Tasker

Charter/Scope/WBS

Tasker

Liam Gooch and Mischa Bergmoser

Table of Contents

[Charter 3](#_Toc535329755)

[Description 3](#_Toc535329756)

[Objectives 3](#_Toc535329757)

[Budget 3](#_Toc535329758)

[Major Deliverable Schedule 3](#_Toc535329759)

[Critical Success Factors 3](#_Toc535329760)

[Risks and Concerns 3](#_Toc535329761)

[Comments 3](#_Toc535329762)

[Scope 4](#_Toc535329763)

[Project Justification 4](#_Toc535329764)

[Long Term Objective 4](#_Toc535329765)

[Characteristics and Requirements 4](#_Toc535329766)

[Admin 4](#_Toc535329767)

[User 4](#_Toc535329768)

[Tasks 5](#_Toc535329769)

[Accounts 5](#_Toc535329770)

[Summary of Project Deliverables 5](#_Toc535329771)

[Project Related Deliverables 5](#_Toc535329772)

[Project Success Criteria 5](#_Toc535329773)

[WBS 6](#_Toc535329774)

[Specifications 7](#_Toc535329775)

[Login Page 7](#_Toc535329776)

[Create Account Page 7](#_Toc535329777)

[Admin Page 7](#_Toc535329778)

[Task Fragment 7](#_Toc535329779)

[Projects Fragment 7](#_Toc535329780)

[Employees Fragment 8](#_Toc535329781)

[User Page 8](#_Toc535329782)

[Add Task Page 8](#_Toc535329783)

[Edit Task Page 8](#_Toc535329784)

[Online Database 8](#_Toc535329785)

[Designs 9](#_Toc535329786)

[Accounts 9](#_Toc535329787)

[Admin 9](#_Toc535329788)

[User 9](#_Toc535329789)

# Charter

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Name:** | Tasker Android | **Project Start Date:** | 01/01/2019 |
| **Application Name:** | Tasker | **Project End Date:** | End of Feb |

## Description

Tasker is a “to-do list” based application to be commissioned for Android, with plans to expand to iOS and Windows devices in the future. Tasker will serve as a communication platform between two parties; the admin and the user. The admin can send a task to a user with details as to what needs to be completed, when it needs to be completed and can be given a priority. The user will receive the task and is able to set a reminder and mark it as complete. There will be only two account types available and users will not be able to switch between these accounts; however, should this change in the future, the system must be designed to allow for a change in accounts. Opening an admin account will incur a fee, the extent to which is undecided at this stage.

## Objectives

* Create secure login and account capabilities
* Allow admin accounts to add tasks to its users
* Allow users to read tasks and mark as complete

## Budget

None

## Major Deliverable Schedule

* January
  + Account scheme
    - Create and login
  + Login UI
  + Create account UI
  + Admin and User UI
  + Task creation
    - Add, edit and remove
  + Task viewing
  + Local and online database syncing of tasks
* February
  + Local and online database syncing of tasks

## Critical Success Factors

* Logins
* Task creation by admin and user can view
* Syncing database

## Risks and Concerns

* Too much work for a small timeframe
* Difficulty in establishing demand for a paid version
* Inability to stand out amongst already flooded market

## Comments

None

# Scope

## Project Justification

Tasker system will act as a base system which can then be re-modelled to meet the requirements of multiple industries. Upon completion of the Tasker system, the source code can then be sold with the intention that it can be used as a base for development of internal systems for organisations.

### Long Term Objective

Potential markets Tasker system can be used in may include

* Small to medium business operation and communication
* Education
* Rental market

## Characteristics and Requirements

### Admin

* Have employee groups
  + Add, remove and edit users in these groups
* Add, remove and edit employees
* Set tasks for multiple users, single users and themselves
  + For multiple users
    - Admin chooses whether each member can see all other members who have this task
  + User gets a notification of added task
  + Task name
  + Task description
  + Date commissioned
  + Date due
  + Priority
  + Admin can see who has completed the task
    - Must mark accepted then it is automatically removed from that users’ inventory
    - Task must be removed from admins list by hand
* Remove tasks from users
  + Users get a notification of removed tasks
* Must oversee removal of completed tasks
* Can set tasks offline but must sync with server
* Can create ‘Task projects’
  + Projects may consist of multiple tasks each with their own completion dates
  + Allows for large projects to be broken down into smaller tasks in order to avoid overwhelming workflows
* Each project can has a set of tasks with the same rules as an ordinary task

### User

* Can request to join group
  + With group code
* Can see all groups they are members of
* Can see everyone with same task (if admin allows)
* Can set reminder or alarm
* Can mark tasks completed
  + Admin will receive notification upon completion
* Can see tasks preloaded offline
* Can remove tasks on request of admin
* Can set follow up dates on assigned tasks to assist with longer term tasks
* Can edit details in order assist with management of user accounts

### Tasks

* Have a priority
* Have a set date and completion date
* Can send push reminders
  + Notifications
  + Alarms
* Have a progress update
  + Set by user
  + Admin can request status update
  + Not-started, in-progress, almost complete, complete
* Can be completed

### Accounts

* Edit accounts
  + Email
  + Name (upon request)
  + Password

# Summary of Project Deliverables

### Project Related Deliverables

* Accounts
* Logins
* UI’s
* Add, edit remove tasks
* User groups
* Database syncing
* Complete tasks

## Project Success Criteria

80% of functional requirements must be met by the end of January.

# WBS

TO BE COMPLETED

# Specifications

## Login Page

Upon app opening, will check with a local database to ensure that a user is logged in.



If the user is not logged in, it will ask for an email and password, attached to the account. The details will then be checked against a firebase database, the account type retrieved and the appropriate account page opened.

## Create Account Page

This page can be opened by tapping “Create Account” from the login page. Will ask the user for their account type, name, email and to choose a password. A verification email will then be sent to their inbox and they must verify their email before they are able to login. The email and password are then sent back to the login page to be added to the login screen.

## Admin Page

A new admin will be greeted with a dashboard fragment which has no tasks and a button to add a task. This page will be the default fragment unless otherwise specified by the admin in the *settings* page (this is not a functional requirement). Bottom navigation will be used to allow the admin to access their dashboard over three categories (and their sub categories);

* Tasks – Tasks will open an Edit Task page (can see users who have completed this task)
  + Alphabetical
  + Completion date (default)
* Projects – Projects will open another page with all tasks relating to that project able to be edited
  + Alphabetical
  + Completion date (default)
* Employees – Similar to projects, each employee group will be editable.
  + Alphabetical (if only one group exists)
  + Groups (default otherwise)

In each fragment there will be a Floating Action Button (FAB) to add either a task, project or employee group.

### Task Fragment

Simply a recycler view of tasks. The tasks are loaded from an online database and synced to a local database (overridden) for offline usage – on each data change.

### Projects Fragment

If only one project exists, it will simply show a list of tasks in that project.

Upon opening the project fragment a recycler view will be used to show a list of projects, which can be opened to show a page full of the tasks involved. Projects and subsequent tasks will be synced with an online database and saved locally (overridden) – on each data change.

### Employees Fragment

Similar to the Projects Fragment but will show employee groups and employees instead.

## User Page

The user will be greeted with a fragment layout similar to that of the admin with three categories to sort from using bottom navigation.

* Tasks – Tasks will open a View Task page
  + Alphabetical
  + Completion date (default)
* Projects – Projects will open another page with all tasks relating to that project able to be viewed
  + Alphabetical
  + Completion date (default)
* Groups – Similar to projects, each employee group can be viewed

## Add Task Page

Accessible by both the admin and user; however, only an admin can assign a task to multiple users. Adding a tasks will involve adding;

* Name
* Description
* Date due
* Priority
* Users involved (admin only)
* Reminder (user can only set for themselves)

As a task is added a commissioned date will be added to the task details.

## Edit Task Page

Can edit any detail of the task.

## Online Database

TBC

# Designs

## Accounts

|  |  |
| --- | --- |
|  |  |
| Login Page | Create Account Page |

## Admin

UI  
Tasks  
Projects  
Employees  
Add UI  
Edit UI

## User

View groups UI  
View tasks UI  
Completed task