Group 16

Interaction and High level Design for the System

Author: Richard Price-Jones, Archie Strange, Greg Sharpe, Rhodri Pearce, Emil Ramsdal

Config Ref: SE_16_DS_01

Date: 27/10/15

Version: 1.1

Status: Released

Department of Computer
Science Aberystwyth
University Aberystwyth
Ceredigion SY23 3DB

Contents

1.0 Introduction

1.1 Purpose of this document

The purpose of this document is to describe and illustrate the specification for the design of our system. It contains all the descriptions that will be necessary for the implementation phase of the project.

1.2 Scope

This document specifies the high level design for our system. It includes designs for interfaces, software structure, components and data. It describes how our applications will look for the user and how they will interact with the each other.

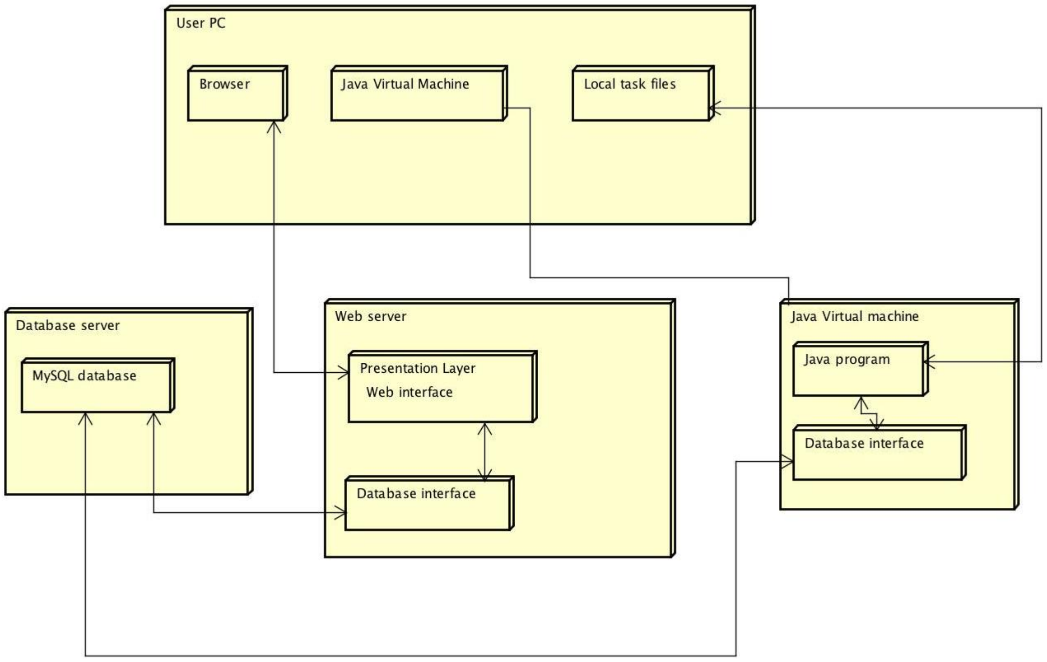
1.3 Objectives

The objective of this document is to provide a framework design that will be used throughout the entire project. It will ensure that the applications have the all the functionality requirements set out by the client. It will also be crucial during the implementation phase as it shows how everything should work and it will be used as a plan.

2.0 Deployment Description

2.1 Applications in the system

The deployment of this software requires the user's computer system to have a modern web browser and the installation of the Java virtual machine. A Web server is required to support PHP which is used to communicate with the database. The database server is required to be running MySQL, the database is also required to allow remote login.



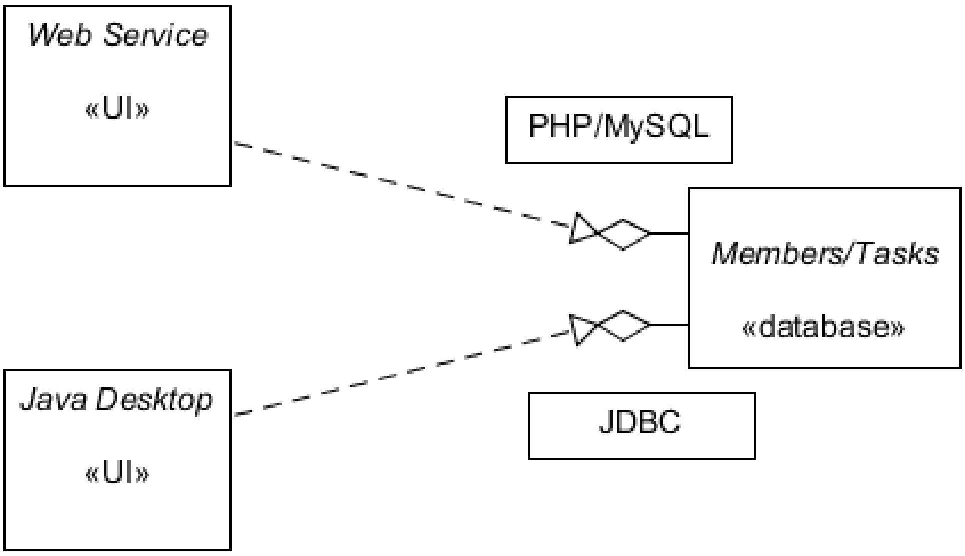
2.2 Applications Interactions

The Web service will use PHP on the server to connect to the database, firstly we must establish a connection within the PHP script. To connect to the database using MySQL we must first use the function mysql_connect, along with the username, password and hostname of the database. Then once connected we may begin to run queries, the function used to perform these queries is named mysql_query(). Lastly, we will need to close the connection, although this isn't necessary as PHP automatically closes the connection when the script ends. By using the mysql_close() function we will close the connection.

JDBC is a driver which allows the Java Application connect to a data source, in our case it's a MySQL database. It will allow the Java Application to send and update query statements and process the results. JDBC will access our remote server using the Internet's file addressing scheme and a file name our on server, which in our case will be our database name.

The communication protocol we will be using within both JDBC and PHP is HTTP. HTTP meaning Hyper Text Transfer Protocol will enable the PHP script and the JDBC to communicate with the database by firstly declaring a port on which our database will be accessible. In this project will we have a connecting PHP page which will include the database address, username and password. This PHP page will be called many times.

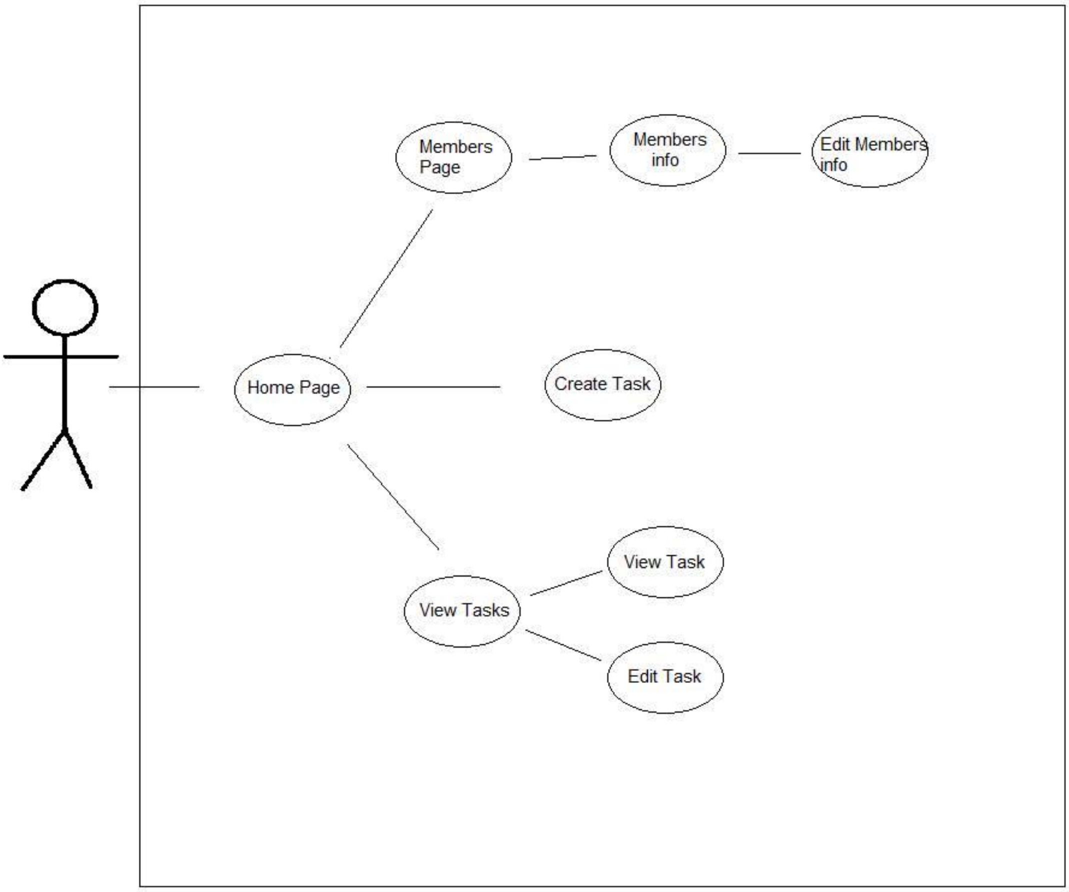
Below is a component diagram illustrating everything that is mentioned above.



3.0 Interaction Design

3.1 Use Case Diagrams

Below is the Use Case Diagram for the Web Application Tasker Man.



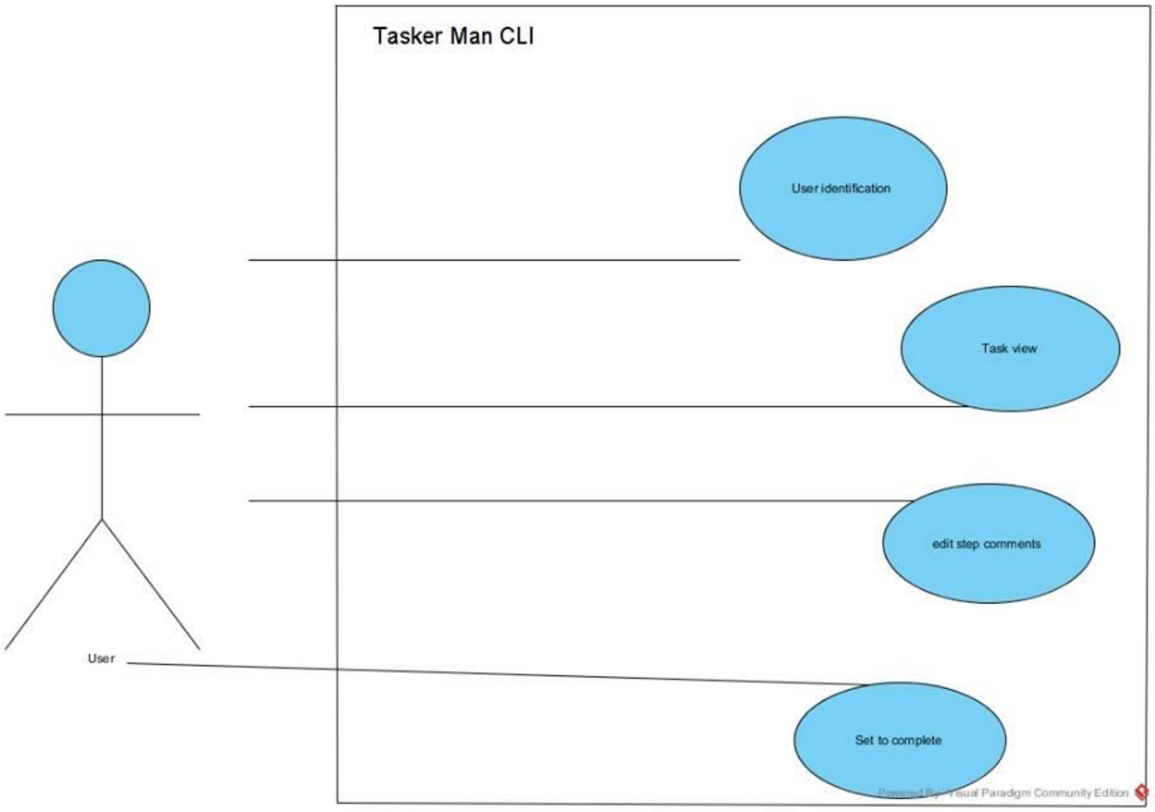
The name of the system (TaskerMAN), is displayed on the top left within the System boundary.The Actor in this case is the Manager of the project, which can be seen to the left of the use case, just outside the system boundary.

The system boundary contains all of the functionality or services provided by the system.

Associations within the use case diagram are used to display the connections between the pages of the website. From this we can see that the user starts at the home page, from there can visit any one of three separate pages (Members, Create task and View Task). From these sections, the user can travel further into the website.

- Through Members, the user can reach ‘Members Info’, and from there, ‘Edit Members Info’.
- From the View Tasks page, the user can then visit one of two options. ‘View Task’ and ‘Edit Task’.

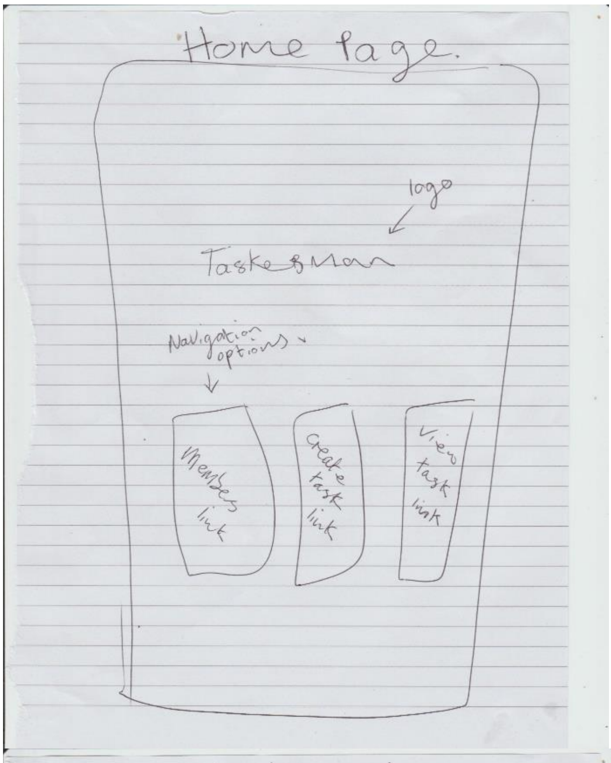
Below is the use case diagram for the user application Tasker CLI.

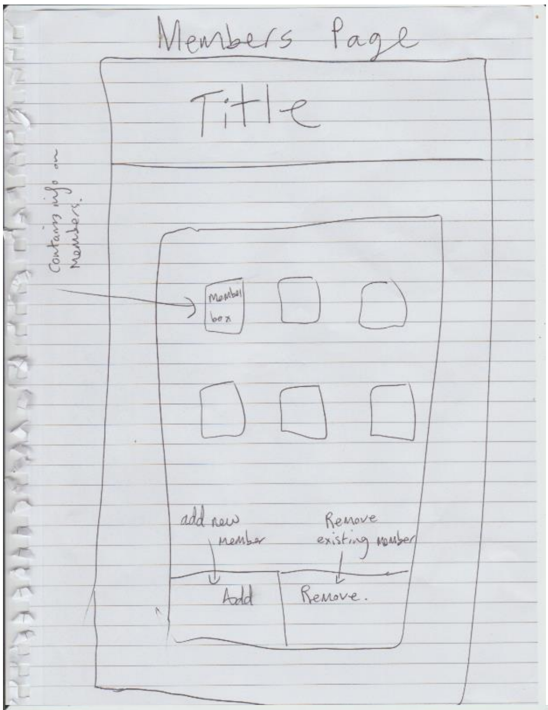


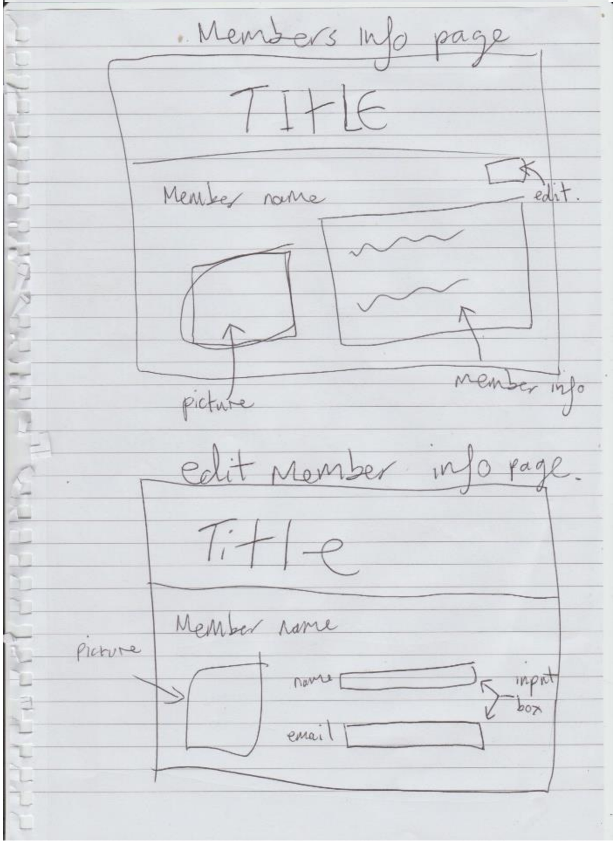
The associations are used in the use case diagram are used to display how the use interacts with the application. You can view all the functionality of the program. So we can see that the application will provides: user identification, the ability to view tasks, to edit the comment about the tasks and finally to able to set the user's assigned task to completed.

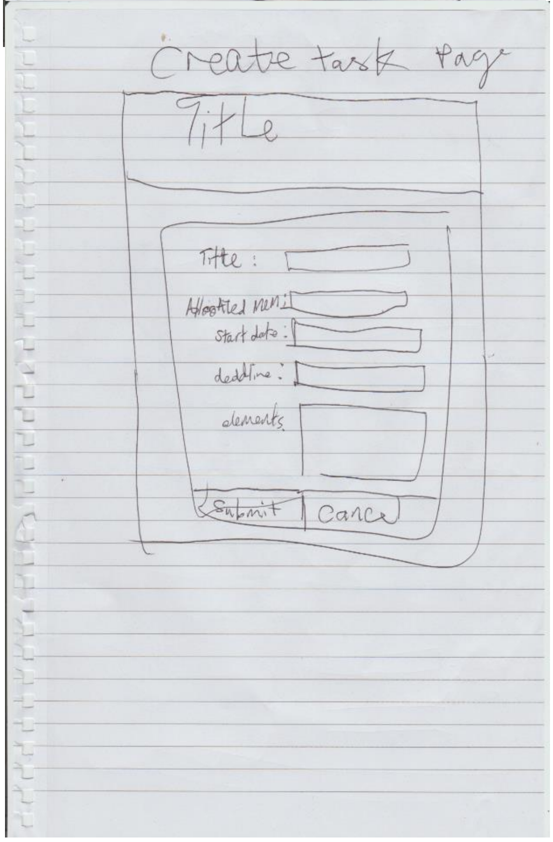
3.2 User Interface Design

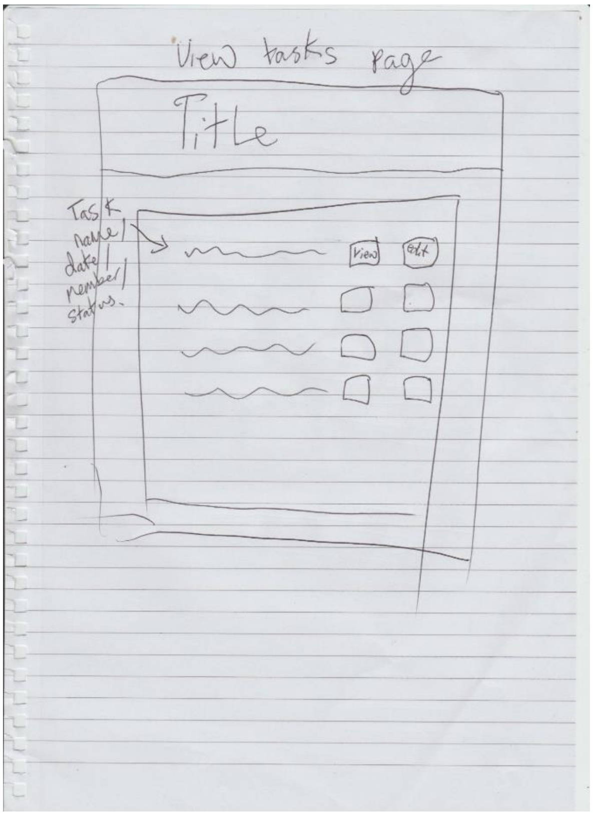
Below are the rough designs drawn up before the final design for the web application Tasker Man.

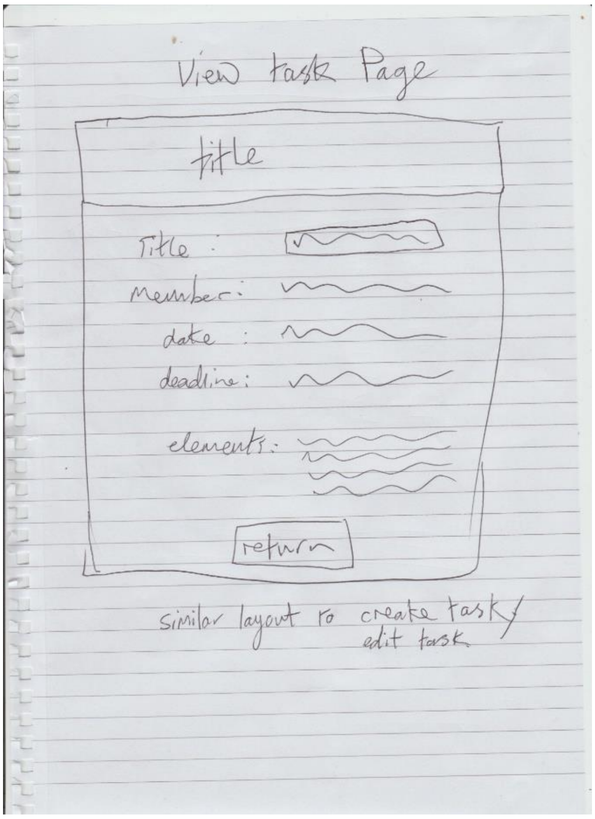




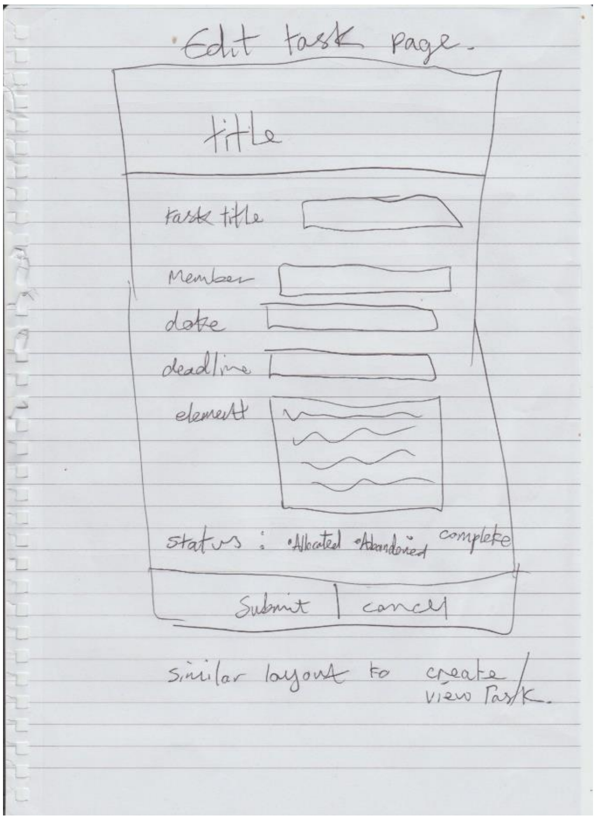








similar layout to create task/ edit task.



similar layout to create/ view Task.

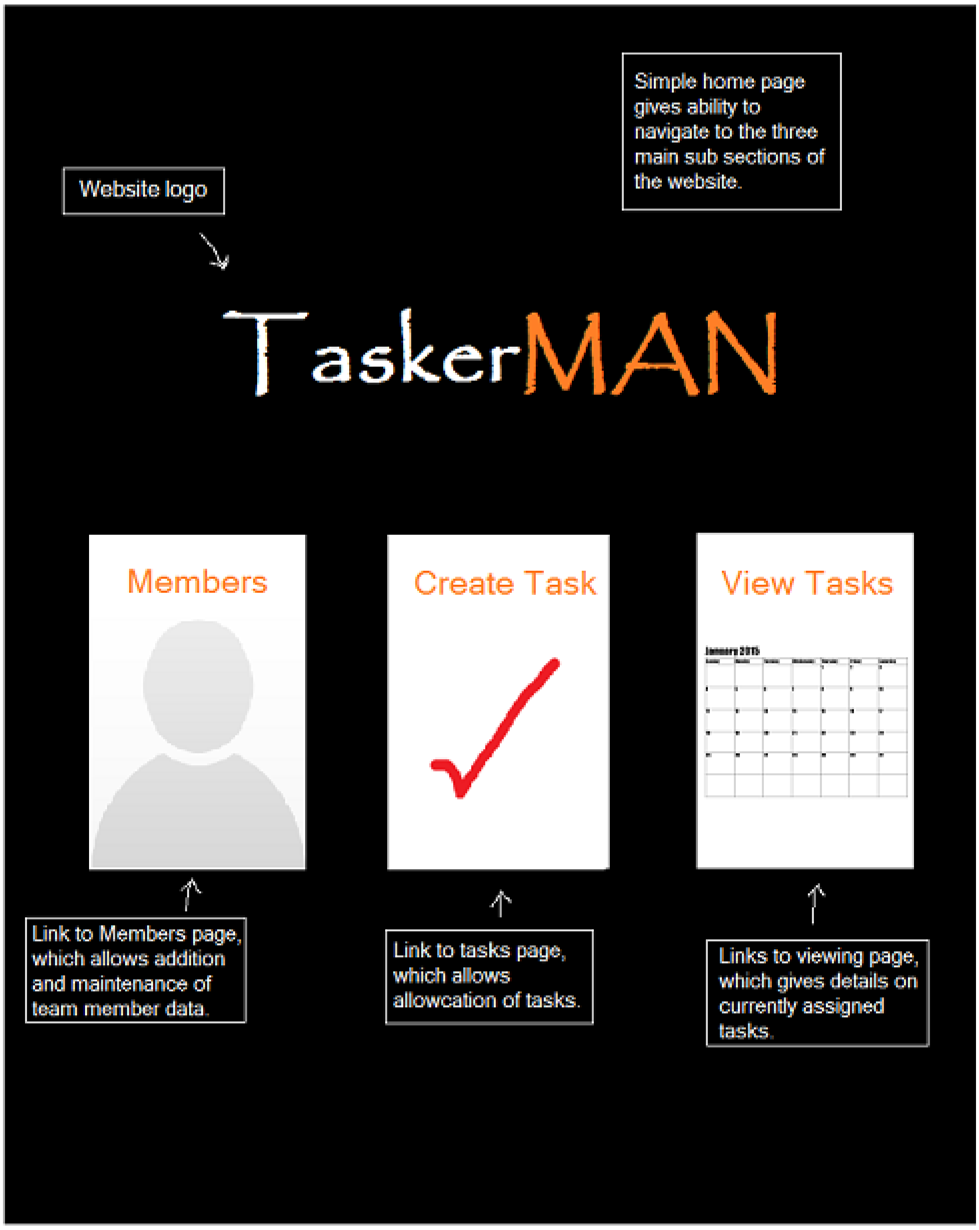
Below are the final designs for the user interface with descriptions.

3.2.1 Template

The header of the website will not change throughout the site, and can be found on every page, with the exception of the home page where is does not exist. This header will give easy access to all of the main sub sections of the site, and also the home page. The navigation is clearly defined, while clicking on the website logo will redirect the user back to the home page.

3.2.2 Home Page

For the home page I've designed a simple page that gives the user the choice to navigate to any of the three main sub sections of the website. Clicking on any of the three boxes will redirect the user to the selected page.



Simple home page gives ability to navigate to the three main sub sections of the website.

Link to Members page, which allows addition and maintenance of team member data.

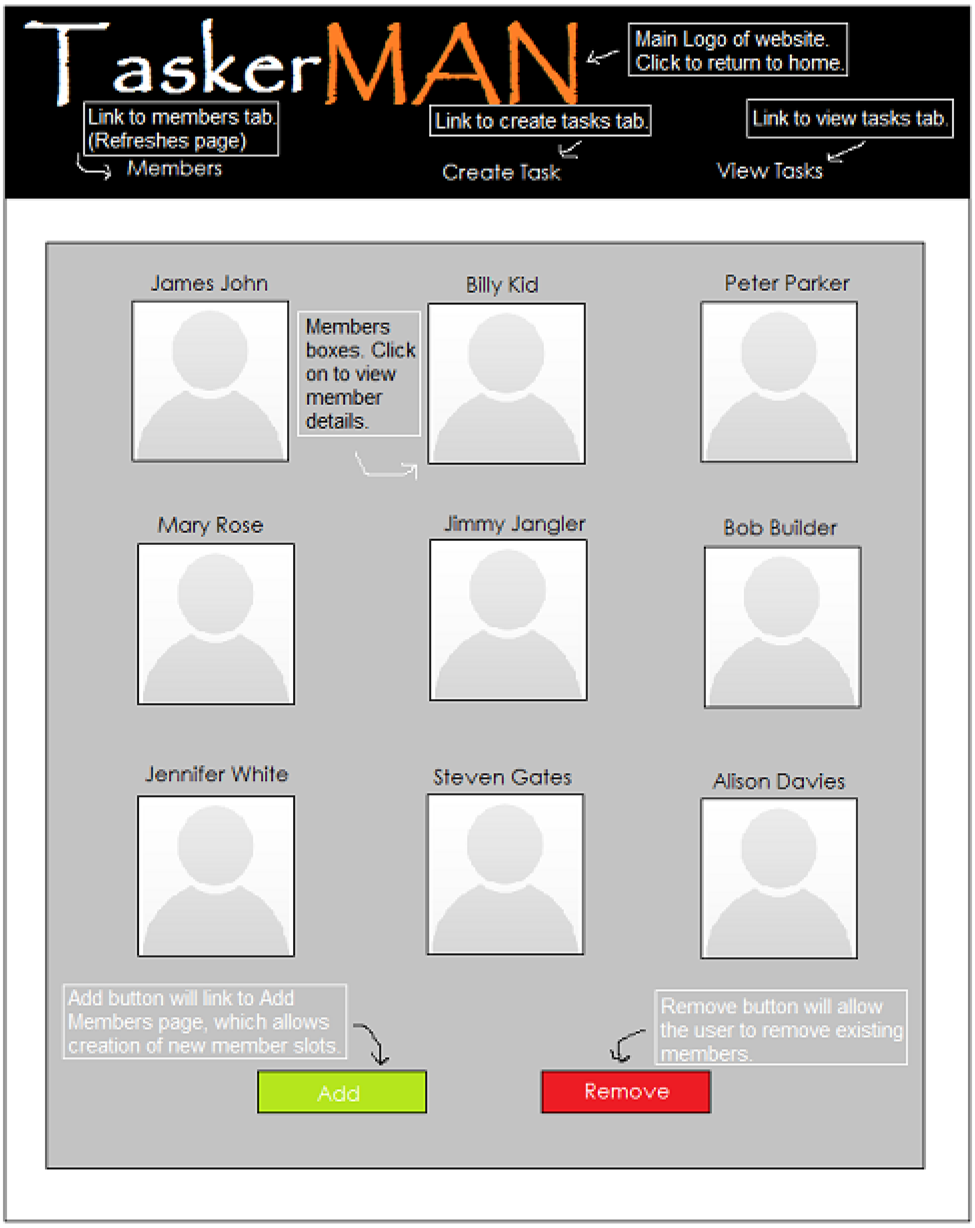
Link to tasks page, which allows allowcation of tasks.

Links to viewing page, which gives details on currently assigned tasks.

3.2.3 Members Page

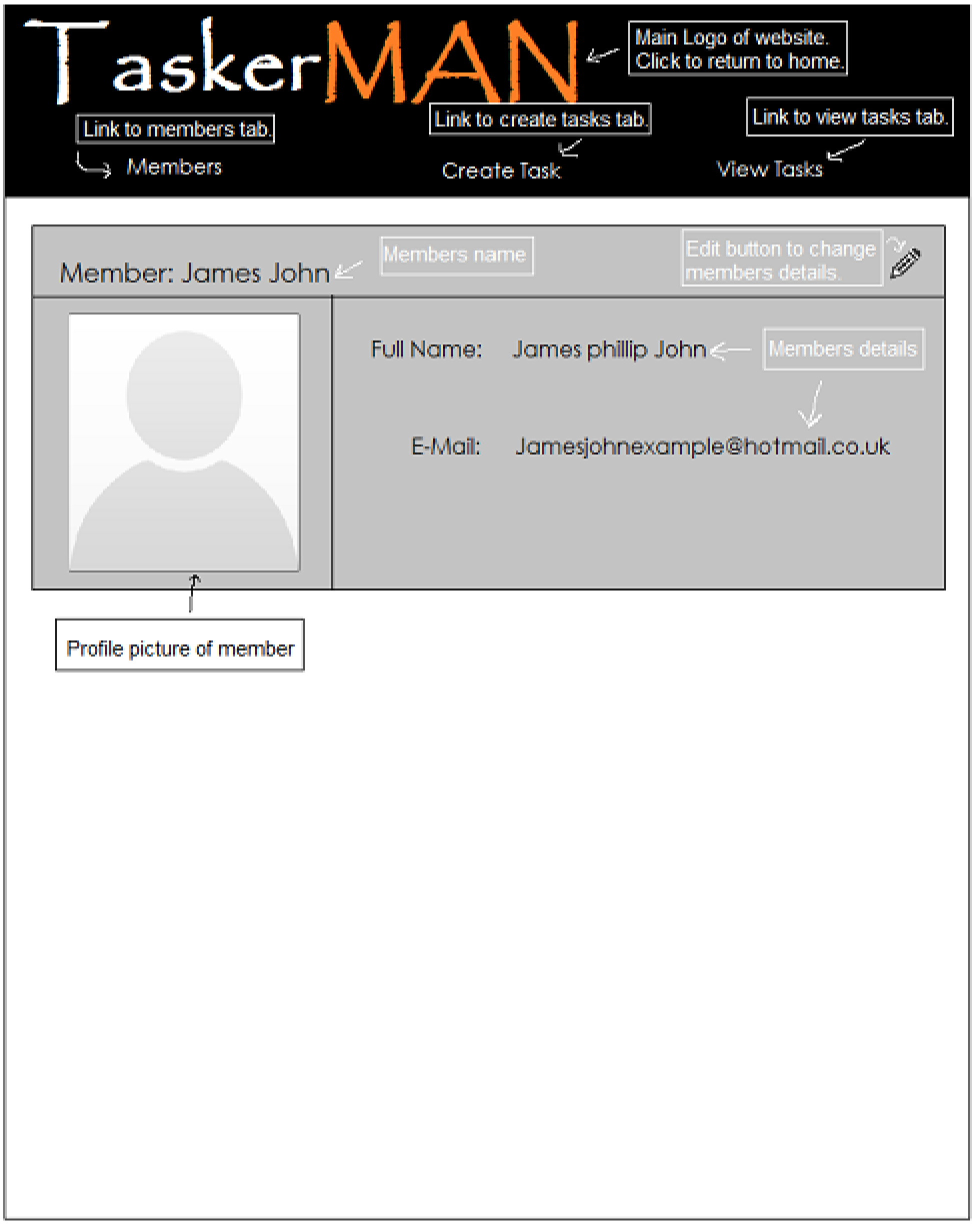
The member's page contains all of the existing members. By clicking on any of the members portraits, the user will be redirected to the member's information page, which holds information on the selected user.

On this page there is the option to add of remove members. Clicking the 'Add' button will redirect to the 'Add Members Page', which gives the user the ability to create a new user with personalised information, while clicking on the 'Remove' button will give the user the ability to then click on the desired profile for deletion, without the hassle of moving to another page.


TaskerMAN
Main Logo of website. Click to return to home.
Link to members tab. (Refreshes page)
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks
James John
Members boxes. Click on to view member details.
Billy Kid
Peter Parker
Mary Rose
Jimmy Jangler
Bob Builder
Jennifer White
Steven Gates
Alison Davies
Add button will link to Add Members page, which allows creation of new member slots.
Remove button will allow the user to remove existing members.
Add
Remove

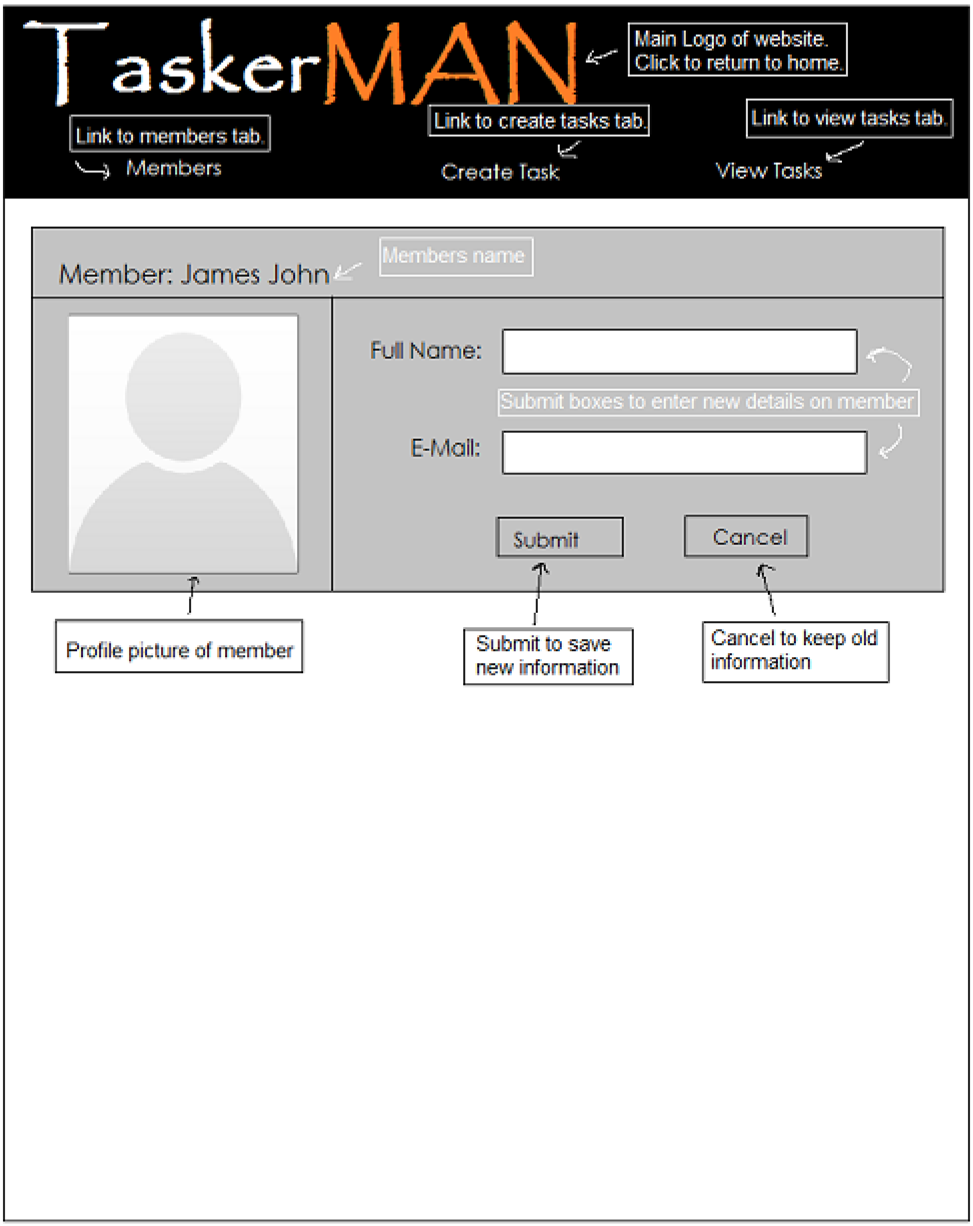
3.2.4 Members Information Page

The information page contains the required details of the member. The edit button on the top right of the users information is a link to the 'Edit Members Information Page', which gives the user the ability to the existing information on the member.


TaskerMAN
Main Logo of website. Click to return to home.
Link to members tab.
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks
Member: James John
Members name
Edit button to change members details.
Full Name:
James phillip John
Members details
E-Mail:
Jamesjohnexample@hotmail.co.uk
Profile picture of member

3.2.5 Edit Members Information Page

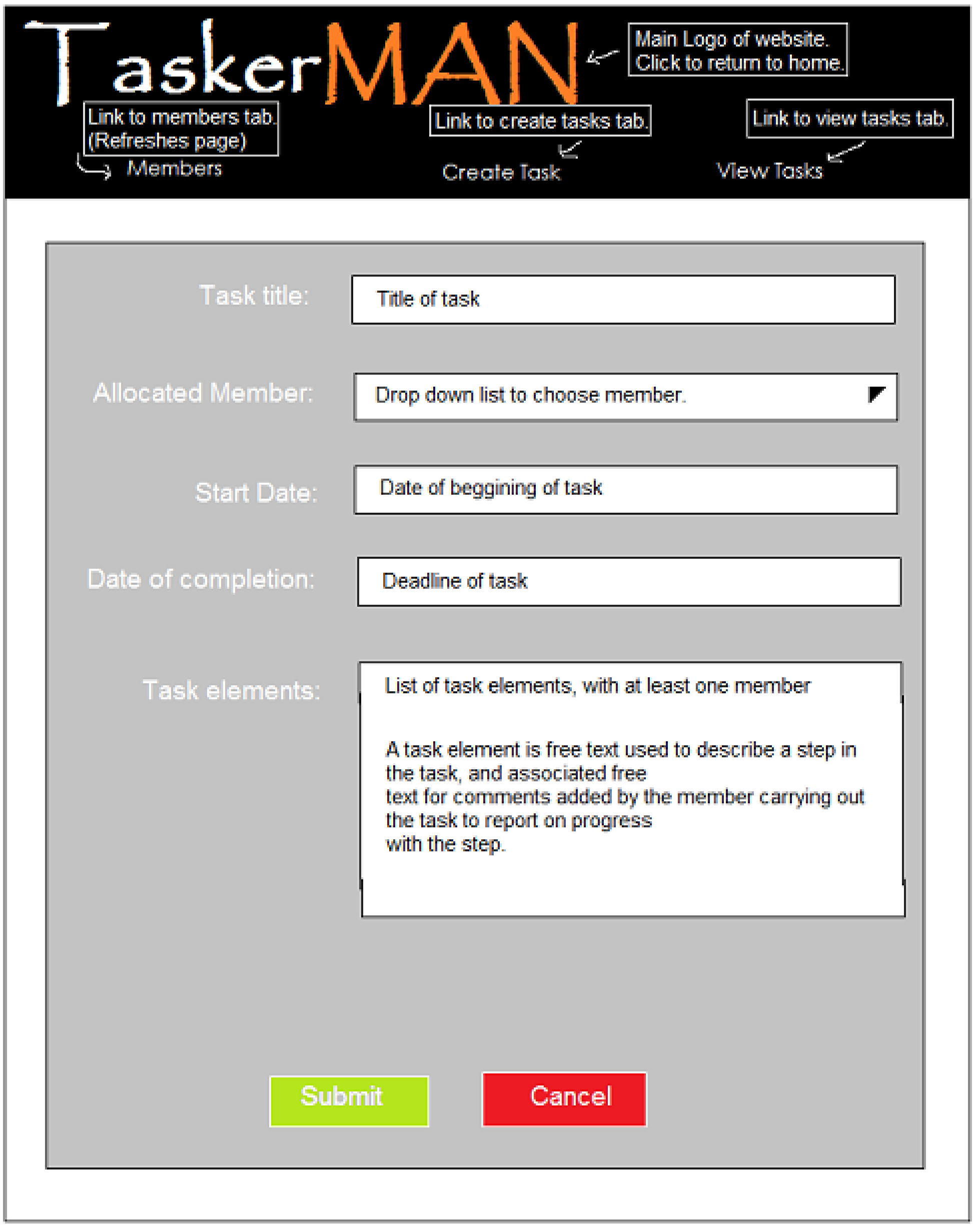
The edit information page is used to change the member's current details, which can be saved by submitting, or restored to its original state prior to editing by pressing the Cancel button.


TaskerMAN
Main Logo of website. Click to return to home.
Link to members tab.
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks
Member: James John
Members name
Full Name:
Submit boxes to enter new details on member
E-Mail:
Submit
Cancel
Profile picture of member
Submit to save new information
Cancel to keep old information

3.2.6 Create Task Page

The create task page allows the user to create tasks.

To create a task the site requires information to be submitted by the user: The member to which it is being assigned and various task details (Title of the task, Starting date, Deadline, Task elements). On clicking the drop down list, all available members will be displayed ready for the user to choose the desired member. On submission the task will automatically be set as "Active". Any mistakes made whilst creating the task can be altered through the 'Edit Task Page'.


TaskerMAN
Main Logo of website. Click to return to home.
Link to members tab. (Refreshes page)
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks

Task title: Title of task

Allocated Member: Drop down list to choose member.

Start Date: Date of beggining of task

Date of completion: Deadline of task

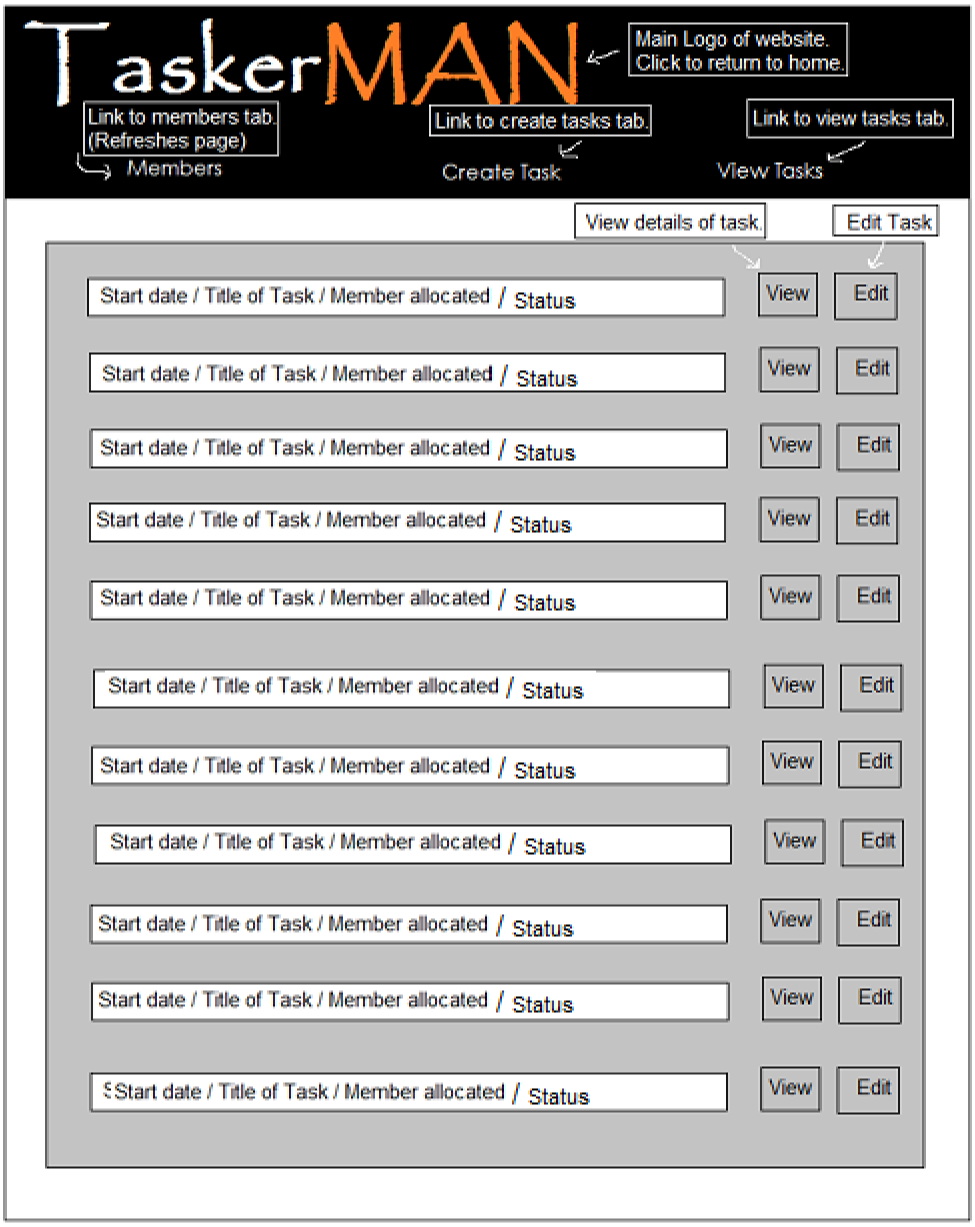
Task elements: List of task elements, with at least one member

A task element is free text used to describe a step in the task, and associated free
text for comments added by the member carrying out the task to report on progress
with the step.

3.2.7 View Tasks Page

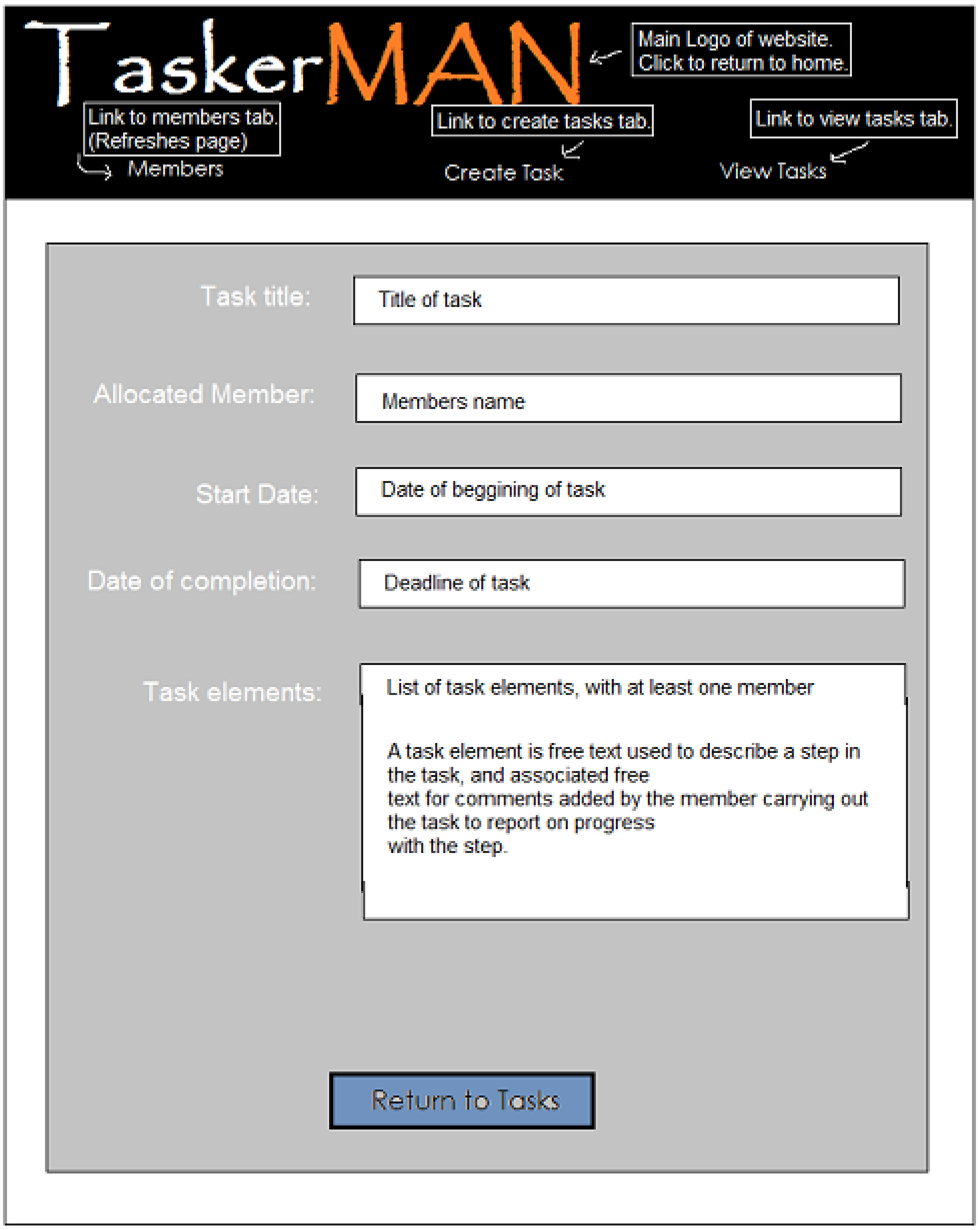
The view tasks page displays all the tasks that have been assigned, whether they be active, abandoned or complete. To view these tasks in greater detail simply click the view button adjacent to the description. The Edit button will redirect the user to the Edit Task page.

This list should be sorted by expected completion date. It should be possible to filter this list by task status and/or allocated team member.


TaskerMAN
Main Logo of website.
Click to return to home.
Link to members tab.
(Refreshes page)
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks
View details of task.
Edit Task
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit
Start date / Title of Task / Member allocated / Status
View
Edit

3.2.8 View Task Page

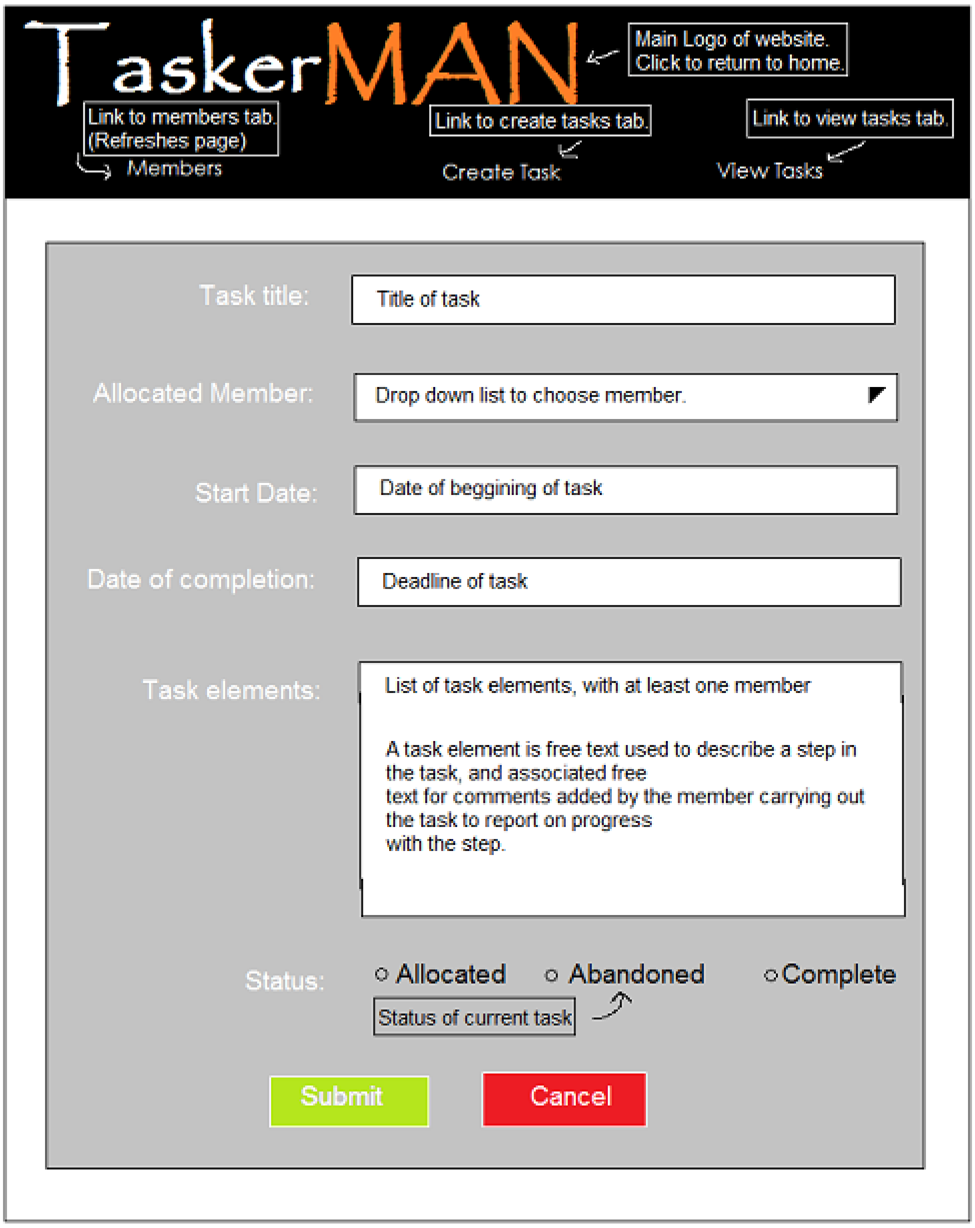
The view task page simply allows the user to view a task in greater detail. Once the user is ready to leave the page, click the "Return to Tasks" button to return to the 'View Tasks' page.



TaskerMAN
Main Logo of website.
Click to return to home.
Link to members tab.
(Refreshes page)
Members
Link to create tasks tab.
Create Task
Link to view tasks tab.
View Tasks
Task title:
Title of task
Allocated Member:
Members name
Start Date:
Date of beggining of task
Date of completion:
Deadline of task
Task elements:
List of task elements, with at least one member
A task element is free text used to describe a step in the task, and associated free text for comments added by the member carrying out the task to report on progress with the step.
Return to Tasks

3.2.9 Edit Task Page

The edit task page will give all the same options as was had when creating the task originally, with the addition allowing the user to edit the tasks current status. From here the user will also be able to change the member to whom the task is allocated. The 'Submit' button will save any changes made to the task, while the 'Cancel' button will restore the original information prior to editing.



Main Logo of website. Click to return to home.

Link to members tab. (Refreshes page)

Link to create tasks tab.

Link to view tasks tab.

Title of task

Drop down list to choose member.

Date of beggining of task

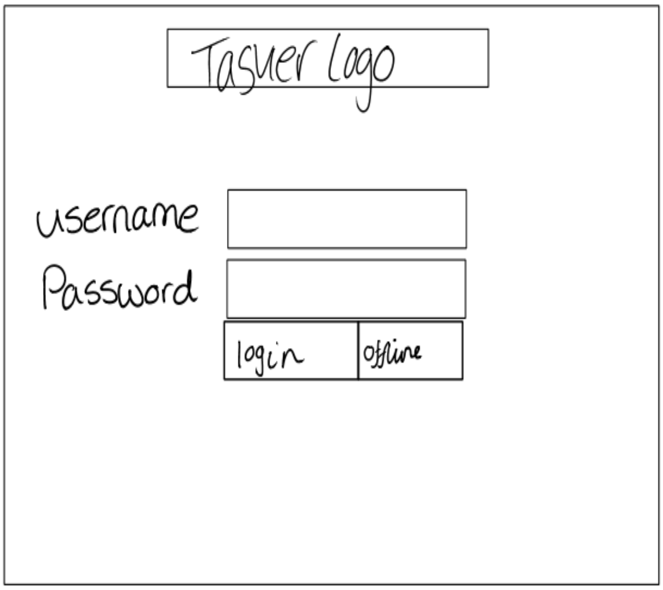
Deadline of task

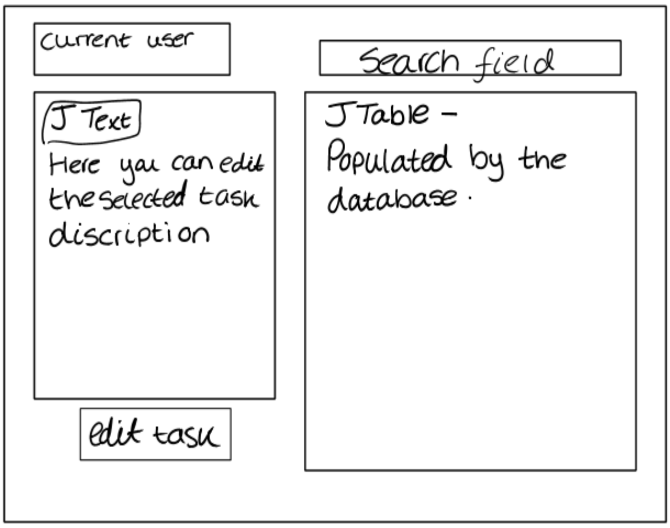
List of task elements, with at least one member

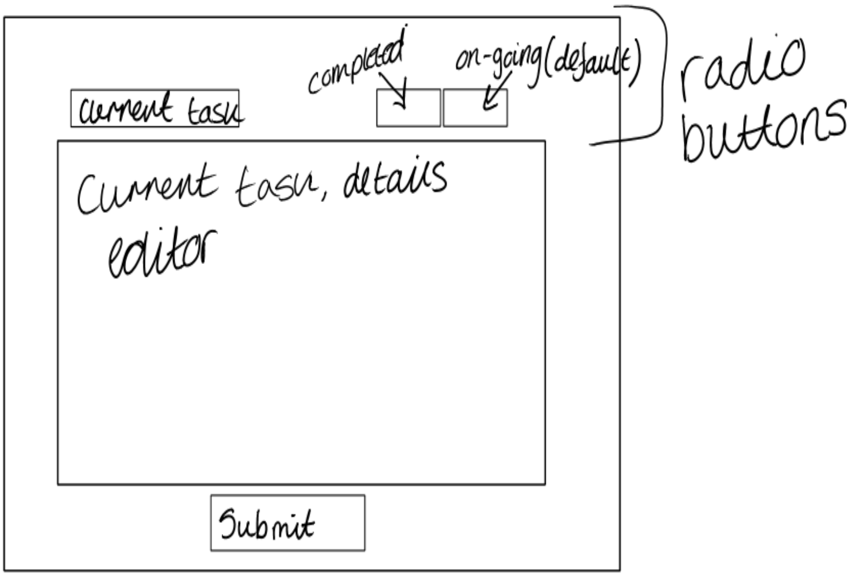
A task element is free text used to describe a step in the task, and associated free text for comments added by the member carrying out the task to report on progress with the step.

Status of current task

Below are the rough designs for the user application Tasker CLI.



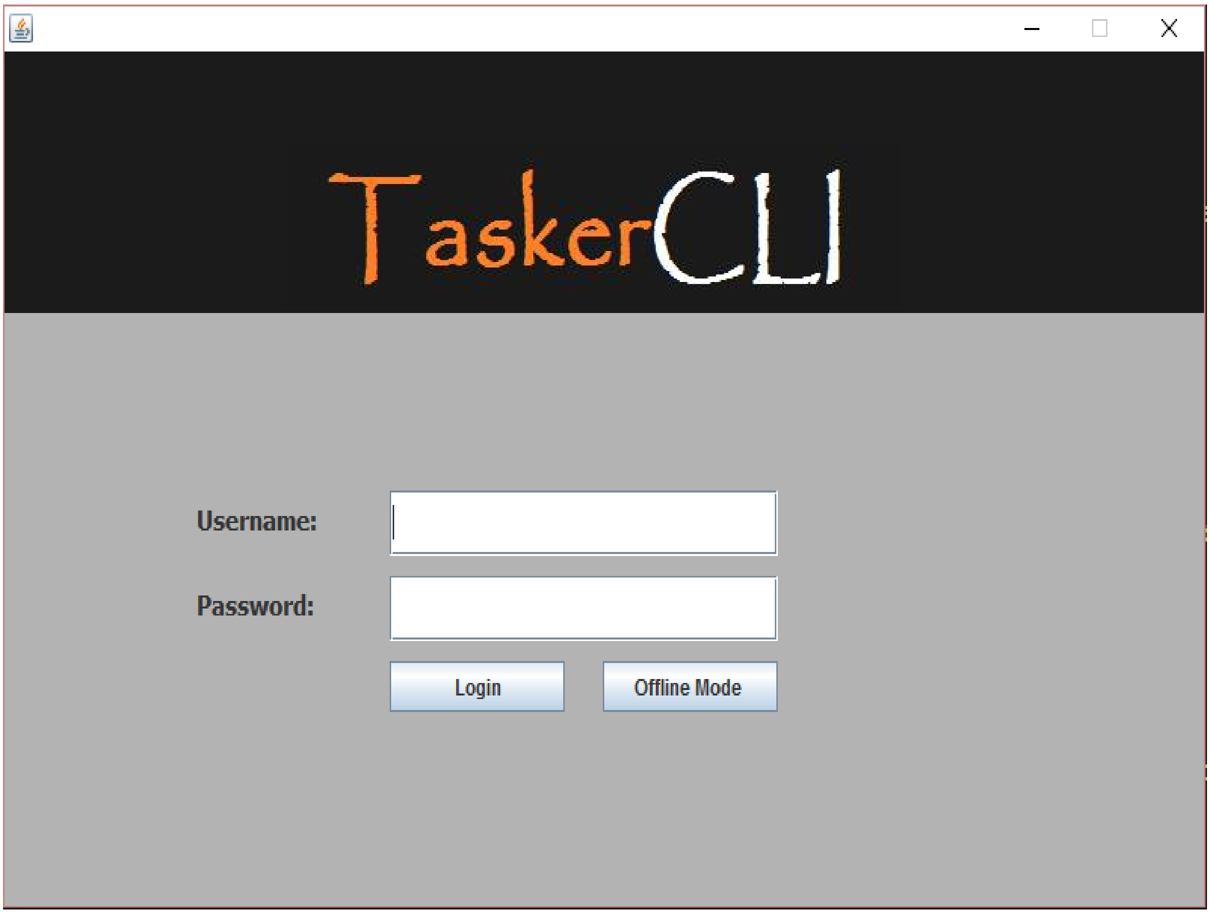

Current user
Search field
J Text
Here you can edit the selected task discription
edit task
J Table –
Populated by the database.



Below are the final designs for the user interface on the user application Tasker CLI

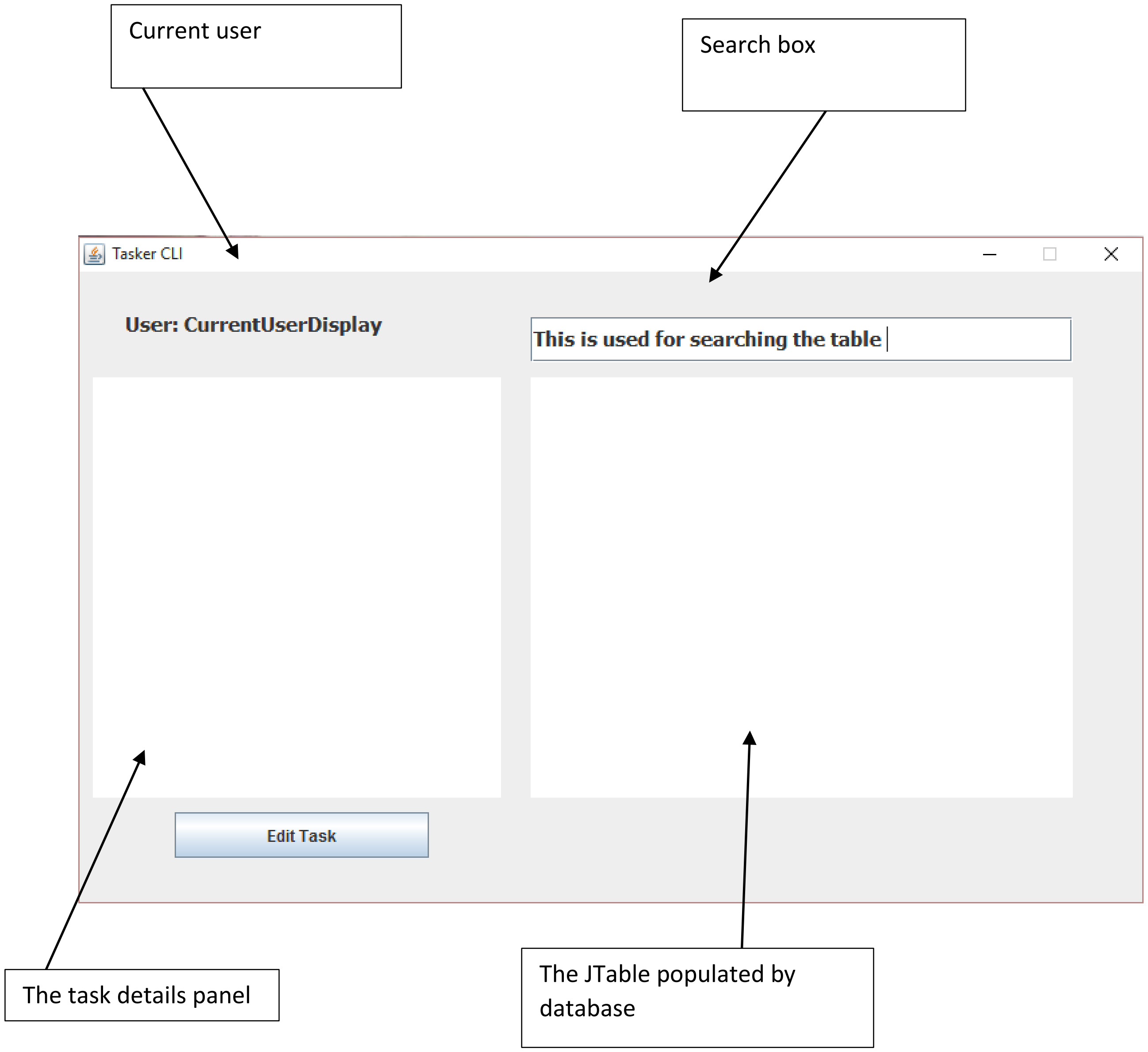
3.2.10 Login page

When the desktop application is first executed, this login windows appears requiring a username and password, which are stored in the database. Alternatively you can login using the offline mode, which uses the files stored on the local machine. Both buttons will redirect you to the user application.



3.2.11 User Application

The user application, is where the main program runs. Here the current user login is displayed, all the current tasks are displayed. Above the table there is search function. The JTable is populated by the database or by local files when running in offline mode. When a task is selected from the table the task details populate the Task details panel. If they wish to edit the task comments they can select the edit task button.

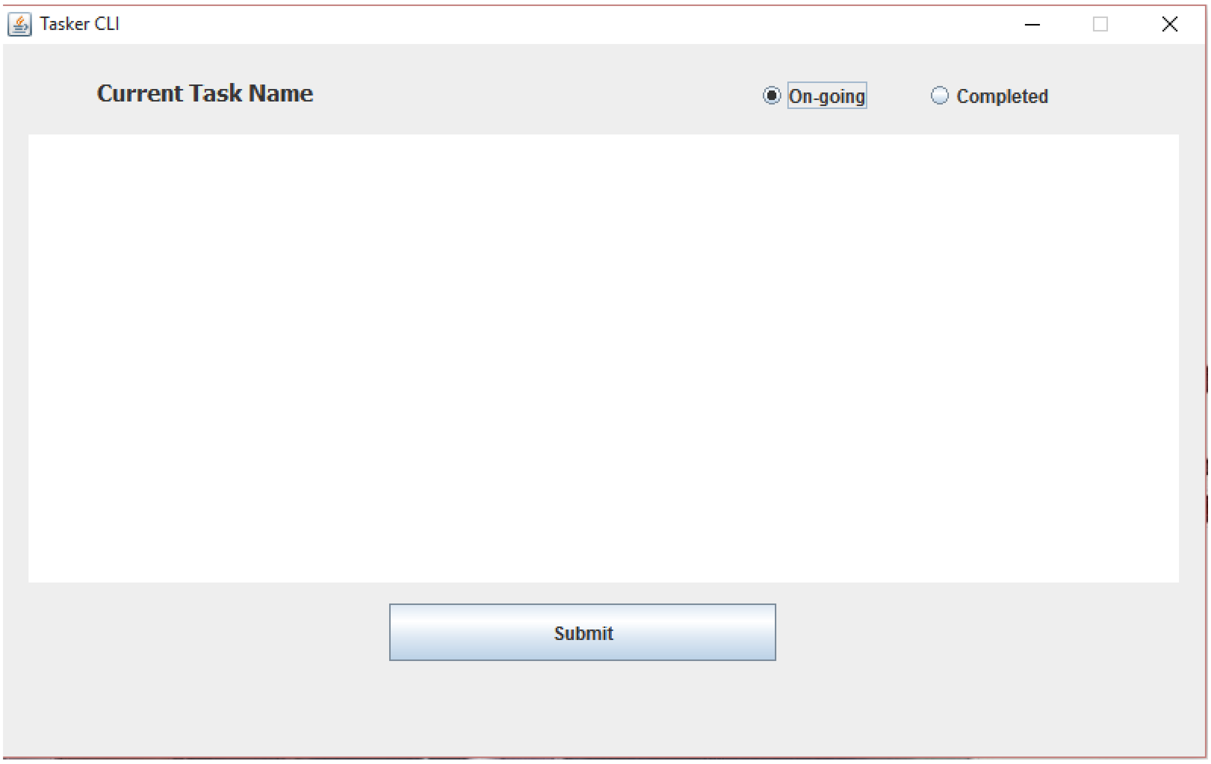


User: CurrentUserDisplay

This is used for searching the table

3.2.12 Editor

The editor window allows users to edit task details which are populated with data from the database or local files. When you press submit your changes will synchronise with the server, however if you’re in offline mode the changes will be saved to local storage and sync to the database as soon possible. You can also set the completion of the task, using the radio buttons. The current selection is retrieved from the database.



● On-going ○ Completed

References

Change History

Version	CCF No.	Date	Changes Made To Document	Changed By
1.1	N/A	2015-10-29	Format changed to suit design specification. Changed release version.	Robert Mouncer – rdm10