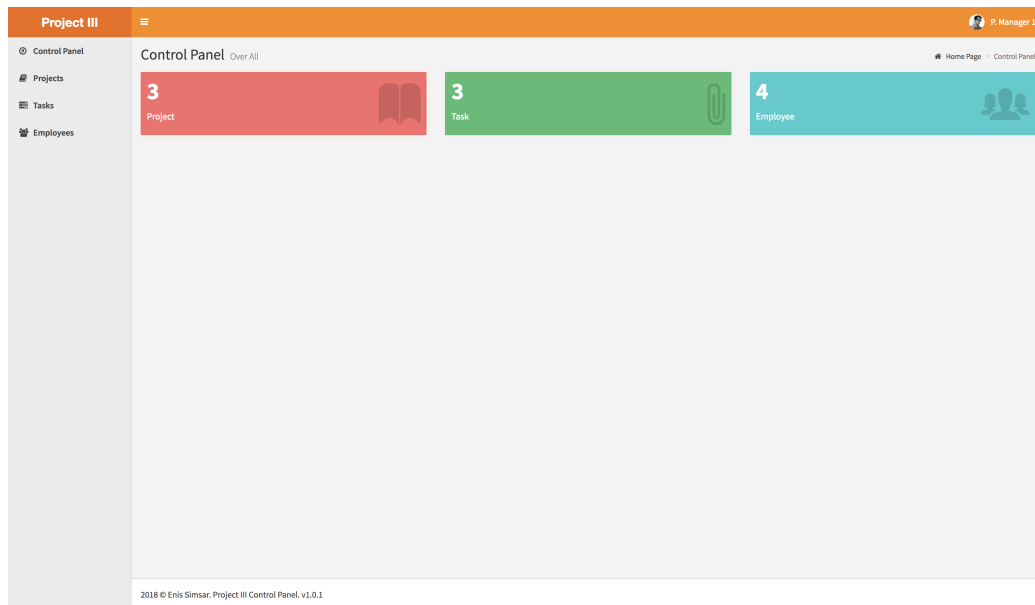


Figure 13: Project Manager Dashboard Page

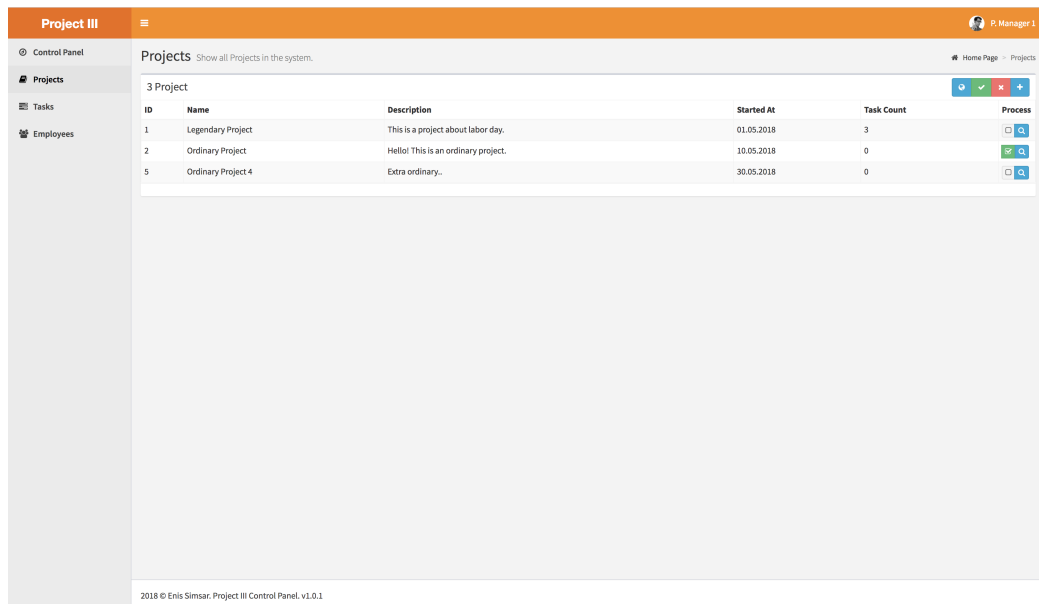


Figure 14: Project Index Page

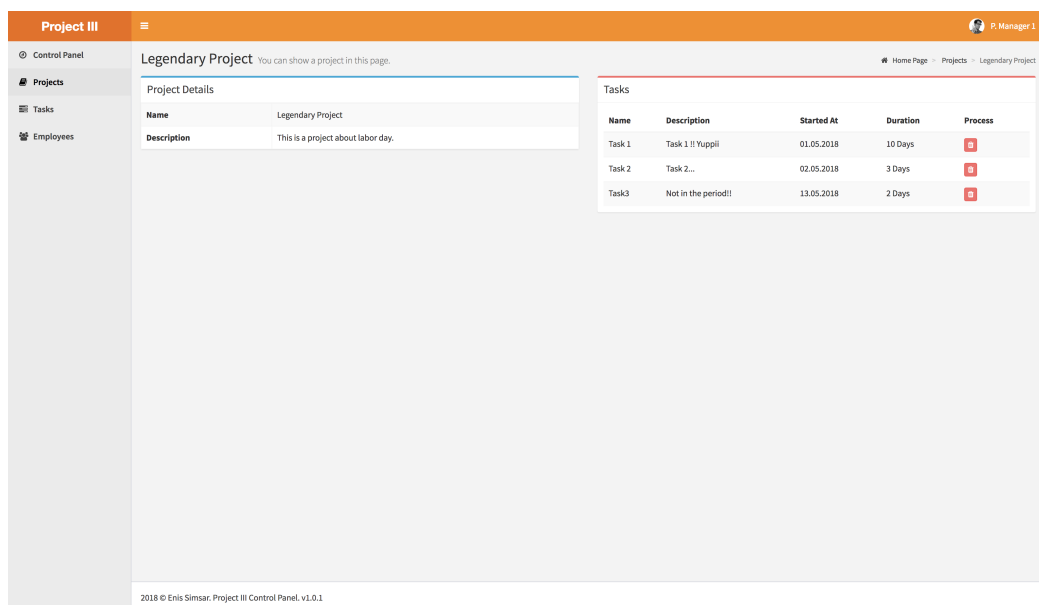


Figure 15: Project Show Page

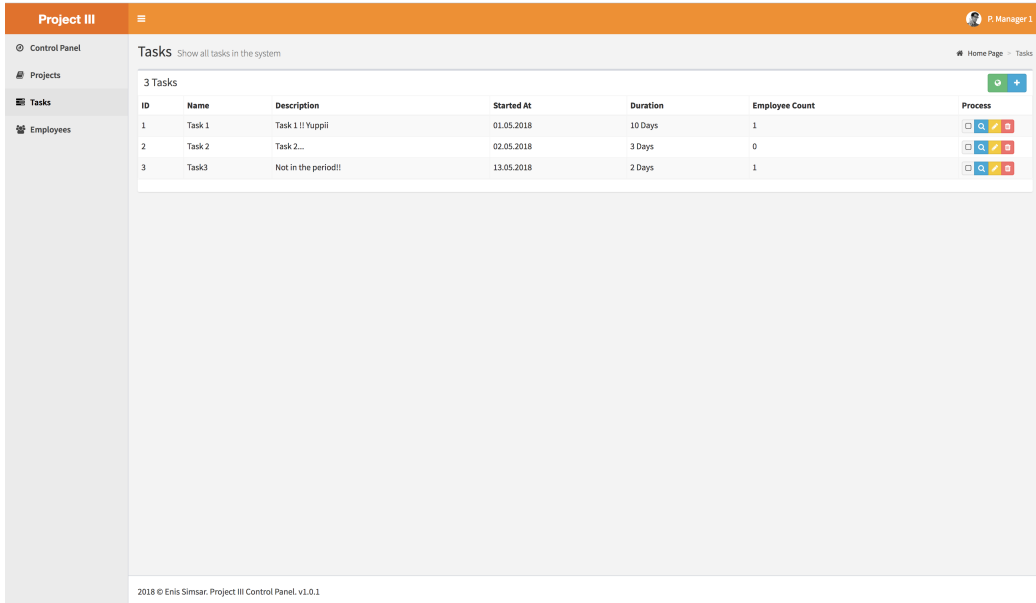


Figure 16: Task Index Page

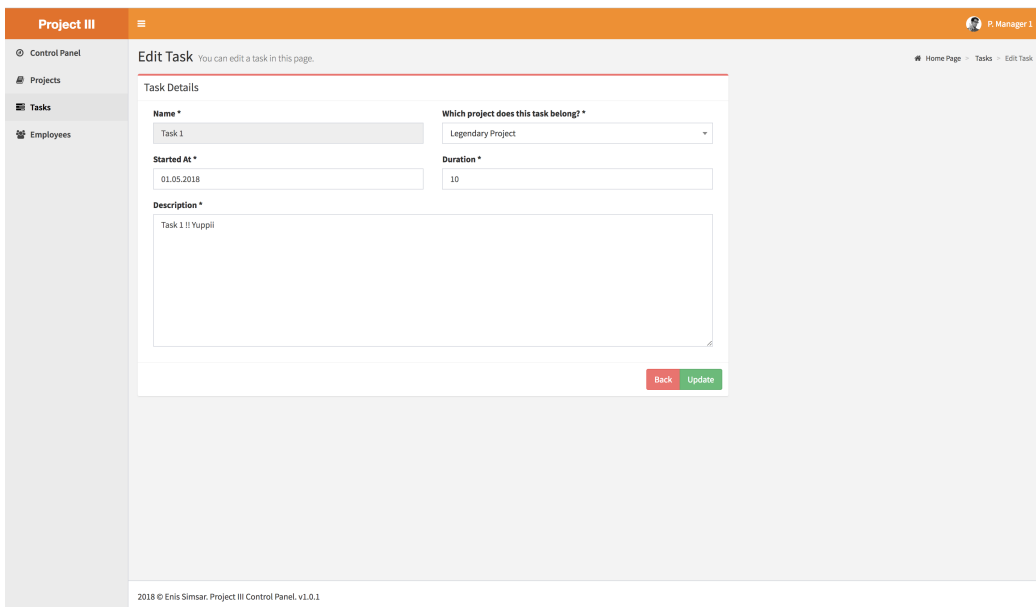


Figure 17: Task Edit Page

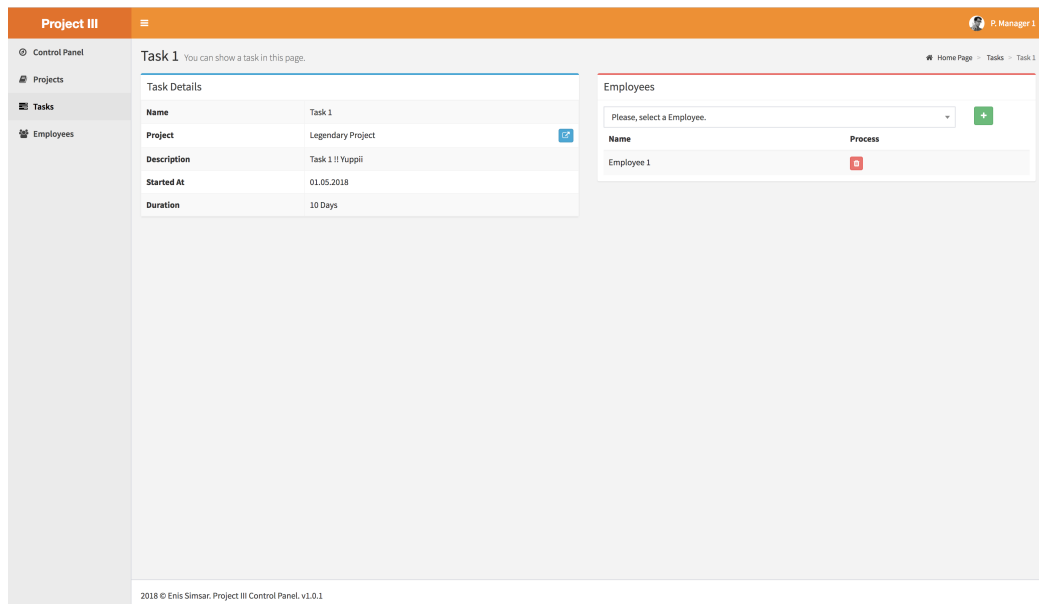


Figure 18: Task Show Page

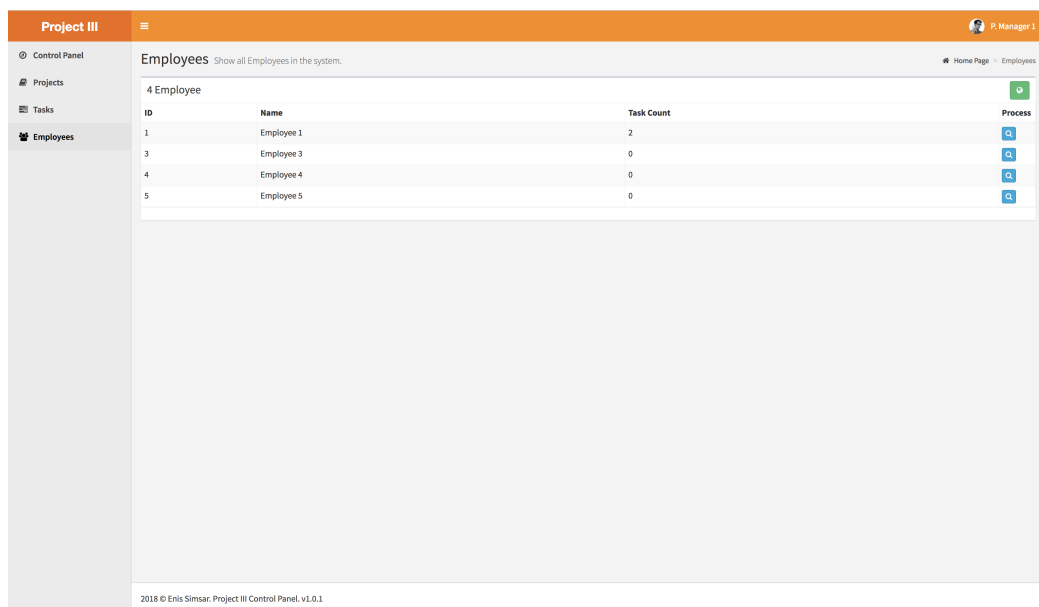


Figure 19: Employee Index Page

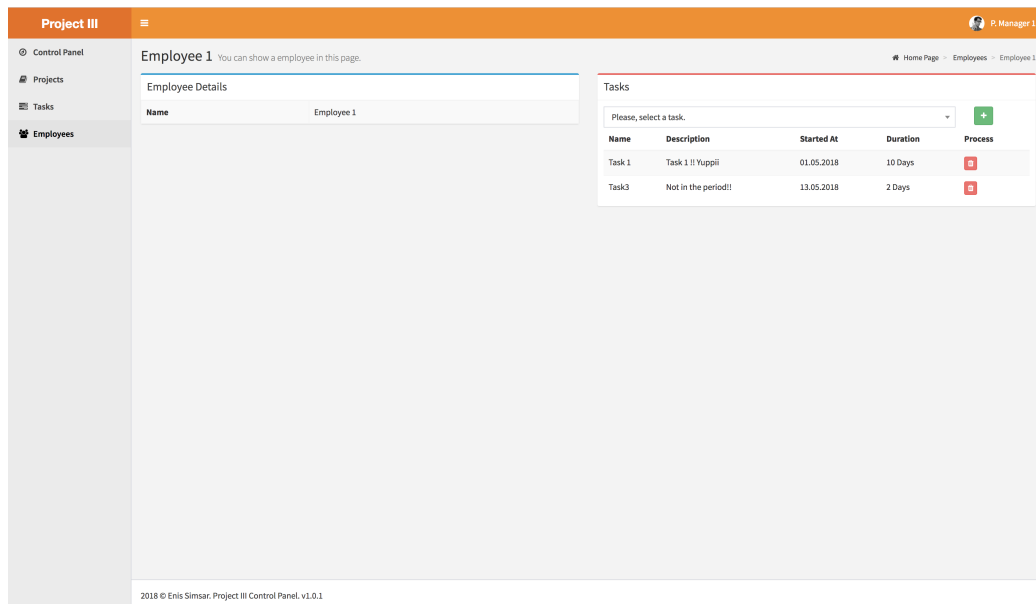


Figure 20: Employee Show Page

3 Database

3.1 ER Model

I added two ER models as follows.

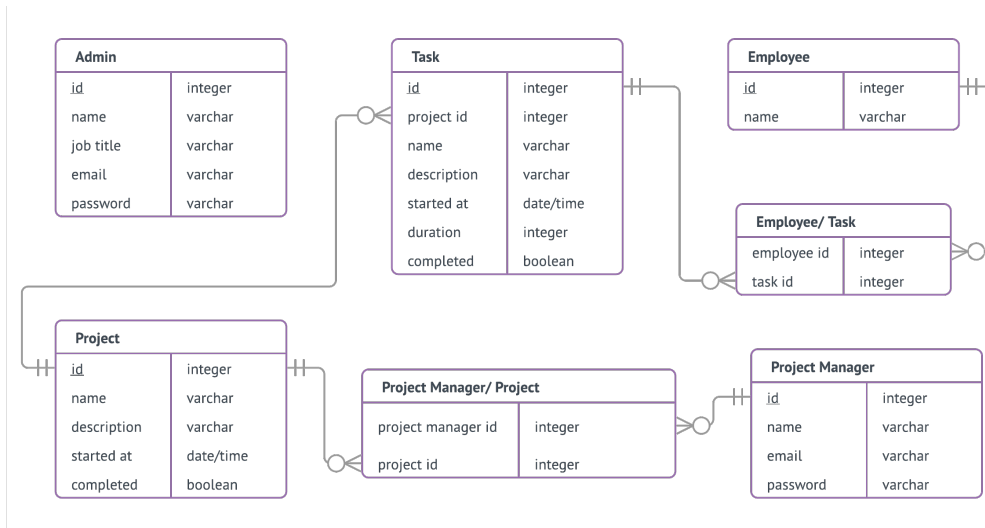


Figure 21: Database ER Diagram

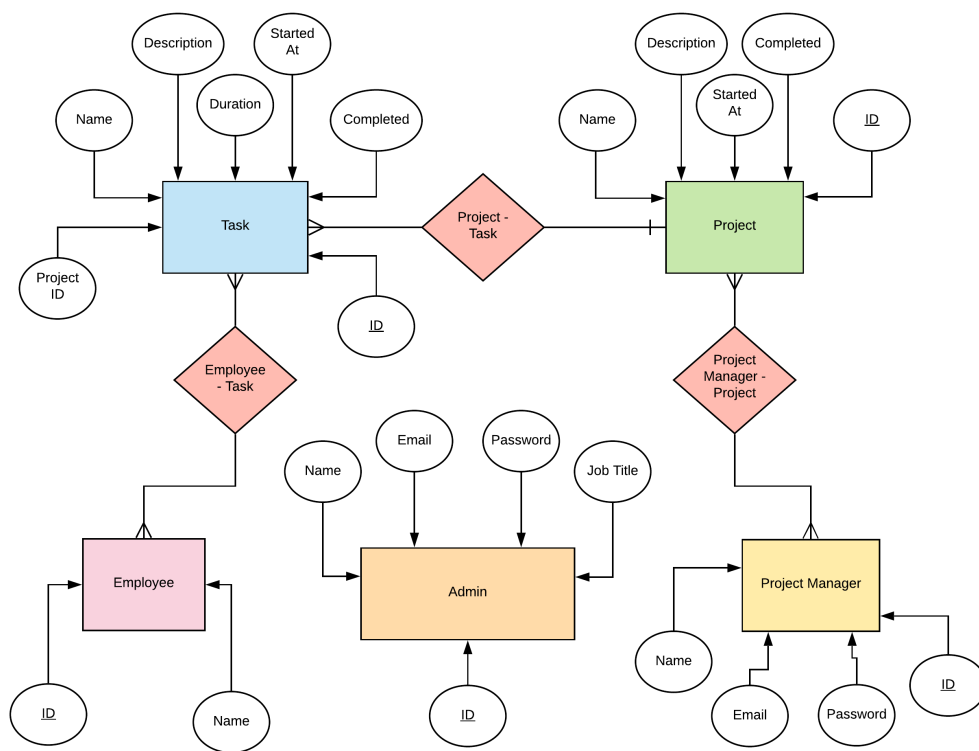


Figure 22: ER Diagram

3.2 Stored Procedures

We have two stored procedures as follows.

```
CREATE PROCEDURE `completed_projects`(IN project_manager_id TEXT)
BEGIN
  IF (project_manager_id = 'ALL') THEN
    SELECT * FROM projects WHERE projects.completed = TRUE;
  ELSE
    SELECT * FROM projects WHERE
      projects.id IN (
        SELECT project_manager_project.project_id FROM project_manager_project
        WHERE project_manager_project.project_manager_id = CAST(project_manager_id AS UNSIGNED)
      ) AND projects.completed = TRUE;
  END IF;
END;
```

Figure 23: Completed Projects Stored Procedure

```
CREATE PROCEDURE `not_completed_projects`(IN project_manager_id TEXT)
BEGIN
  IF (project_manager_id = 'ALL') THEN
    SELECT * FROM projects WHERE projects.completed = FALSE;
  ELSE
    SELECT * FROM projects WHERE
      projects.id IN (
        SELECT project_manager_project.project_id FROM project_manager_project
        WHERE project_manager_project.project_manager_id = CAST(project_manager_id AS UNSIGNED)
      ) AND projects.completed = FALSE;
  END IF;
END
```

Figure 24: Not Completed Projects Stored Procedure

3.3 Triggers

We have two triggers as follows.

```
CREATE TRIGGER add_project_to_least_project_pm
AFTER INSERT
ON projects
FOR EACH ROW
BEGIN
    IF EXISTS(SELECT * FROM project_managers) THEN
        SELECT project_managers.id
        FROM project_managers
        LEFT JOIN project_manager_project
        ON project_managers.id = project_manager_project.project_manager_id
        GROUP BY project_managers.id
        ORDER BY COUNT(project_manager_project.project_id) ASC
        LIMIT 1 INTO @pm_id;
        INSERT INTO project_manager_project
        (project_id, project_manager_id)
        VALUES (NEW.id, @pm_id);
    END IF;
END;
```

Figure 25: Add New Project to PM with Least Projects Trigger

```
CREATE TRIGGER remove_free_relations
BEFORE DELETE
ON employees
FOR EACH ROW
BEGIN
    DELETE FROM employee_task
    WHERE employee_task.employee_id = OLD.id;
END;
```

Figure 26: Delete Employee-Task Relation When Employee is deleted

4 Conclusions & Assessment

In this assignment, I implemented a project management system. It has three main components: Front End which serves information to guest users or employees, Admin Dashboard which serves information to admin users and Project Manager Dashboard which serves information to project manager users.

I implemented this project with PHP Laravel and MySQL. I used some useful packages. In addition to these, I used Docker to easily deploy my project.

Before this assignment, I have not known the stored procedure and trigger concepts in SQL. I researched these concepts on the web and gained some useful information. I learned these concepts and liked them. Actually, in the past, I have been worked on big projects with Laravel, MySQL, PostgreSQL, MongoDB (NoSQL), Python Tornado and etc. However, with this project, I improved my SQL abilities.

I enjoyed while I was implementing this project. However, I was bored while I was implementing some parts because I worked on big projects that include Front End and Back End as I mentioned before. Overall, I liked this project because I learned different concepts that are related with SQL, stored procedures and trigger.

In conclusion, this is a good implementation for Project Management System because it has all functionality and addition to this, simple UI design. I mean it has some pros and cons just like other application, but I am happy to construct an application like this.