



Group Project 2015 - Design Specification

Group 07

Authors: K. Hodges, T. Doyle, C. Bassi, I. Gombar, J. Hallam, G. Heard, E. Dugdale.

Config Ref: SE.QA.05A

Date: 27-10-2015

Version: 1.1

Status: Release

Department of Computer Science,
Aberystwyth University,
Aberystwyth,
Ceredigion, SY23 3DB,
U.K

Copyright © Aberystwyth University 2015

CONTENTS

1. INTRODUCTION

1.1 Purpose of this document

1.2 Scope

1.3 Objectives

2. DEPLOYMENT DESCRIPTION

2.1 Applications in the system

2.2 Application interactions

3. INTERACTION DESIGN

3.1. Use-cases

3.2. TaskerMan User interface design

3.3. TaskerCli User interface design

4. COMPONENT DESCRIPTION

5. SIGNIFICANT CLASSES

5.1. Interface Description

6. DETAILED DESIGN

6.1. Sequence diagrams

6.2. State diagrams

6.3. Activity diagrams

6.4. Significant data structures

1. INTRODUCTION

1.1 Purpose of this document

The purpose of this document is to provide a design overview for the interaction and high-level design of our group project.

1.2 Scope

This overview design specification aims to provide information about the initial design for our system “Tasker”, giving information about the components in our system and how they will interact. Use-case diagrams will be used to then demonstrate system functionality, with corresponding user interface designs to show how a user would execute these actions.

1.3 Objectives

The objectives of this document are:

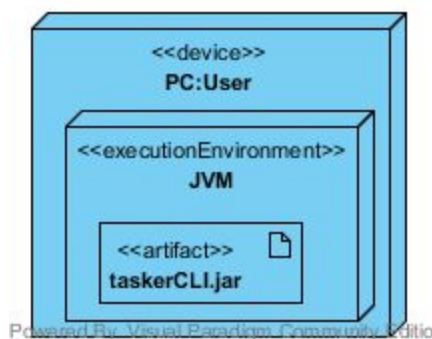
- To describe the deployment of the applications in our system:
 - Deployment of takserMAN, taskerCLI and taskerSRV
 - How these applications will interact with one another (where appropriate)
- To outline system design:
 - Use-case diagrams to show how the user would interact with the system
 - Designs of the user interface for each application (as listed above)

2. DEPLOYMENT DESCRIPTION

2.1 Applications in the system

We will employ many different applications within the system. The three subsystems we will be using within the system will be taskerCLI, taskerMAN and taskerSRV.

TaskerCLI



TaskerCLI will be a Java desktop application run in the JVM on a user's computer(s). It is designed to be used by users who have task(s) assigned to them by the manager in the taskerMAN application. A user will sign onto the system using their e-mail, and this is how they will be able to receive user-specific tasks assigned to them by the manager. Synchronisation between taskerSRV and taskerCLI will also be implemented.

The user can view, update and provide information on the tasks they have been assigned such as writing comments about task elements as they are being worked on, reading comments left by their manager on task elements as well as requesting to change a task status to be abandoned, although this will have to be confirmed by the manager using taskerMAN.

Thus, the primary use of the taskerCLI will be to allow users to easily provide feedback on their progress with assigned tasks, including any problems encountered, whilst also being able to see all tasks assigned to them by their manager, allowing them to easily keep track of their workload.

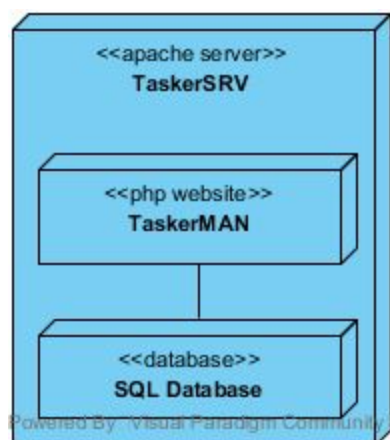
TaskerMAN

TaskerMAN will be a web based application running on an Apache web server. It will be used by the manager to assign tasks to users in a group. The manager has open access to the TaskerMAN application at any time using the web.

The manager can create and assign tasks to users, as well as having most of the functionality available to users in TaskerCLI. They can also approve or deny requests to change the task status.

TaskerSRV

The TaskerSRV will be an Apache web server running PHP and a MySQL database. It will be able to hold, edit, and create team member data (full name and e-mail address) and task data (task title, allocated team member, start date, expected completion date, a list of task elements and a status).



2.2 Application interactions

TaskerMAN and TaskerSRV

With TaskerMAN, the following information will be received from and stored in the database located on on TakserSRV:

- Relating to users:
 - First name
 - Last name
 - E-mail address

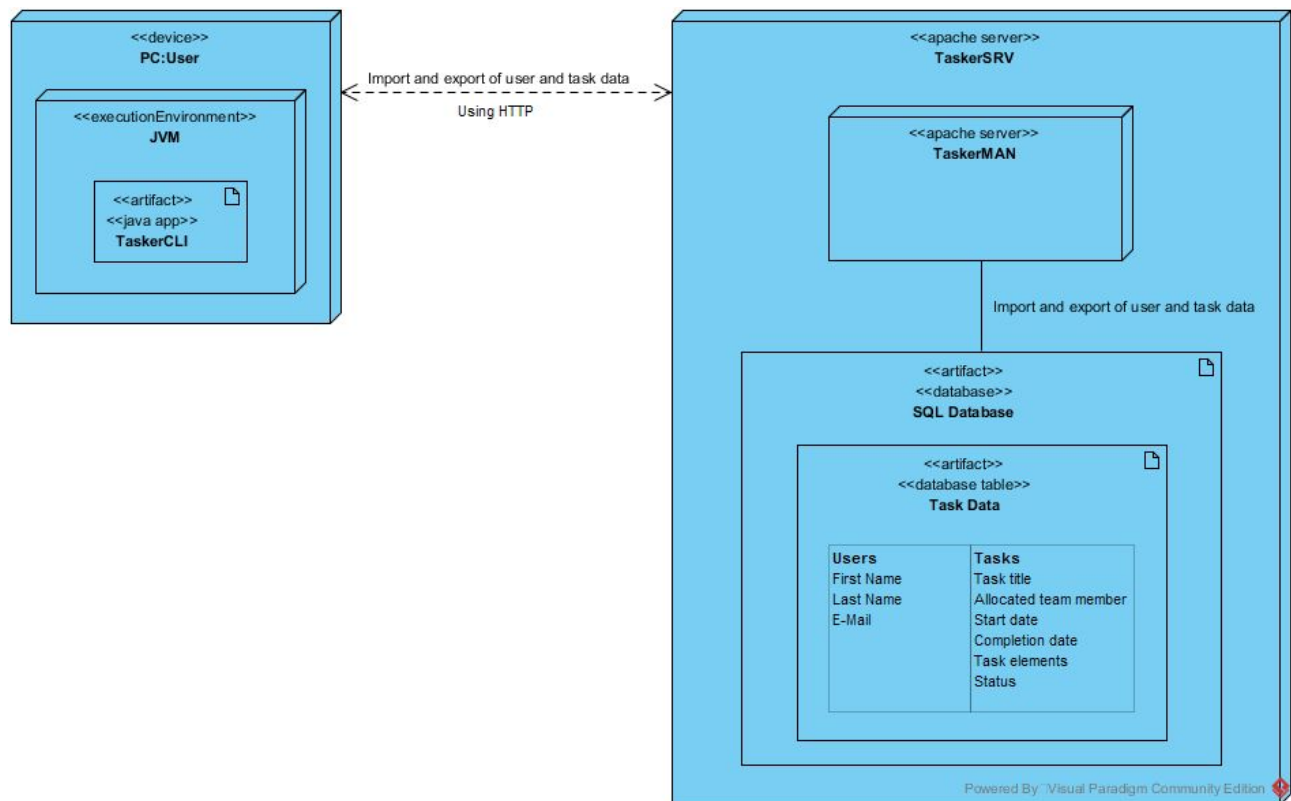
- Relating to tasks:
 - Task title
 - Allocated team member
 - Start date
 - Expected completion date
 - List of task elements
 - Status

TaskerCLI and TaskerSRV

With TaskerCLI, the following information will be received from and stored in the database located on TaskerSRV:

- User email (to authorise user access).
- Relating to task(s):
 - Task title
 - Start date
 - Expected completion date
 - List of task elements
 - Free text section for comments relating to each task element
 - Ability to flag tasks as “complete” (for review by the manager)

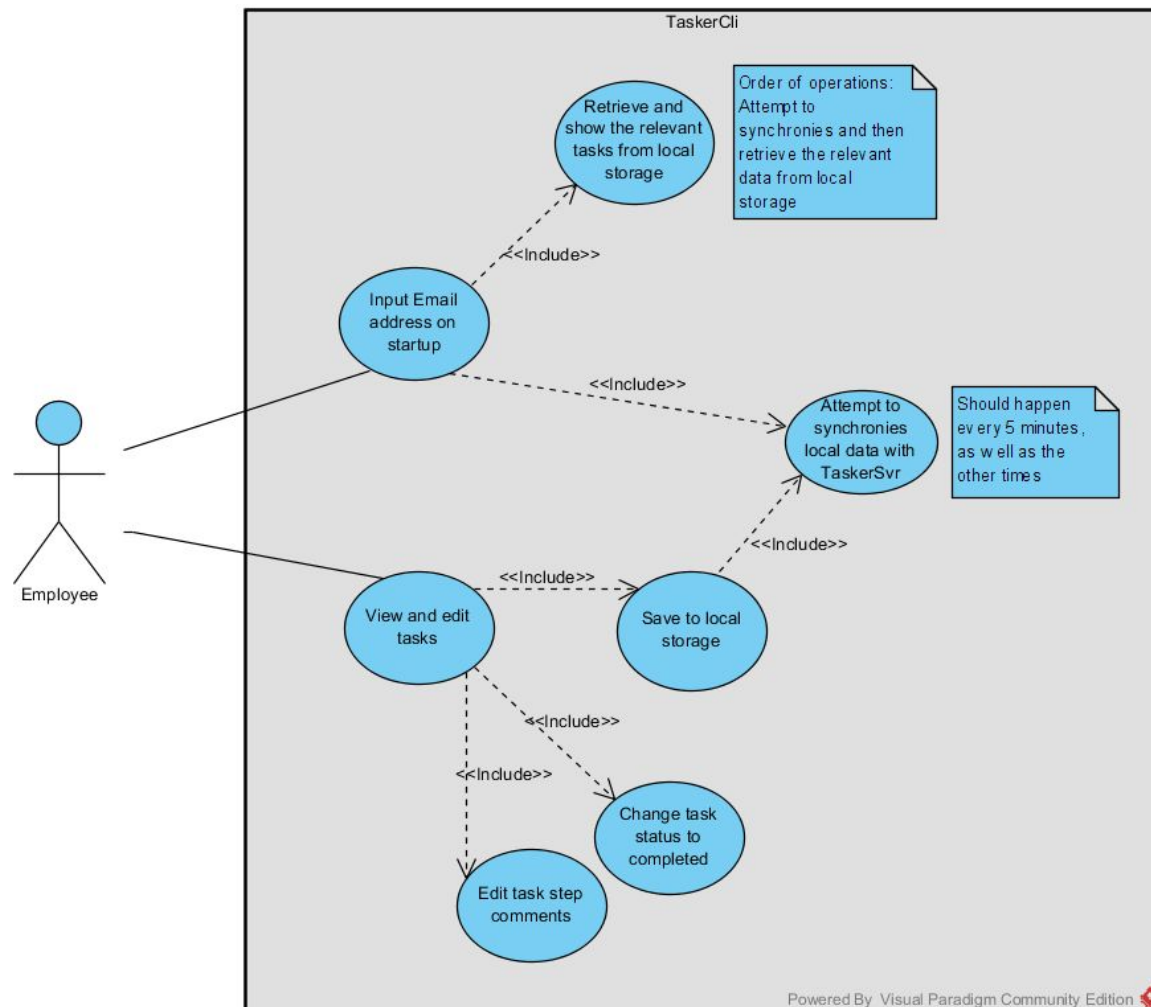
The diagram below shows the interaction between all components of the TaskerSRV, TaskerMAN and TaskerCLI.



3. INTERACTION DESIGN

3.1. Use-cases

The diagram below shows the interaction between TaskerCLI and the employee using it.

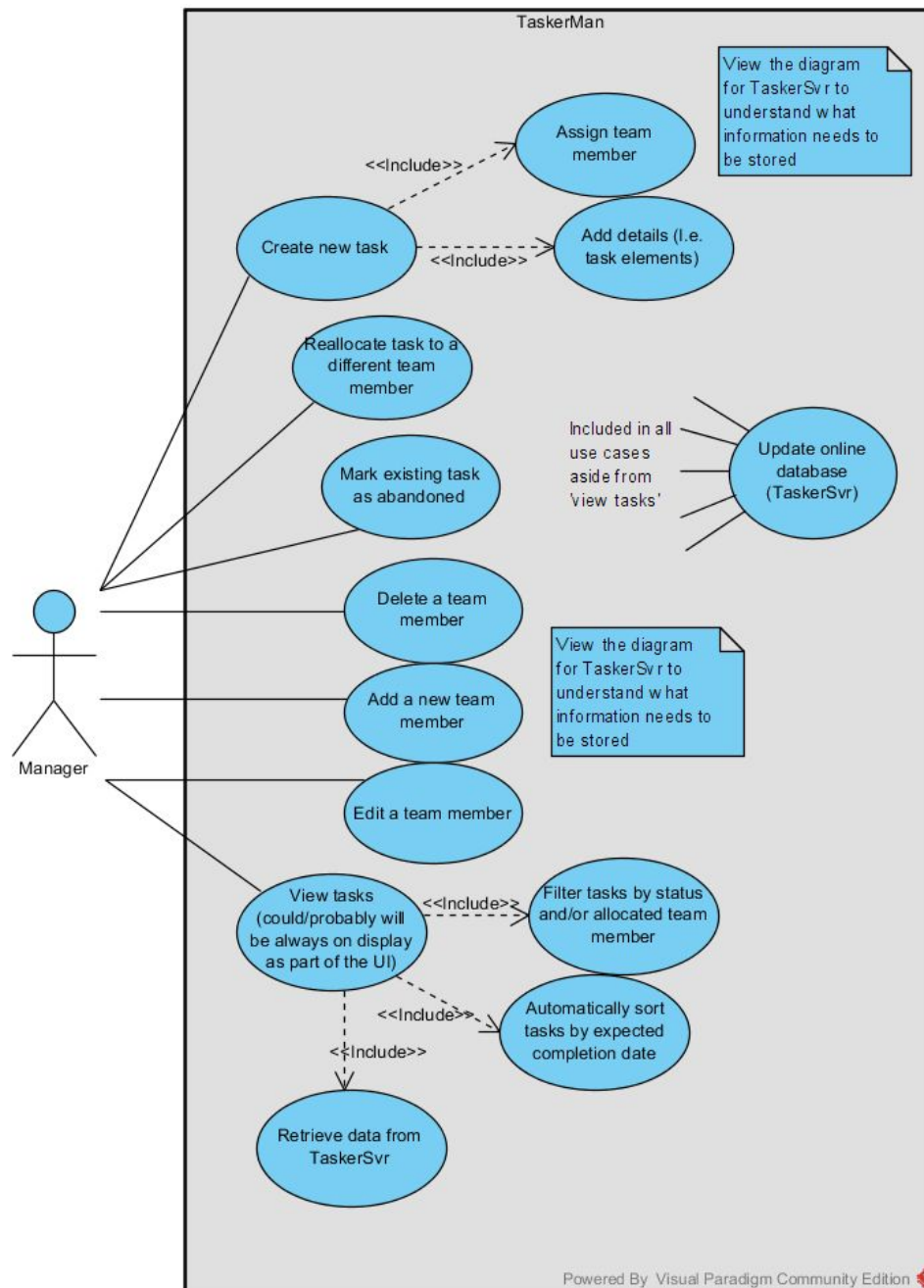


The user must input their email address (as seen in the top use case) before being able to proceed. The system then attempts to update the locally stored tasks with the tasks on TaskerSRV. All tasks then will be loaded from local storage even if synchronization failed.

The user can then view any of their assigned tasks, edit the comments for any of the provided task elements and/or change the status of the task to completed. After any edits have been confirmed the program first saves any changes to local storage and then attempts to synchronise with TaskerSRV. For example the user might want to update a task to tell the manager that 2 more of the task steps have been completed by editing

the comments for those task steps. Another user might want to set a recently finished task to completed.

The second use case diagram shows the interaction between TaskerMAN and the manager.



The manager does not have to log on to start using the application, although we may include this feature in future. The application will have to be able to retrieve all of the task data from TaskerSRV and show the manager a list of all of the tasks, sorted by

expected completion date. The manager must have the option to be able to filter this list as described in the diagram.

The manager must be able to make changes to existing tasks and create new tasks. The creation of new team members must also be supported as well as the ability to edit or delete existing team members. All changes should then be communicated with TaskerSRV.

For example the manager might want to view each task assigned to "M. Jones" that are marked as "allocated". This will allow the manager to keep updated on the progress being made for each task assigned to that employee. They might want to assign one of those tasks to a different user or mark a now unnecessary task as abandoned.

3.2. TaskerMAN User interface design

The diagram illustrates the TaskerMAN user interface. At the top left is the title "taskerMAN". At the top right are two buttons: "Managetasks" and "Manage members". The main content area is titled "Welcome" and contains a login form. The login form is titled "Please log in" and includes fields for "Username" and "Password", followed by a "Login" button. Two speech bubbles provide additional information: one on the left states "Username and password can be left blank or filled to log in", and one on the right states "This will be included in case log in authorisation is required in the future".

```
taskerMAN
```

Managetasks Manage members

Welcome

Please log in

Username

Password

Login

Username and password can be left blank or filled to log in

This will be included in case log in authorisation is required in the future

taskerMAN

Managetasks Manage members

New task Filter tasks

Task 1 Task 2 Task 3

Manage tasks

Manage tasks

"Manage tasks" is selected when in that part of the website

Also when having just logged in

This is the "default" page, where the user lands when "Managetasks" is clicked or "Cancel" clicked when making a new task

taskerMAN

Managetasks Manage members

New task Filter tasks

Task 1 Task 2 Task 3

New task

Task title [string(50)]

Allocated team member [dropdown]

Start date [dd] [dd] [dd]

Expected completion date [dd] [dd] [dd]

Create Cancel

Task elements

[multiline text area]

"New task" highlighted when user is on that page

taskerMAN

Manage tasks

Manage members

New task

Filter tasks

New task

Task 1

Task 2

Task 3

Task title

Company Website

Allocated team member

Doyle, Tom

Start date

[dd]

[dd]

[dd]

Expected completion date

[dd]

[dd]

[dd]

Create

Cancel

Task elements

Plan

Design

Write front end

Write rear end

Write report

Submit project

taskerMAN

Manage tasks

Manage members

New task

Filter tasks

Company Website

Task 1

Task 2

Task 3

Company Website

Task title

Company Website

Allocated team member

Doyle, Tom

Start date

[dd]

[dd]

[dd]

Expected completion date

20/01/2016

Status

"allocated" or "abandoned"

[dropdown]

Edit task

Cancel

Task elements

Plan

Design

Write front end

Write rear end

Write report

Submit project

Task auto selected when created

taskerMAN

Manage tasksManage members

New taskFilter tasks

Task 1

Task 2

Task 3

Company Website

Task auto selected when created

Company Website

Task title

Company Website

Allocated team member

Doyle, Tom

Start date

[dd][dd][dd]

Expected completion date

20/01/2016

Status

"allocated" or "abandoned"

[dropdown]

Edit taskCancel

Task elements

Plan

Design

Write front end

Write rear end

Write report

Submit project

taskerMAN

Manage tasksManage members

New taskFilter tasks

Task 1

Task 2

Task 3

Company Website

Manager given warning when status changed

EDIT MODE

Company Website

Doyle, Tom

[dd][dd][dd]

20/01/2016

[dropdown]

Confirm editCancel edit

Task elements

Plan

Design

Write front end

Write rear end

Write report

Submit project

taskerMAN

Managetasks

Manage members

New task

Filter tasks

Filter tasks

Task 1

Task 2

Task 3

Company Website

Filter by

☒ Task status

allocated

☒ Allocated team member

Doyle, Tom

Submit

Cancel

taskerMAN

Managetasks

Manage members

New task

Filter tasks

Filter tasks

Company Website

After filtering,
with a task
selected, filtered
tasks will remain
(filter will
remain active)

Filter by

☒ Task status

allocated

☒ Allocated team member

Doyle, Tom

Submit

Cancel

The wireframe shows the 'taskerMAN' application with a top navigation bar containing 'Managetasks' and 'Manage members' buttons. The 'Manage members' button is highlighted. Below the navigation bar, there is a 'New member' button on the left and a large 'Manage members' section on the right. The 'Manage members' section contains a list of member names: Bassi, Chris; Doyle, Thomas; Dugdale, Edward; Gombar, Ivan; Hallam, Jacob; Heard, George; and Hodges, Kurt. A vertical scrollbar is next to the list. A callout bubble points to the list with the text 'Will be available size to big'. Another callout bubble points to the 'Manage members' section with the text '“Manage members” is selected when in that part of the website'. A third callout bubble points to the 'Manage members' section with the text 'This is the “default” page, where the user lands when “Manage members” is clicked'.

taskerMAN

Managetasks Manage members

New member

Manage members

Will be available size to big

“Manage members” is selected when in that part of the website

This is the “default” page, where the user lands when “Manage members” is clicked

Manage members

Bassi, Chris

Doyle, Thomas

Dugdale, Edward

Gombar, Ivan

Hallam, Jacob

Heard, George

Hodges, Kurt

The wireframe shows the 'taskerMAN' application with a top navigation bar containing 'Managetasks' and 'Manage members' buttons. The 'Manage members' button is highlighted. Below the navigation bar, there is a 'New member' button on the left and a large 'New member' section on the right. The 'New member' section contains a form with three input fields: 'Last name' (string(25)), 'First name' (string(25)), and 'E-mail address' (string(254)). A callout bubble points to the 'E-mail address' field with the text 'Validation provided by JavaScript'. Below the form, there are two buttons: 'Create' and 'Cancel'. A callout bubble points to the 'Create' button with the text 'Creates and selects the user'. Another callout bubble points to the 'Cancel' button with the text 'Goes to the previous page'.

taskerMAN

Managetasks Manage members

New member

New member

Last name [string(25)]

First name [string(25)]

E-mail address [string(254)]

Validation provided by JavaScript

Create

Cancel

Goes to the previous page

Creates and selects the user

Bassi, Chris

Doyle, Thomas

Dugdale, Edward

Gombar, Ivan

Hallam, Jacob

Heard, George

Hodges, Kurt

taskerMAN

ManagetasksManage members

New member

Bassi, Chris

Doyle, Thomas

Dugdale, Edward

Gombar, Ivan

Hallam, Jacob

Heard, George

Hodges, Kurt

New member

Last name

Marley

First name

Bob

E-mail address

reggaeking@hotmail.com

Create

Cancel

taskerMAN

ManagetasksManage members

New member

Bassi, Chris

Doyle, Thomas

Dugdale, Edward

Gombar, Ivan

Hallam, Jacob

Heard, George

Hodges, Kurt

Marley, Bob

New member

Last name

Marley

First name

Bob

E-mail address

reggaeking@hotmail.com

Edit member

Cancel

Member auto selected when created

Goes into edit mode

The image shows a web interface titled "taskerMAN". At the top right are two buttons: "Managetasks" and "Manage members". Below these is a "New member" button. The main area is titled "New member" and contains an "EDIT MODE" section. On the left is a vertical list of member names in buttons: Bassi, Chris; Doyle, Thomas; Dugdale, Edward; Gombur, Ivan; Hallam, Jacob; Heard, George; Hodges, Kurt; and Marley, Bob. The "Marley, Bob" button is highlighted. To the right of this list is a form with the following fields: "Last name" (Marley), "First name" (Bob), "E-mail address" (reggaeking@hotmail.com), and "Delete member?" (a dropdown menu). Below these fields are two buttons: "Confirm edit" and "Cancel edit".

3.3. TaskerCLI User interface design

The image shows a login screen titled "Tasker Cli". Below the title is the text "Welcome, Please enter your email address:". There is a text input field containing "remembered@email.com". Below the input field is a "Remember me?" label followed by a checked checkbox (X). At the bottom is a "Login" button. A speech bubble points to the checked checkbox with the text "Shows the last input email if the box was checked".

Tasker Cli

Current tasks

Task1

Design a new UI

Task3

Task4

Task5

Task6

Task7

Welcome

Select the task you wish to view

Tasker Cli

Current tasks

Task1

Design a new UI

Task3

Task4

Task5

Task6

Task7

Design a new UI

Start Date: xx/yy/zz

Expected completion date: aa/bb/cc

Task details

- Draw the initial design

//Add your comments here

- Ask boss for feedback

Task completed? ☐

Submit changes

Cancel changes

Might
change to
dropdown
box

The scaling is far
from accurate. In
reality more will be
able to fit on screen
as well as more text
per element.