



Software Engineering Project

Task Monitoring App

By: Malek Emad Eldin	6168
Mahmoud Ahmed	6375
Mostafa Adel	6189

Contents

contents	1
1.Introduction.....	2
1.1 Purpose of Document	2
1.2 Document Convention	2
1.3 Background	2
1.4 Scope.....	3
1.5 System Purpose	3
2. General Description.....	4
2.1 Product Description	4
2.2 System and Application Architectures	4
2.3 Development Enviroment	5
2.4 Software Process and Implmentation	5
3. Functional Requirements	6
3.1 High Priority Tasks.....	6
3.2 Medium Priority Tasks	8
3.3 Low Priority Tasks	10
4. Non Functional Requirements	12
4.1 Description	12
4.2 External Interfaces and Minimum requiremnts.....	12
5.Preliminary Use Case Models and Sequence Diagrams	13
5.1 Use Case Diagrams.....	13
5.2 Use Case Description(For selected Cases)	14
5.3 Sequence Diagrams	18
6. Activity Diagrams	20
7. Other Diagrams	22
8. Testing Functionalities	23

1.Introduction

1.1 Purpose of Document

The purpose of this document is to define and describe the **requirements**(functional/non-Functional) of the term project for the **software engineering course** (CC472) taught in **spring 2022** at the **Faculty of Engineering, Alexandria university** and to spell out the system's **functionality** and its **constraints**.

As well as, describing the system's **architecture** and approach we used in the **realization** of the system.

1.2 Document Convention

This Document uses the following **Conventions** and **abbreviations** in the description of the application which are mentioned in **table 1.2.1** below.

DB	Database
UML	Unified Modelling Language
SQL	Structured Query Language
ER	Entity Relationship
SPA	Single Page Application

Table 1.2.1: Conventions used

1.3 Background

Modern day-to-day life of people in major cities is **very demanding** and the **schedules** are equally hectic. In such times, it is **practically impossible** to keep a **track** of all the **activities/appointments**. Many a times, it happens that we may miss an important task; for example: taking medicines, attending a meeting, returning library books, paying the bills etc. And this **cycle** can keep going on **endlessly**. The **human mind** is not designed to **multitask**, it needs to

work things out **one at a time**. This requires us to maintain our focus on the task at hand, and as a result other important things take a **backstage** and some may even **slip out** of our minds. This motivated us to implement task monitoring System described in the document.

1.4 Scope

The scope of this project is a **web-based** system that supports the assigning of user new tasks, deleting them if needed or mark them as **completed**.

Issues of **Advanced website security is not implemented**.

1.5 System Purpose

Those who will **primarily benefit** from the new system are the users who upon **completion** of the project will be able to **add** tasks, **prioritize** them and **search** according to certain criteria's such as **priority**.

2.General Description

2.1 Product Description

The product should make assignment of tasks easier with convenient interface, as well as viewing the tasks.

2.2 Architecture

There are generally two types of architectures in which a software project may be interested in which are **System** and **Application** respectively.

Concerning The **system architecture** which relies on **client/server model** at which a client via website can request certain data which the server responds to, as well as existence of centralized database in the server being accessed which can be viewed in the figure 2.1 below.

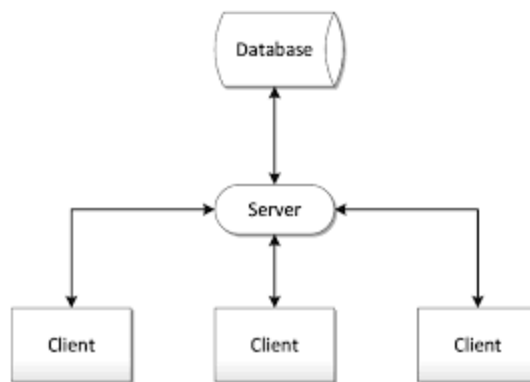


Figure 2.1 client/server

Concerning the **application architecture**, we made our **web application** adhere to architecture known as **Single-Page Application (SPAs)** which instead of loading completely new pages from the server each time the user interacts with it, single page web applications allow for a dynamic interaction by means of passing **updated components** to the **current page** by using **react framework** as well as it is considered as **Event processing system** due to the fact it relies on the interactions coming from the environment.

Generally, **single-page applications** prevent interruptions in **user experience**, they, in a way, **resemble** traditional desktop applications.

2.3 Development Environment

Operating system: Windows.

database: MySQL Database

Backend: Nodejs v16.14.2

Backend Framework: ExpressJS

Frontend Framework: ReactJS

Software Used: MYSQL Workbench, VsCode, GitHub Desktop

2.4 Software Processes

We found it more **convenient** since the number of team members are limited to **three** , to use **agile mythology** by arranging a meeting in which we **divide** the project into **different tasks**, each team member is assigned a task which upon **completion**, we integrate it with the code written by other members and test it and didn't focus on **documentation** upon the starting of the project due to the tight deadlines which would slow our **development** process.

However, we realize there are certain limitations according to our implementation of **agile** resulting that a client is not available to be able to ask him and make him participate in testing due to it is not somehow commercial project thus we had to assume what would the client want.

3.Functional Requirements

3.1 High Priority Tasks

Requirement ID: Req-001	Requirement Name: Login
Primary Actor: User	Criticality: High
Status: Mandatory	
Description: User of application will open location task management application. User will be asked to enter his Username and Password then press a button which will redirect him into the home page in case of correct credentials entered.	
Reason behind Criticality: As it is an essential part of the system, and it would imply to different tasks to appear according to which user is signed in.	

Requirement ID: Req-002	Requirement Name: Assigning Tasks
Primary Actor: User and Server	Criticality: High
Status: Mandatory	
<p>Description: User of application after login inside the system, will see a group of input forms which will have labels inside them such as Task name, Task Description, and a drop-down menu in which they can Select the priority of the task(high/medium/low) and date input.</p> <p>After filling in the required data they can press a button to be inserted inside the DB.</p>	
<p>Reason behind Criticality: As it is an essential part of the system, and it would lead to the viewing of the tasks later and applying operations upon them.</p>	

3.2 Medium Priority Tasks

Requirement ID: Req-003	Requirement Name: Viewing Tasks
Primary Actor: User and Server	Criticality: Medium
Status: Mandatory	
Description: User of application can click a button with label of view inside a navbar in the home page which will make him able to view all the tasks he inserted before inside the DB.	
Reason behind Criticality: It is considered important as in terms of a requirement but at the same type there are not other requirements which is completely dependent on it and wouldn't conflict with many tasks inside the system.	

Requirement ID: Req-004	Requirement Name: Searching Tasks
Primary Actor: User and Server	Criticality: Medium
Status: Mandatory	
Description: User of application inside view age, will have input component in which they can type search query and drop-down menu with label of filter by (in progress/upcoming/overdue) and sort by menu to choose sorting criteria (none/date/priority).	
Reason behind Criticality: Important feature of the system but not that critical. As well as it would require effort to implement it.	

3.3 Low Priority Tasks

Requirement ID: Req-005	Requirement Name: Logout
Primary Actor: User and Server	Criticality: Low
Status: Mandatory	
Description: User of application can click a button with label of logout inside a navbar which will make him able to logout outside the system and redirect him to login page.	
Reason behind Criticality: It is not considered as backbone functionality of the system and no other requirement is dependable on it thus it is low priority Task.	

Requirement ID: Req-006	Requirement Name: Editing Task
Primary Actor: User and Server	Criticality: low
Status: Mandatory	
Description: User of application inside view age will have button labeled edit which upon clicking, enables user to edit fields of the task.	
Reason behind Criticality: Important feature of the system but have minimum conflicts with other tasks.	

Requirement ID: Req-007	Requirement Name: Alerting due Task
Primary Actor: Server	Criticality: Low
Status: desirable	
Description: if any task is the system is due, the server will send an email to the user notifying him of the task name and that it is due.	
Reason behind Criticality: It is a desirable feature of the system, and it won't take much effort in implementation.	

4.Non-Functional Requirements

4.1 General Description

CORRECTNESS: User should be able to view the tasks he assigned to himself only and by no other circumstance the tasks of any other user.

Portability: The system shall be designed in a way that shall allow it to be runnable on different browsers and operating Systems.

Modifiability: Modification term is used for making changing in the functionality of the system. The system is flexible. If organization want to modify the functionality of the system, it is possible. Software is adaptive to changing. Changing in the software is very easy as we introduced another DB table to be accommodate if an organization want to use the software for collaboration and assigning tasks.

4.2 External Interfaces

4.2.1 User interfaces

GUI

The user interface for this program is provided as a website which has multiple Components such as (buttons/text fields/navbars) etc. which the user interacts with.

CLI

There is no command line interface

API

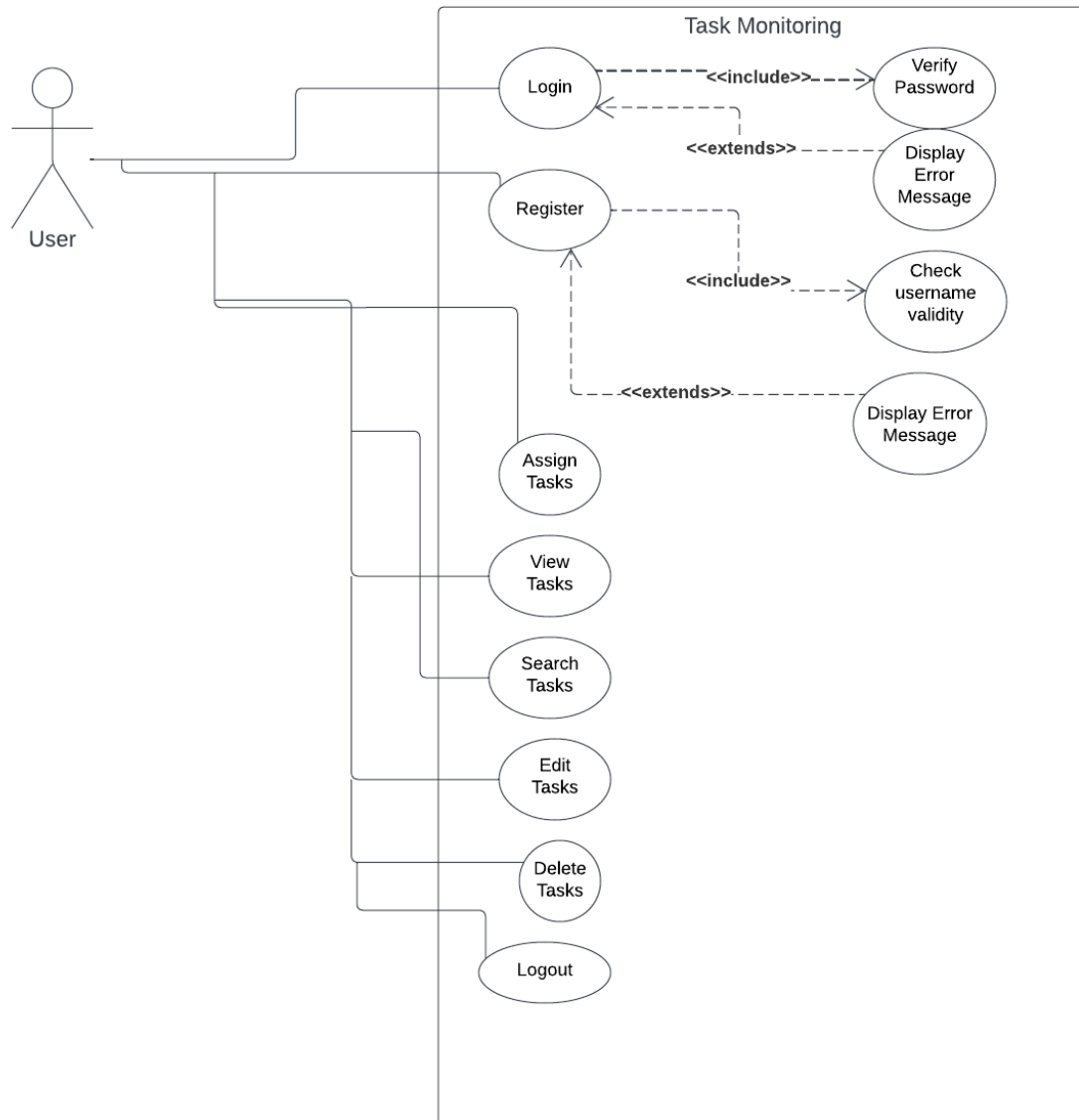
There is no API used.

4.2.2 Minimum Requirements

- **Modern browser which supports JavaScript and HTML.**
- **Any operating System installed inside the device.**

5. Preliminary Use Case Models and Sequence Diagrams

5.1 Use case Diagram



5.2 Use Case Description(For selected Cases)

Use Case Name:	Register
Basic Flow:	<ol style="list-style-type: none">1. The use case starts when a user indicates that he wants to register.2. The system requests a username and password.3. The user enters a username and password.4. The system checks that the username does not duplicate any existing registered usernames.5. The system executes use case <i>Login</i>. <p>The system redirects the user into login Page.</p>
Alternative Flows:	<ul style="list-style-type: none">• Step 4: If the username duplicates an existing username the system displays a message, and the use case goes back to step 2. <p>6. Step 5: If the user does not enter a required field, a message is displayed, and the use case repeats step 3.</p>
Extension Points:	<ul style="list-style-type: none">• Login
Preconditions:	none
Postconditions:	The user can now Login and start using the System.

Use Case Name:	Login
Basic Flow:	<ol style="list-style-type: none"> 1. The use case starts when a user indicates that he wants to login. 2. The system requests the username and password. 3. The user enters his username and password. 4. The system verifies the username and password against all registered users. 5. The system starts a login session and displays a welcome message based on the user's preferences.
Alternative Flows:	<ul style="list-style-type: none"> • Step 4: • if username or password is invalid, the system displays an error message, and the use case goes back to step 2.
Extension Points:	<ul style="list-style-type: none"> • None
Preconditions:	User is registered
Postconditions:	The user can now Login and start using the System.

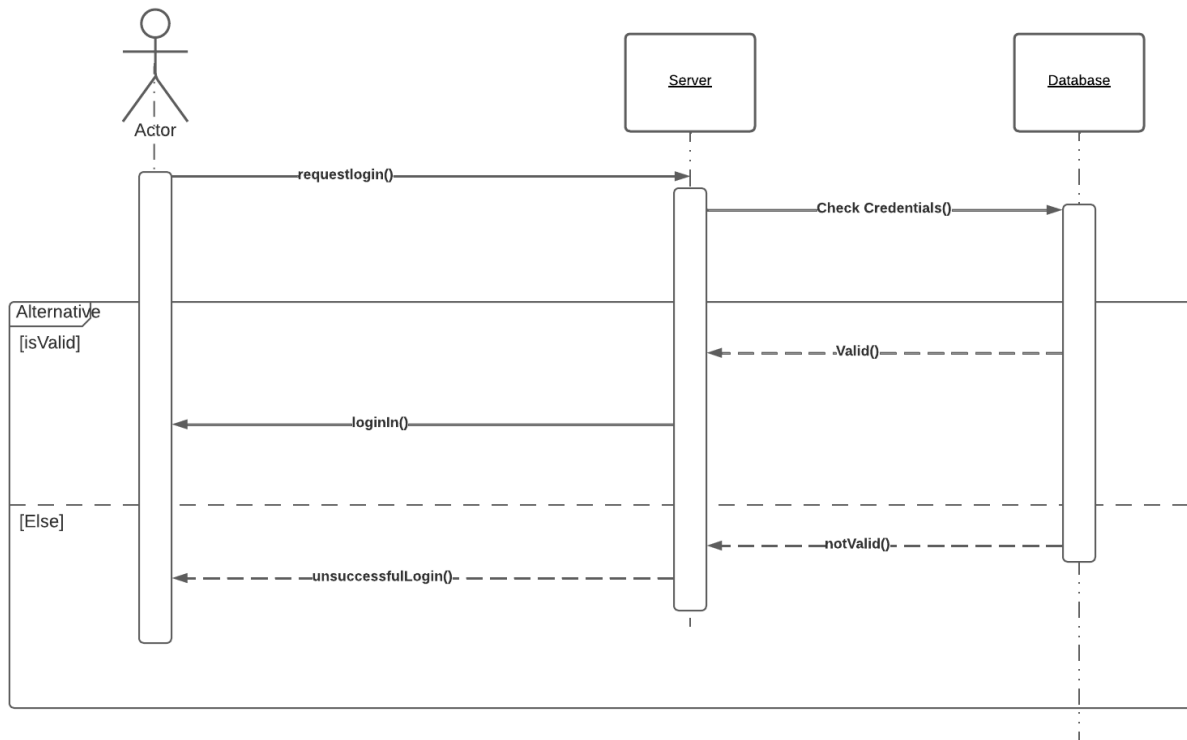
Use Case Name:	View Tasks
Basic Flow:	<ol style="list-style-type: none"> 1. The use case starts when a user indicates that he wants to View his tasks. 2. The user clicks on button labeled view which is inside the main navbar. 3. The system renders all tasks stored inside the DB associated with the user.
Alternative Flows:	<ul style="list-style-type: none"> • None
Extension Points:	<ul style="list-style-type: none"> • None
Preconditions:	User is Logged In
Postconditions:	None

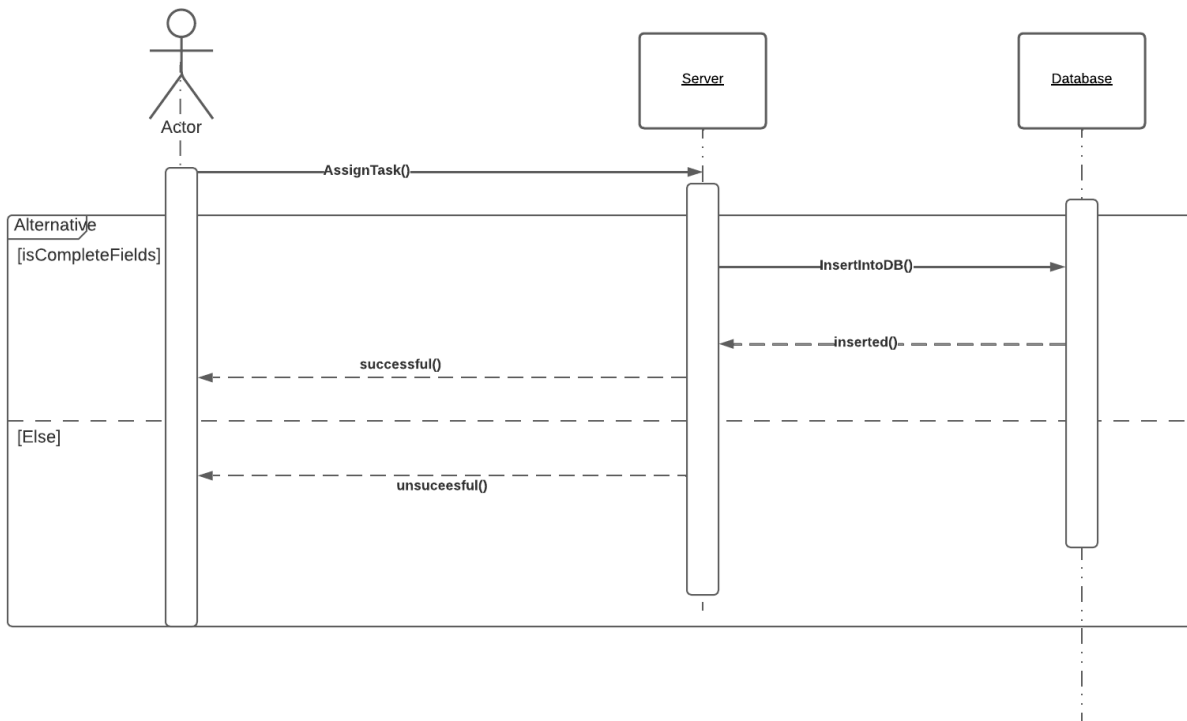
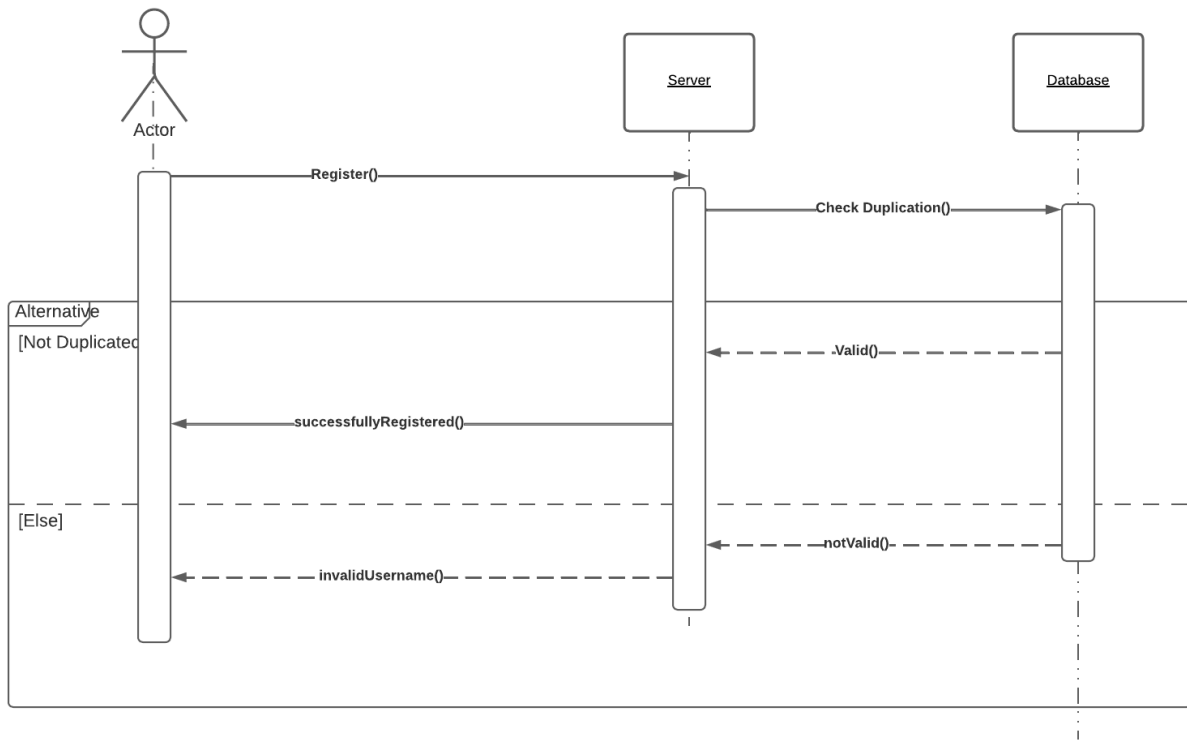
Use Case Name:	Assign Tasks
Basic Flow:	<ol style="list-style-type: none"> 1. The use case starts when a user indicates that he wants to assign new task. 2. The system requests task name, task description, priority, and date. 3. The user enters his task name, task description, priority, and date. 4. The system verifies that all fields are filled. 5. The system inserts successfully into the DB.
Alternative Flows:	<ul style="list-style-type: none"> • Step 4: • if there any field that is not filled, the system displays an error message, and the use case goes back to step 2. • If date of the task is earlier than today's date, the system displays an error message, and the use case goes back to step 2
Extension Points:	<ul style="list-style-type: none"> • None
Preconditions:	User is Logged In
Postconditions:	None

Use Case Name:	Delete Task
Basic Flow:	<ol style="list-style-type: none"> 1. The use case starts when a user indicates that he wants to delete new task. 2. The user presses delete button beside the task. 3. The System removes the task successfully.
Alternative Flows:	<ul style="list-style-type: none"> • None
Extension Points:	<ul style="list-style-type: none"> • None
Preconditions:	User is Logged In
Postconditions:	None

Use Case Name:	Search Tasks
Basic Flow:	<ol style="list-style-type: none"> 1. The use case starts when a user indicates that he wants to search a task. 2. The user types inside search bar his desired Criteria. 3. The user specifies filter by criteria and sort by 4. The user presses apply button. 5. The system returns all tasks matching his criteria.
Alternative Flows:	<ul style="list-style-type: none"> • None
Extension Points:	<ul style="list-style-type: none"> • None
Preconditions:	User is Logged In
Postconditions:	None

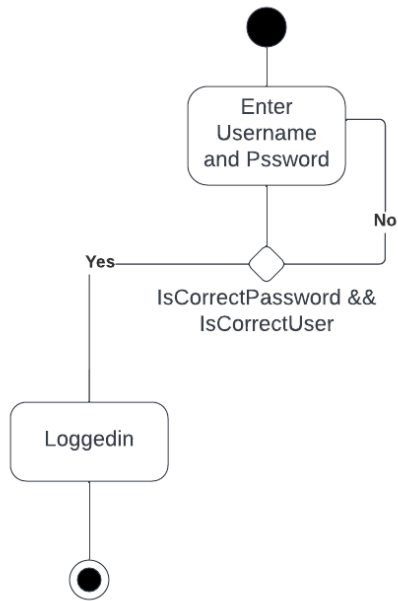
5.2 Sequence Diagrams





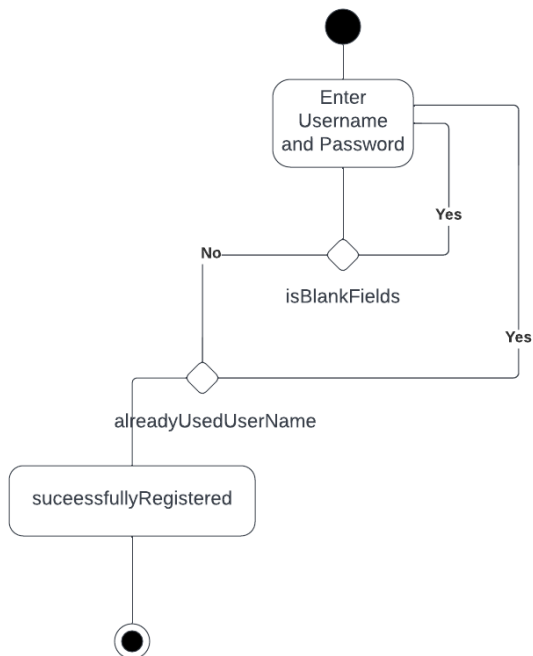
6.Activity Diagrams

- **Login Activity**



Description: The User is asked to enter his credentials to log in, in case of correct one he will successfully login otherwise he will be asked to re enter them.

- **Register Activity**



Description:

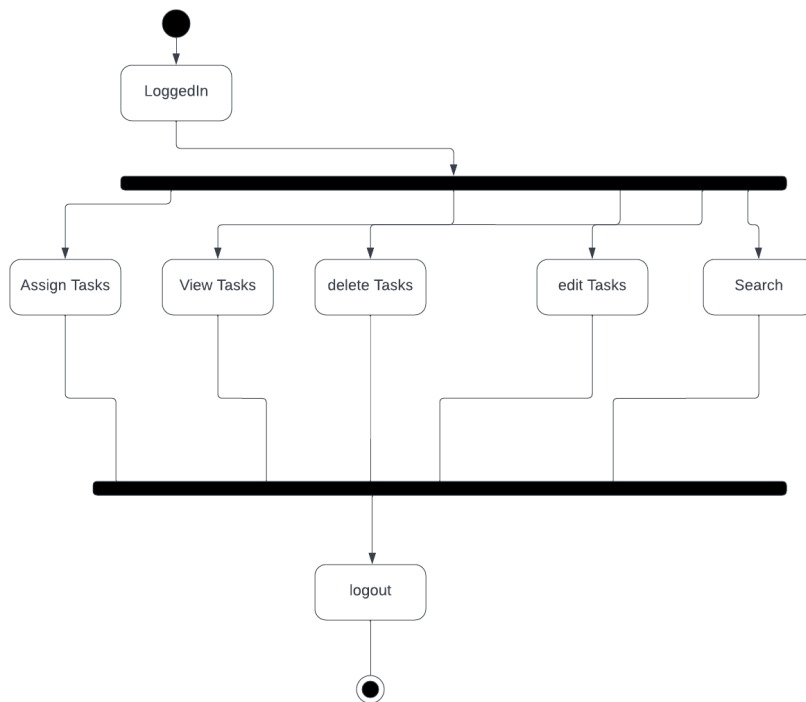
User is asked to enter username and password, if any of these fields are blank user is asked to reenter them.

If the username is already taken, he is also asked to enter a new one which leads upon successful in successfully registered state.

Activity Diagram For other activities

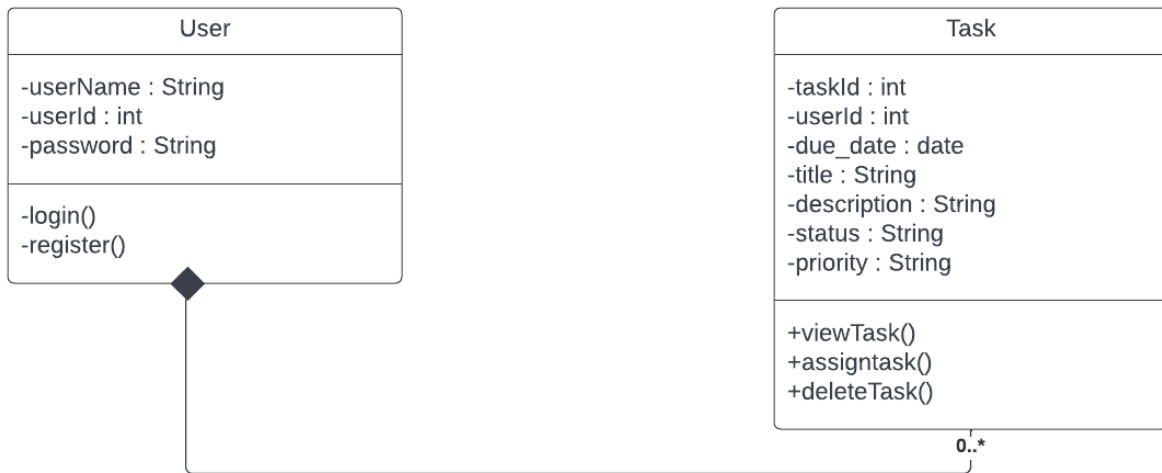
Description:

After login in user can do one of the activities described in the document, we divided into multiple diagrams for the sake of simplicity then after finishing what he wants, then he can logout.

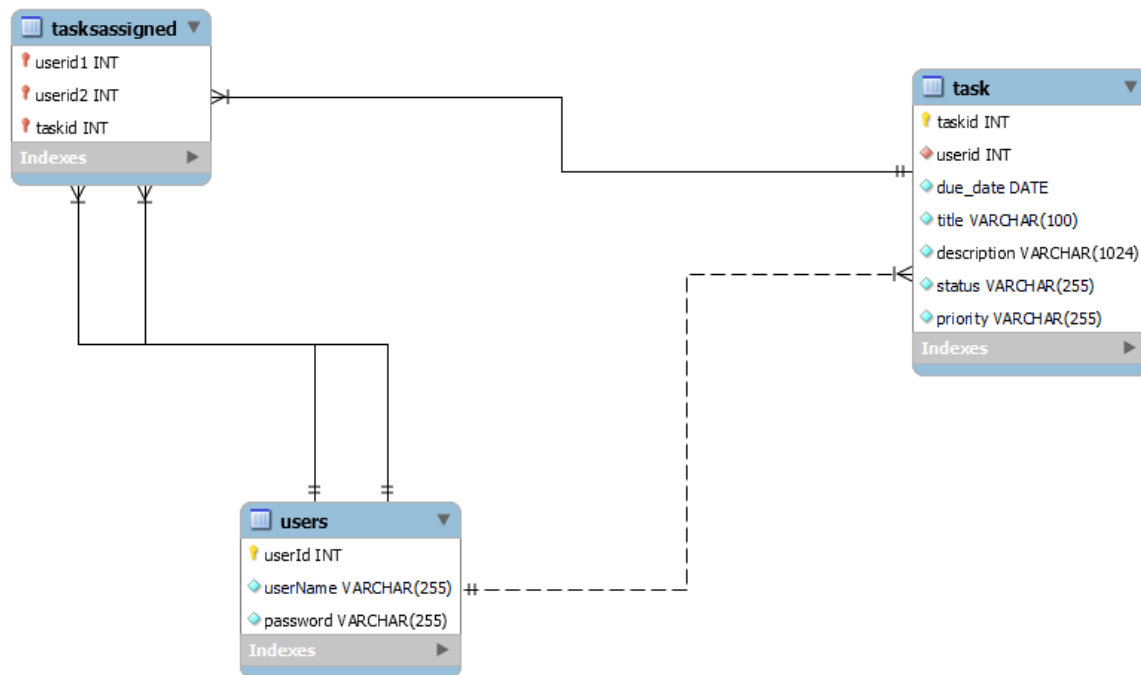


7. Other Diagrams

Class Diagram



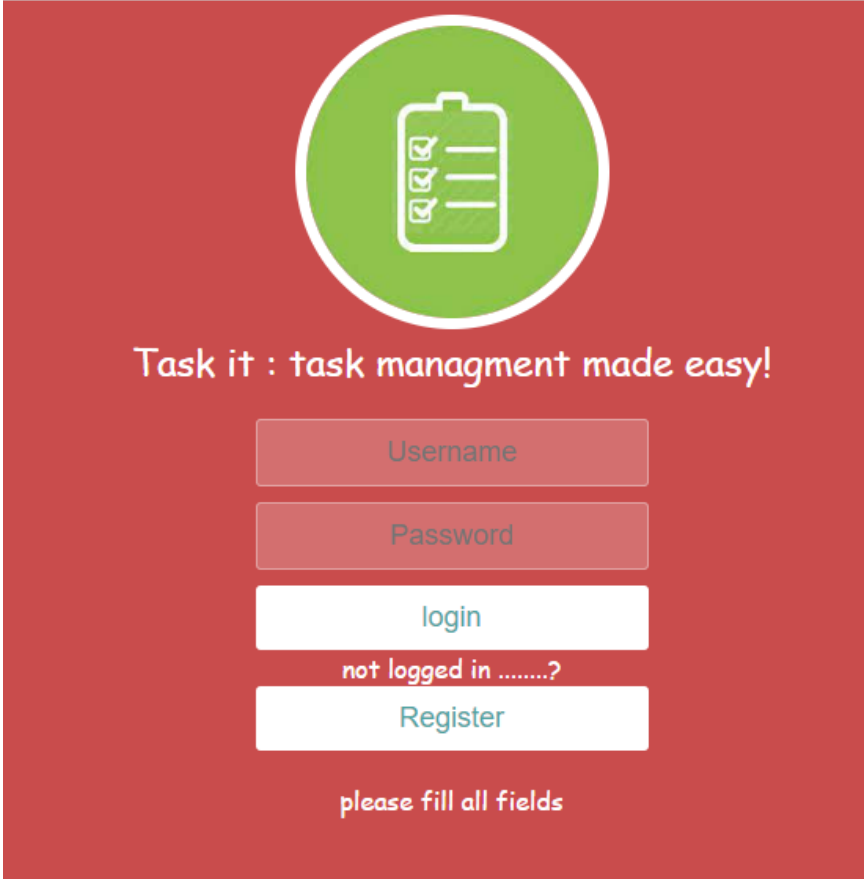
ERD Diagram (Exported from Database Engine)



8. Testing Functionalities

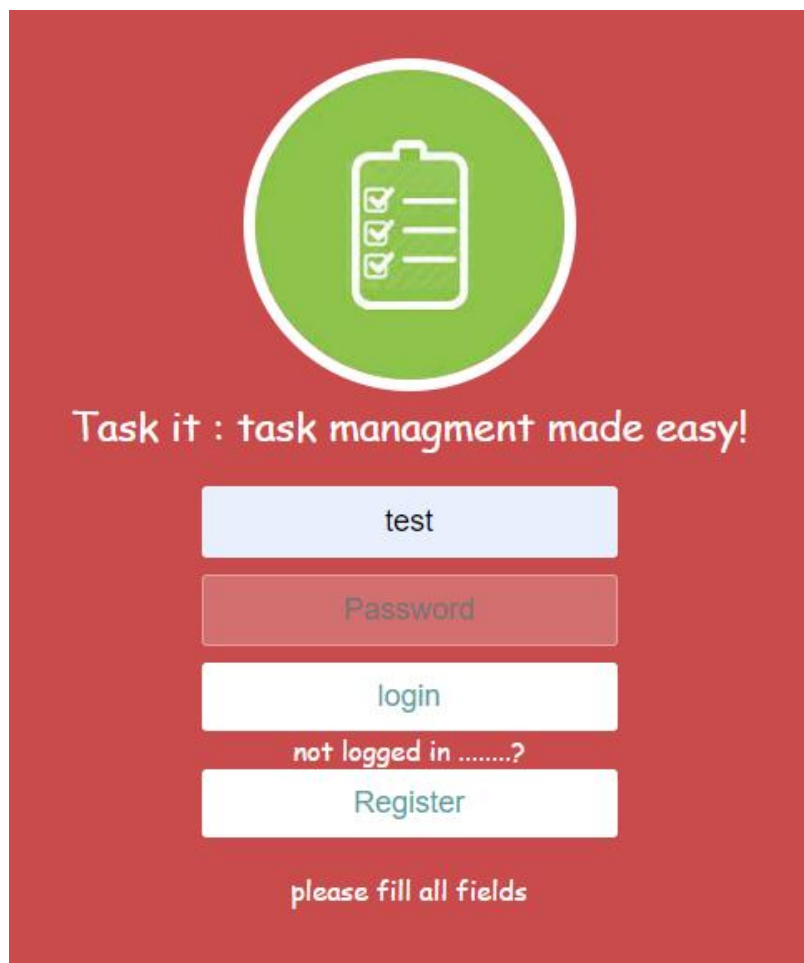
Login Functionality

Requirement Name: Login	
Requirement ID: Test case 001	Results: Cleared Yes
Brief Description: Attempt to login without Entering any of the Required Fields.	
Expected Output: System will produce an error message indicating blank Fields and refuse to login.	
Actual Output: When tried to press login button, System produced message requiring filling blank fields.	



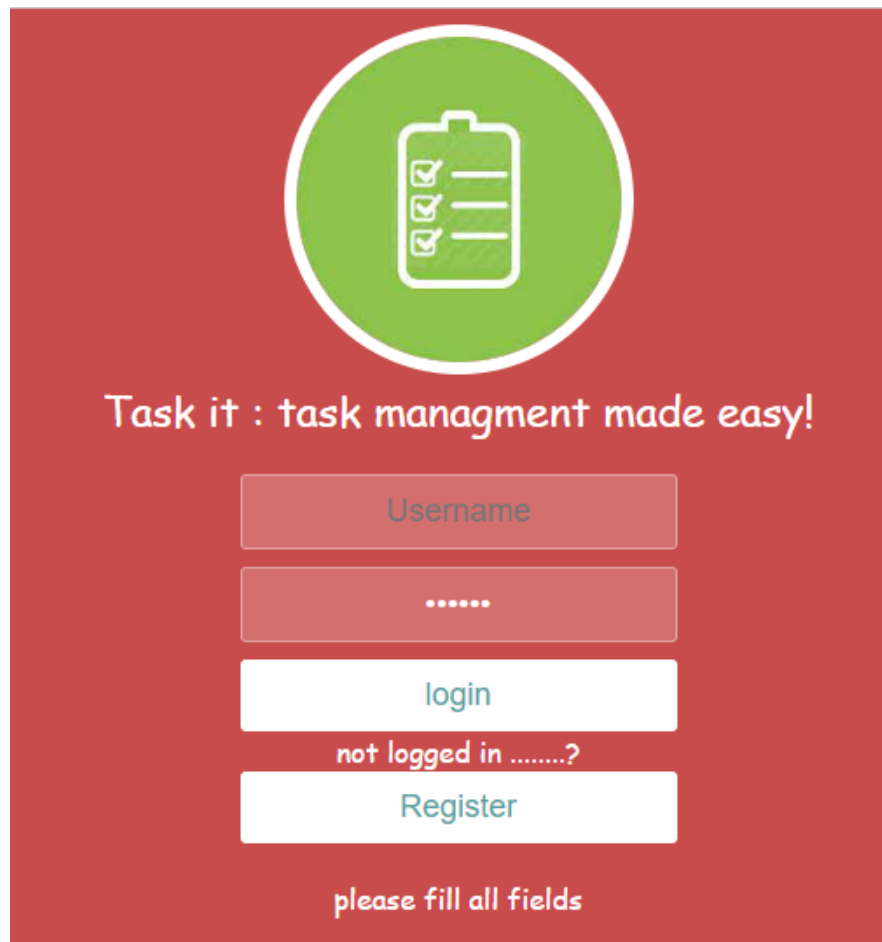
The image shows a mobile application interface for task management. At the top, there is a green circular icon containing a white clipboard with three checkmarks. Below the icon, the text "Task it : task managment made easy!" is displayed in a white, sans-serif font. Underneath the text, there are two input fields: "Username" and "Password", both with a light red background and rounded corners. Below these fields is a white button with the text "login" in blue. To the right of the "login" button, there is a red text message "not logged in?". Below the "login" button is another white button with the text "Register" in blue. At the bottom of the interface, there is a red text message "please fill all fields". The entire interface is set against a solid red background.

Requirement Name: Login	
Requirement ID: Test case 002	Results: Cleared Yes
Brief Description: Attempt to login with entering only username but without password	
Expected Output: System will produce an error message indicating blank Field and refuse to login.	
Actual Output: When tried to press login button with entering only username, System produced message requiring filling blank fields.	



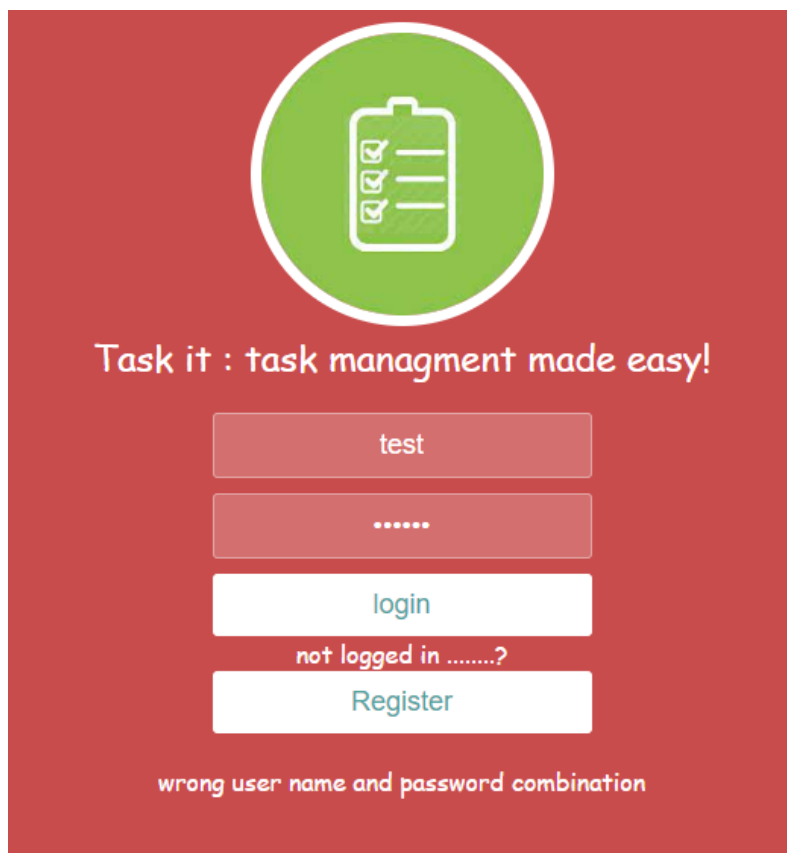
The image shows a mobile application interface for task management. At the top, there is a green circular icon containing a white clipboard with a checklist. Below the icon, the text "Task it : task managment made easy!" is displayed in a white, sans-serif font. The interface features a vertical stack of buttons: a light blue button labeled "test", a red button labeled "Password", a white button labeled "login", and a white button labeled "Register". Below the "login" button, the text "not logged in" is visible in red. At the bottom of the screen, the text "please fill all fields" is displayed in white. The entire interface is set against a solid red background.

Requirement Name: Login	
Requirement ID: Test case 003	Results: Cleared Yes
Brief Description: Attempt to login with entering only password but without username	
Expected Output: System will produce an error message indicating blank Field and refuse to login.	
Actual Output: When tried to press login button with entering only password, System produced message requiring filling blank fields.	



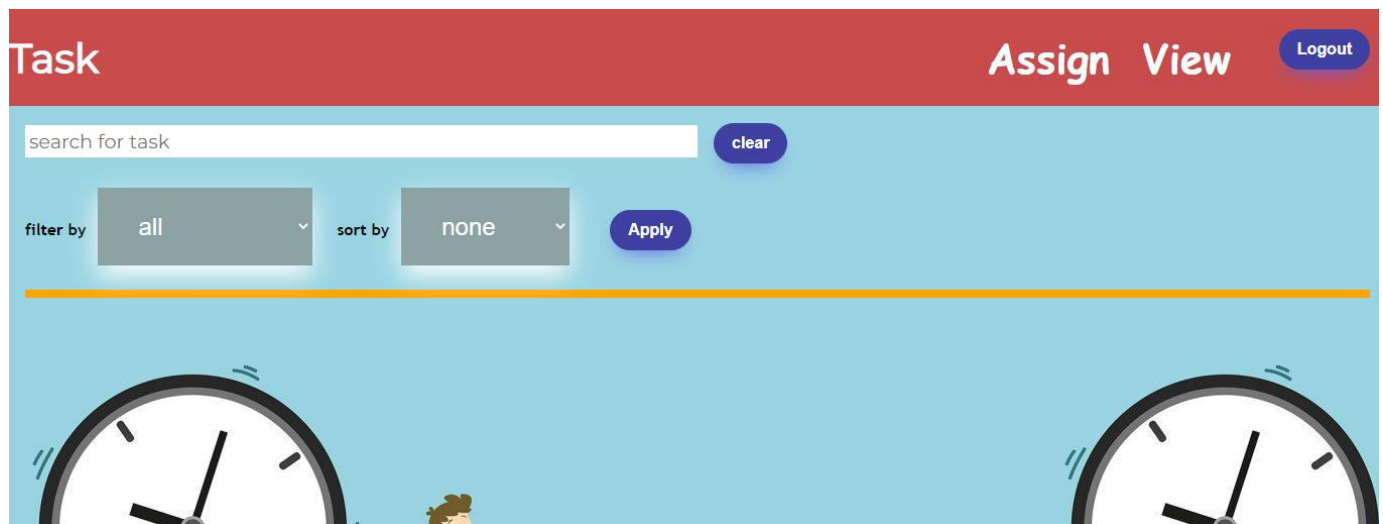
The image shows a mobile application interface for task management. At the top, there is a green circular icon containing a white clipboard with a checklist. Below the icon, the text "Task it : task managment made easy!" is displayed in a light red font. The interface includes two input fields: the first is labeled "Username" and the second contains a series of dots, indicating a password field. Below these fields are three buttons: a white "login" button, a red "not logged in?" button, and a white "Register" button. At the bottom of the screen, the text "please fill all fields" is displayed in a light red font.

Requirement Name: Login	
Requirement ID: Test case 004	Results: Cleared Yes
Brief Description: Attempt to login with entering username that is present inside the DB but with wrong Password.	
Expected Output: System will produce an error message indicating wrong email and password Combination.	
Actual Output: When tried to press login button with entering Username and incorrect password password, System produced message indicating wrong Combination.	

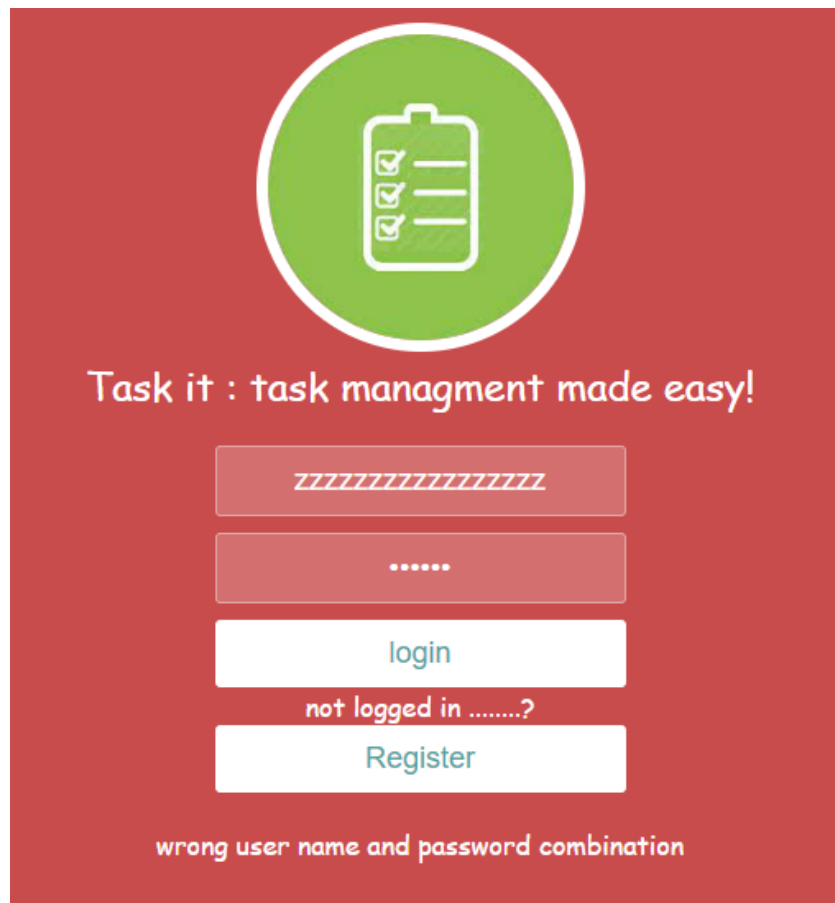


The image shows a login screen for a task management application. At the top, there is a green circular icon with a white clipboard and checklist symbol. Below the icon, the text "Task it : task managment made easy!" is displayed in a light red font. The login form consists of two red input fields: the first contains the text "test" and the second contains six dots ".....". Below these fields is a white "login" button. Under the button, the text "not logged in?" is shown in red. At the bottom of the form is a white "Register" button. Finally, at the very bottom of the screen, the message "wrong user name and password combination" is displayed in red.

Requirement Name: Login	
Requirement ID: Test case 005	Results: Cleared Yes
Brief Description: Attempt to login with entering username that is present inside the DB and Correct Password.	
Expected Output: System will redirect user into homepage.	
Actual Output: User Successfully logged in and redirected into Homepage.	



Requirement Name: Login	
Requirement ID: Test case 006	Results: Not Cleared.
Brief Description: Attempt to login with entering Incorrect username that isn't present inside the DB.	
Expected Output: System will produce an error message indicating absent username	
Actual Output: When tried to press login button with entering username that isn't present inside Db indicated wrong combination.	



The image shows a login screen for a task management application. At the top, there is a green circular icon with a white clipboard and checklist symbol. Below the icon, the text "Task it : task managment made easy!" is displayed in a light red font. The login form consists of two input fields: the first contains a series of red slashes (//////////), and the second contains six red dots (.....). Below these fields is a white "login" button. Underneath the button, the text "not logged in?" is shown in red. Below that is a white "Register" button. At the bottom of the screen, the message "wrong user name and password combination" is displayed in red.

Assigning Task Functionality

Requirement Name: Assigning Task	
Requirement ID: Test case 007	Results: Cleared Yes
Brief Description: Attempt to assign task with entering only task name and other fields are Blank.	
Expected Output: System will produce an error message indicating blank Fields and refuse to add task in the DB.	
Actual Output: When tried to press add button with only filling task title, the system produced an error message and refused to add task inside the DB.	

The screenshot shows a task assignment form with a light blue background. The form has a white rounded rectangle containing the following elements:

- A text input field with the text "sadadas" and a cursor at the end.
- A placeholder text "Task description ..." below the first field.
- A dropdown menu labeled "choose priority" with a downward arrow.
- A date input field with the placeholder "mm/dd/yyyy" and a calendar icon to its right.
- A yellow circular button labeled "Add" at the bottom right of the form.

Below the form, a blue banner contains the text "please fill all fields properly" in white.

Requirement Name: Assigning Task	
Requirement ID: Test case 008	Results: Cleared Yes
Brief Description: Attempt to assign task with entering only task name and task description but other fields are blank.	
Expected Output: System will produce an error message indicating blank Fields and refuse to add task in the DB.	
Actual Output: When tried to press add button with only filling task title and task description, the system produced an error message and refused to add task inside the DB.	

The screenshot shows a task assignment form on a light blue background. The form has a white background and contains the following elements:

- Two text input fields, both containing the text "ewrwer". The second field has a red dashed underline.
- A dropdown menu labeled "choose priority" with a downward arrow.
- A date input field labeled "mm/dd/yyyy" with a calendar icon to its right.
- A yellow circular button labeled "Add" with a white border.
- A blue error message at the bottom: "please fill all fields properly".

Requirement Name: Assigning Task	
Requirement ID: Test case 009	Results: Cleared Yes
Brief Description: Attempt to assign task with entering only task name, task description and with priority Chosen but other fields are blank.	
Expected Output: System will produce an error message indicating blank Fields and refuse to add task in the DB.	
Actual Output: When tried to press add button with only filling task title, task description, and priority is chosen the system produced an error message and refused to add task inside the DB.	

The screenshot shows a task assignment form on a light blue background. The form is a white rounded rectangle containing the following elements:

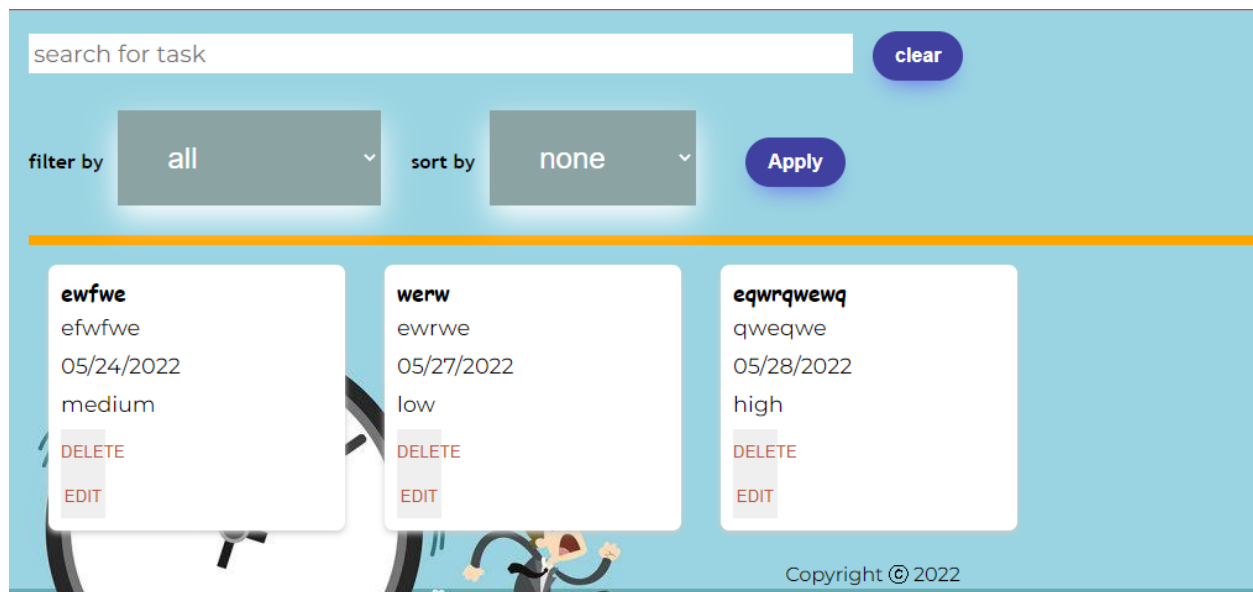
- Two text input fields at the top, each containing placeholder text "weqwqe".
- A dropdown menu below the inputs, showing "high" and a downward arrow.
- A date input field at the bottom left with the placeholder "mm/dd/yyyy".
- A yellow circular "Add" button with a calendar icon at the bottom right.

Below the form, centered on the light blue background, is the error message "please fill all fields properly" in a blue font.

Requirement Name: Assigning Task	
Requirement ID: Test case 010	Results: Cleared Yes
Brief Description: Attempt to assign task with entering only task name, task description, with priority Chosen and correct date (\geq today's Date).	
Expected Output: System will accept the task and successfully inserts it.	
Actual Output: When tried to press add button with filling all fields correctly inserted, the system inserted the task inside the DB and can be viewed in the view page.	

The screenshot shows a task assignment form with the following fields and values:

- Task Name: eqwrqwewq
- Task Description: qweqwe (underlined in red)
- Priority: high (selected from a dropdown menu)
- Date: 05/28/2022
- Action: Add (yellow button)



Requirement Name: Assigning Task	
Requirement ID: Test case 011	Results: Cleared Yes
Brief Description: Attempt to assign task with entering only task name, task description, with priority Chosen and Incorrect date (<today's Date).	
Expected Output: System will accept the task and successfully inserts it.	
Actual Output: When tried to press add button with filling all fields correctly inserted except the date, the system produced an error asking to correct insert data.	

qweqw

qweqwe


low

05/01/2022

Add

you cant enter a task with a date earlier than todays date

Requirement Name: Register	
Requirement ID: Test case 012	Results: Cleared Yes
Brief Description: Attempt to register without entering any of the required Fields.	
Expected Output: System will produce an error message indicating Blanks fields.	
Actual Output: When tried to register with leaving all fields blank, The System produced an error message.	



Enter Password and Email

Username

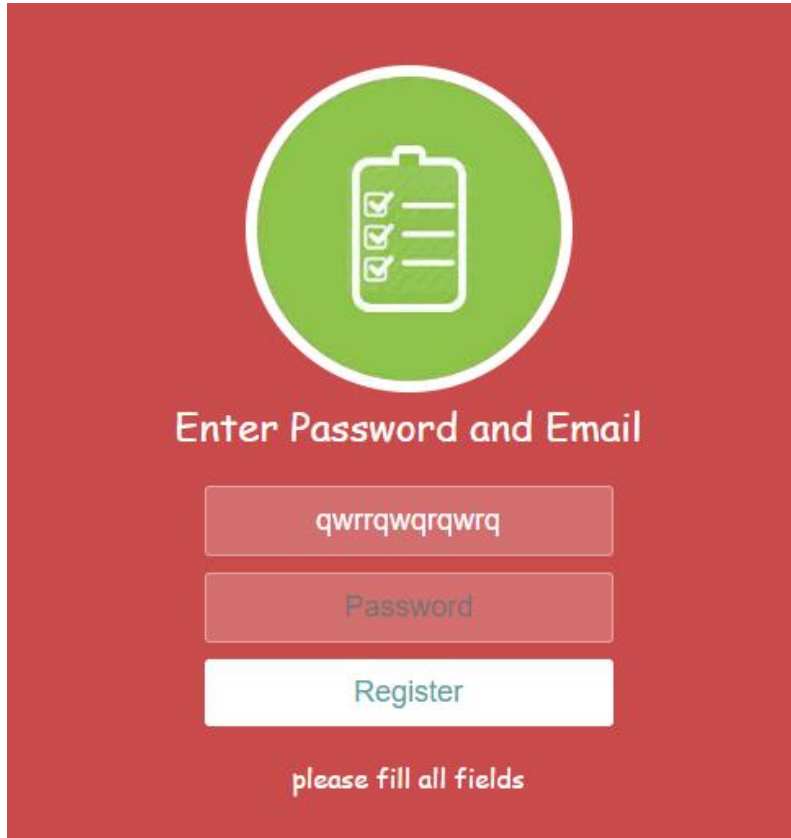
Password

Register

please fill all fields

Requirement Name: Register	
Requirement ID: Test case 013	Results: Cleared Yes
Brief Description: Attempt to register with entering only username with other fields blank.	
Expected Output: System will produce an error message indicating Blanks fields.	

Actual Output: When tried to register with entering only username, The System produced an error message.

A screenshot of a registration form on a red background. At the top is a green circular icon containing a white clipboard with three checkmarks. Below the icon, the text "Enter Password and Email" is displayed in white. There are two input fields: the first contains the text "qwrrqwqrqwrq" and the second is labeled "Password". Below these fields is a white "Register" button. At the bottom of the form, the text "please fill all fields" is written in white.

Enter Password and Email

qwrrqwqrqwrq

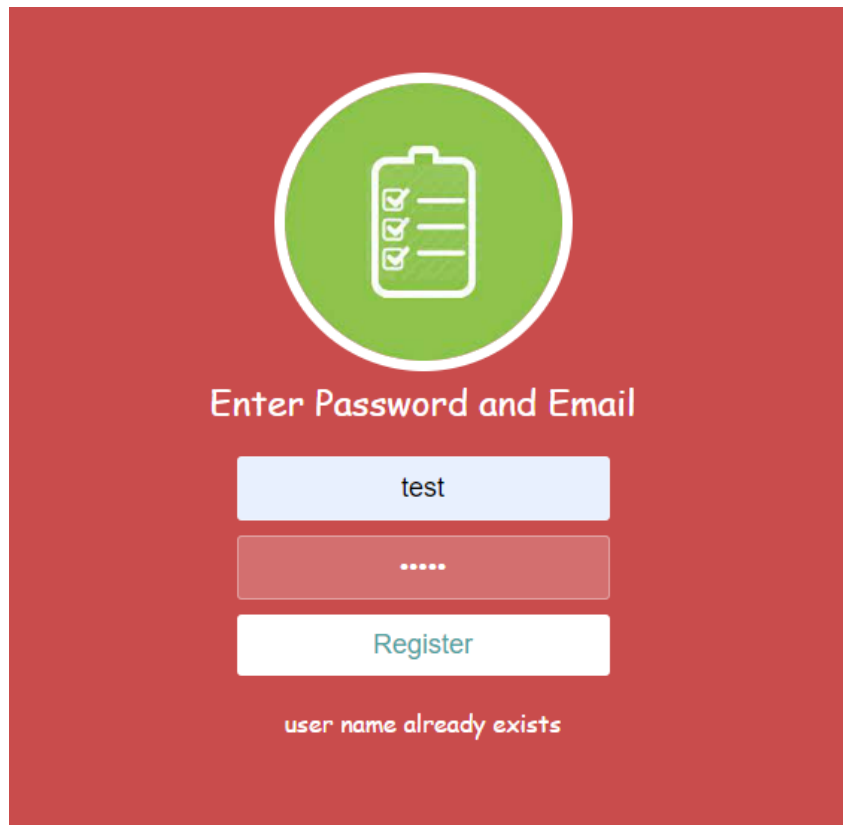
Password

Register

please fill all fields

Requirement Name: Register	
Requirement ID: Test case 014	Results: Cleared Yes
Brief Description: Attempt to register with username already inside the DB.	
Expected Output: System will produce an error message indicating username already inside DB	

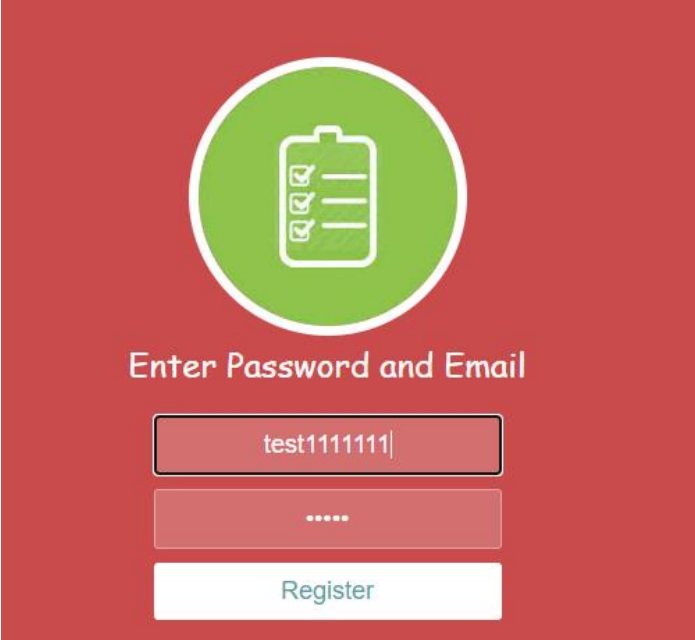
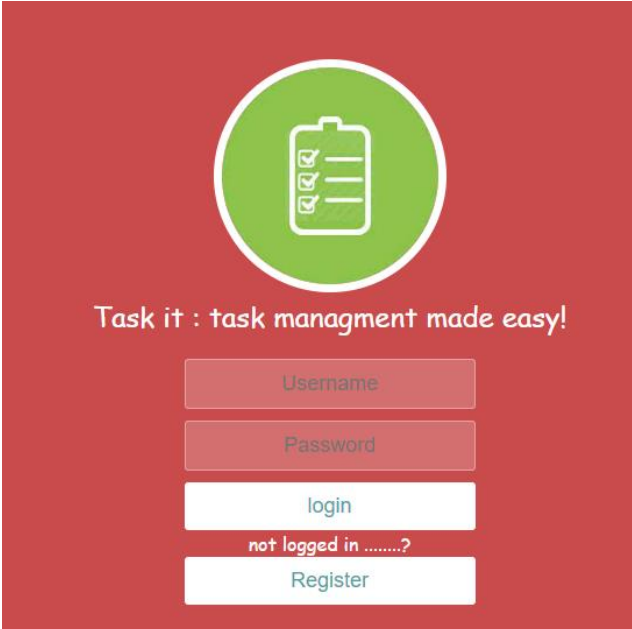
Actual Output: When tried to register with entering username that inside the DB, The system produced an error.



The image shows a registration form on a red background. At the top is a green circular icon with a white clipboard and checklist. Below it, the text "Enter Password and Email" is displayed. There are three input fields: a light blue field containing "test", a red field with masked characters ".....", and a white field with the text "Register" in blue. At the bottom, the error message "user name already exists" is shown in red.

Requirement Name: Register	
Requirement ID: Test case 015	Results: Cleared Yes
Brief Description: Attempt to register with valid username (is not already registered) and password	
Expected Output: System will redirect into login page.	

Actual Output: The system redirected to login page.

A screenshot of a registration form on a red background. At the top is a green circular icon containing a white clipboard with a checklist. Below the icon, the text "Enter Password and Email" is displayed in white. There are two input fields: the first contains the text "test1111111|" and the second contains six dots. Below these fields is a white button with the text "Register" in blue.A screenshot of a login form on a red background. At the top is a green circular icon containing a white clipboard with a checklist. Below the icon, the text "Task it : task managment made easy!" is displayed in white. There are two input fields labeled "Username" and "Password" in light gray. Below these fields is a white button with the text "login" in blue. Below the "login" button is the text "not logged in?" in red, followed by a white button with the text "Register" in blue.

Requirement Name: Register

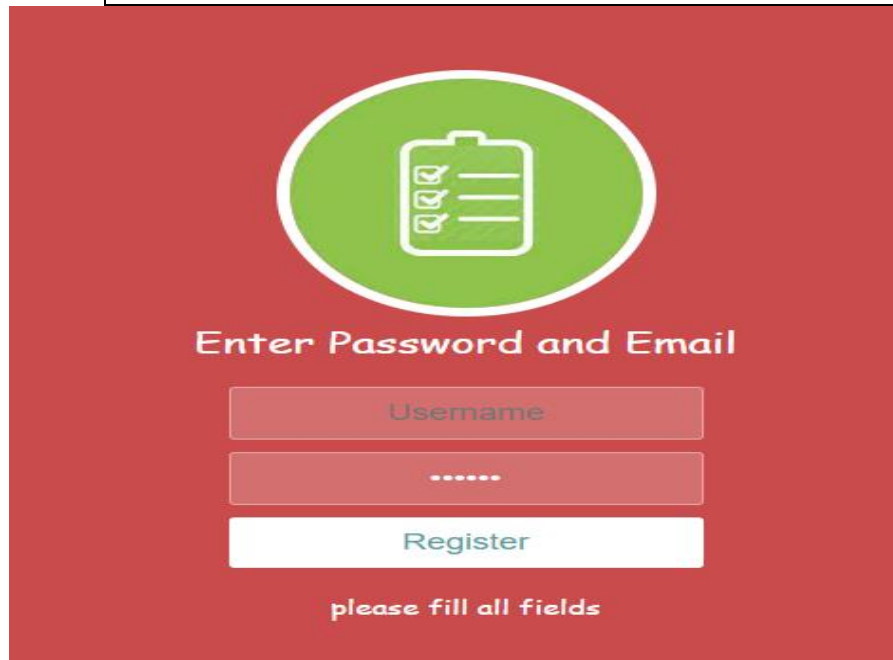
Requirement ID: Test case 016

Results: Cleared Yes

Brief Description: Attempt to register with entering password only without username.

Expected Output: System will produce an error message indicating blank fields.

Actual Output: The system Produced an error.

A screenshot of a registration form on a red background. At the top is a green oval icon containing a white clipboard with a checklist. Below the icon, the text "Enter Password and Email" is displayed in white. There are two input fields: the first is labeled "Username" and the second contains six dots, representing a password field. Below these fields is a white "Register" button. At the bottom of the form, the text "please fill all fields" is written in white.

Requirement Name: Login, Assigning Tasks, Viewing Tasks (Integration)	
------------------------------------------------------------------------------	--

Requirement ID: Test case 017	Results: Cleared Yes
--------------------------------------	-----------------------------

Brief Description: Attempt to register then login then add a task and view the task.	
---------------------------------------------------------------------------------------------	--

Expected Output: Correct integration (user credentials inserted inside the DB successfully retrieved successfully and logged in, task is added and correctly retrieved).	
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Actual Output: No issues when performing all tasks described.	
----------------------------------------------------------------------	--

Requirement Name: Login, Assigning Tasks, Viewing Tasks then edit and delete (Integration)	
Requirement ID: Test case 018	Results: Cleared Yes
Brief Description: Attempt login then add a task and view the task. Then edit it and delete it.	
Expected Output: Correct integration (user credentials inside the DB successfully retrieved and logged in, task is added and correctly retrieved then edited successfully and deleted).	
Actual Output: No issues when performing all tasks described.	