Description

Intended User

Features

User Interface Mocks
- Screen 1 All Projects (Main)
- Screen 2 New Project (Edit Project)
- Screen 3 Project Detail
- Screen 4 Actions
- Screen 5 New Action(Edit Action)
- Screen 6 Action Detail
- Screen 7 Notes list
- Screen 8 Note
- Screen 9 Widget
  - How will your app handle data persistence?
  - Describe any edge or corner cases in the UX.
  - Describe any libraries you'll be using and share your reasoning for including them.
  - Describe how you will implement Google Play Services or other external services.

Next Steps: Required Tasks
- Task 1: Project Setup
- Task 2: Create DataBase
- Task 3:Add Basic UI
- Task 4:Add Extra functionality
- Task 5:Polish UI
- Task 6:Test and create a release build

GitHub Username: Pongopom

PersonalPM

Description

PersonalPM is a basic project management tool that helps you manage your projects without the complexity and cost of a professional Project Management Tool. PersonalPM works by breaking your projects down into Actions. PersonalPM allows you to create a project set its priority, start date, completion date and create actions. By breaking your projects down into actions it is easy to see which parts of your project need your attention. Like projects actions can have there start date, end date, and priority set as well as the following.

1. Assign an Owner to an action (Person or organisation responsible for the action).
2. Set the size or scale of the action from 1 to 10. For example you would might set the scale to 8 for building a wall and only 2 for painting it.
3. Set the progress of the action as a percentage 0 not started 100 complete.
4. Create and share multiple notes for each action

Intended User

PersonalPM is a Project Management tool for the rest of us.
All of us have projects to manage at some point be it a large DIY project or organise a wedding. Personal PM will be of use to all of us from time to time but will be especially use full to Small Building Contractors, electricians, plumbers and trades that need to manage projects and where the use of a professional Project Management Tool will be over complicated and to expensive.

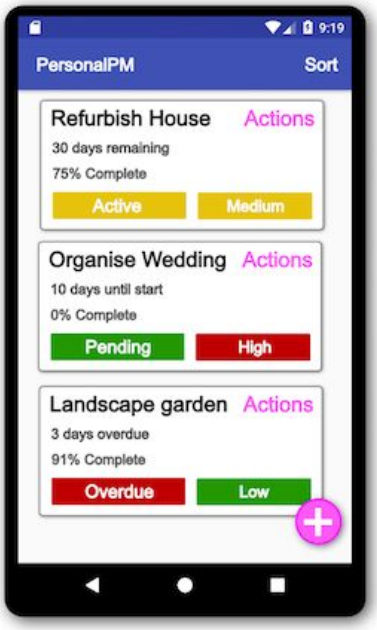
Features

List the main features of your app. For example:

- Create multiple Projects
- Break projects down into various actions
- Use the Actions to keep track of parts of your project that need your attention.
- Create multiple notes for each action
- Share notes with interested parties (contractors, suppliers, etc)

User Interface Mocks

Screen 1 All Projects (Main)



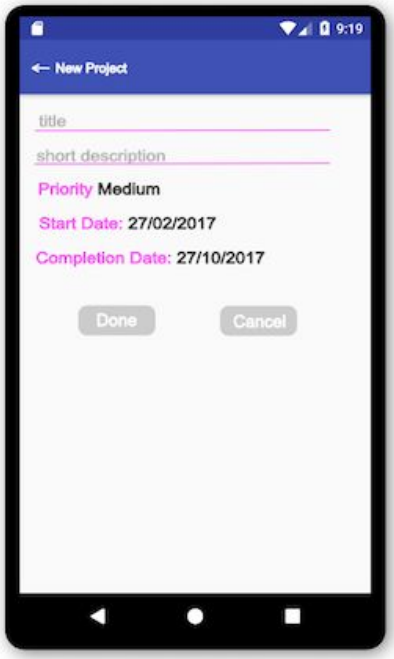
All Projects Screen

This is the first screen showing multiple projects.

1. Tap the + FAB button to create a new project.
2. Tap the Sort button to see a list of project sort options.
3. Tap the project card to be taken to the projects detail screen
4. Tap the Actions button to be taken to that projects list of actions(shortcut to actions).

A project can have 4 status types Pending(Green), Active(Yellow), Overdue(Red) and Complete(Blue). A project can be given a priority of Low(Green), Medium(Yellow) or High(Red). A Project can be sorted by name, status, priority, start date or completion date

Screen 2 New Project (Edit Project)



New Project screen

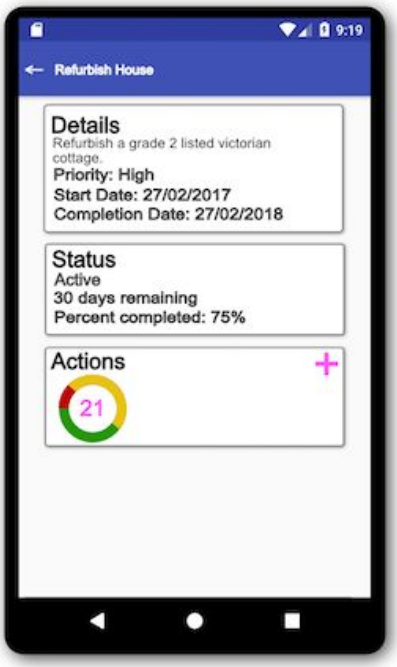
This screen is shown when the user taps the + fab button on Screen 1(the All Projects Screen).

1. Give the project a title.
2. Give the project a short description.
3. Set the project priority.

Tap done to save the project or cancel.

Note this screen will also be used to edit existing projects. In edit mode the bar title will be Edit Project and there will also be an option to delete a project.

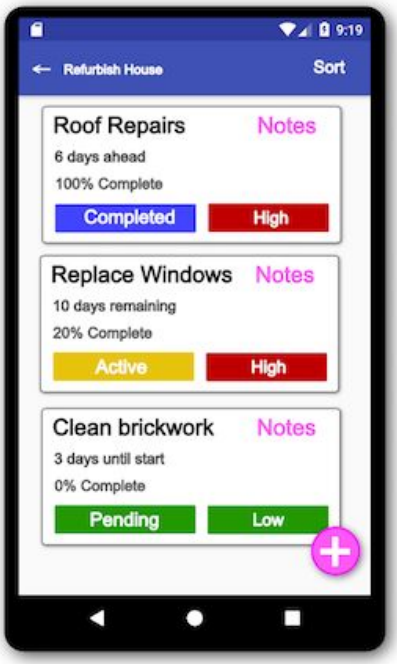
Screen 3 Project Detail



Project Detail Screen

Tapping on a project card in the All Projects Screen will bring the user to the Project Detail Screen. This screen shows the project detail and includes the actions that make up the project. Tap the + to create a new action or tap the chart to navigate to the list of actions for this project. The action chart allows you to quickly see the amount of actions as well as the status Pending(Green), Active(Yellow), Overdue(Red) and Complete(Blue). So in the example above the user can quickly see there is 21 Actions. Half are active, None are complete, nearly half are pending and there are a few that are overdue.

Screen 4 Actions

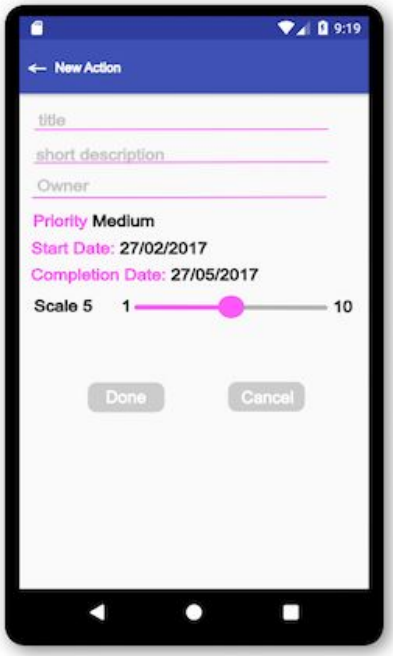


Actions Screen

Tapping on the Actions chart in the Project Detail Screen or tapping on the actions button(shortcut to actions) on the All Projects Screen will bring the user to the Actions Screen. This is a list of all the actions that make up a project it is very similar to the All Projects Screen as Actions are basically mini projects with a few differences we will discuss later. Like projects a action can have 4 status types Pending(Green), Active(Yellow), Overdue(Red) and Complete(Blue). An action can be given a priority of Low(Green), Medium(Yellow) or High(Red). An action can be sorted by name, status, priority, start date or completion date

1. Tap the + FAB button to create a new action.
2. Tap the Sort button to see a list of action sort options.
3. Tap the action card to be taken to the actions detail screen
4. Tap the Notes button to be taken to that actions list of notes(shortcut to notes).

Screen 5 New Action(Edit Action)



New Action Screen

This screen is shown when the user taps the + fab button on Actions Screen. As discussed earlier actions are similar to projects the difference being. The user can assign a person or organization as an owner of the action. The user can set the size or scale of the action from 1 to 10. There is a couple of other differences that we will look at in the Action Detail Screen.

1. Give the action a title.
2. Give the action a short description.
3. Set the action priority.
4. Tap done to save the action or cancel.

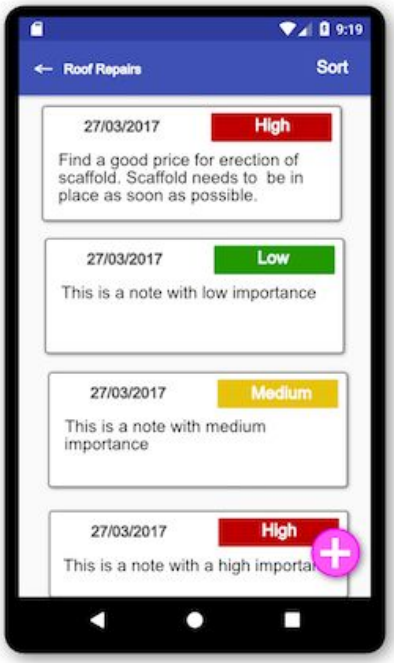
Note this screen will also be used to edit existing actions. In edit mode the bar title will be Edit Action and there will also be an option to delete an action.

Screen 6 Action Detail

Action Detail Screen

Tapping on a action card in the Actions Screen will bring the user to the Action Detail Screen. This screen shows the action detail and includes the notes that have been added to the action. Tap the + to create a new note or tap the chart to navigate to the list of notes for this action. The note chart allows the user to quickly see the amount of notes as well as the proportions of priority high medium or low. The slider(seekBar) lets the user easily update the progress of the action as it progresses.

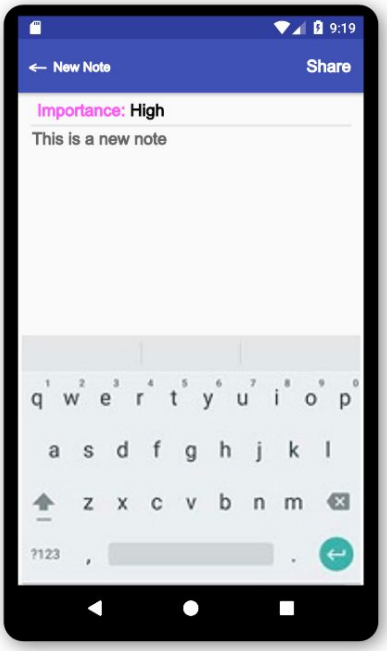
Screen 7 Notes list



Notes List Screen

Tapping on the Notes chart in the Action Detail Screen or tapping on the note button(shortcut to notes) on the Actions Screen will bring the user to the Actions Screen. This is a list of all the notes added to an action. Notes can be sorted by creation date or importance. Tapping the + FAB button will create a new note.

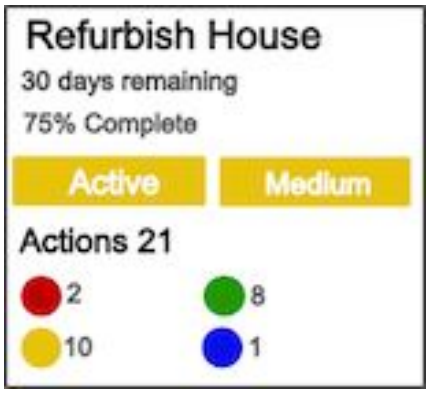
Screen 8 Note



Note Screen

This is a note the user can set the importance of a note to High, Medium or Low.
A user can share notes. Notes belong to an action.

Screen 9 Widget



Widget

The application will show a widget with a chosen projects details.

Key Considerations

How will your app handle data persistence?

The Application will use SQLite and a SQLiteOpenhelper subclass to create the database which will consist of three tables Projects Actions and Notes. The application will use a content provider to access the dataBase.

Describe any edge or corner cases in the UX.

Case 1
If a user is in the middle of creating a new project and leaves the screen before tapping done. If the user has entered a title the application will save the project. Any empty fields will be filled in with default values. The user will be able to edit these later if they wish. If the use has not entered any data the application will not save the project.

Case 2
If a user is in the middle of creating a new action and leaves the screen before tapping done. If the user has entered a title the application will save the project. Any empty fields will be filled in with default values. The user will be able to edit these later if they wish. If the use has not entered any data the application will not save the action.

Case 3
If a user is in the middle of creating a new note and leaves the screen before tapping done. If the user has not entered any text into the note the application won't save the note.

Describe any libraries you'll be using and share your reasoning for including them.

Butter knife to reduce boilerplate code.
MPAndroidchart for the pie charts in the Project and Action detail views.
Apache POI this is to allow the user to export a project as a SpreadSheet.

Describe how you will implement Google Play Services or other external services.

Google drive. The application will use this service to allow the user to export a project as a spreadsheet to their google drive.
Google ads. The application will use ads for monetisation.

Next Steps: Required Tasks

Task 1: Project Setup

1. Create the application using Android Studio
2. Add all the needed dependencies to the gradle
3. Check that we have the latest versions for the repositories.
4. Add any needed permissions to the Android Manifest.
5. Make sure the application runs with no errors or warnings.
6. Push the application to gitHub (first push).

Task 2: Create DataBase

1. Add a SQLiteOpenHelper subclass to build a database with 3 tables Projects, Actions and Notes.
2. Add a ContentProvider subclass the application will access the database via the Content Provider.
3. Add a temporary fake data class to test the basic functionality of the database.
4. Remember to keep pushing to gitHub.

Task 3:Add Basic UI

1. Add the the activities and Fragments needed for core functionality.
2. Add any views needed for core functionality
3. Add any loaders needed to query the database
4. Get UI showing real data.

Task 4:Add Extra functionality

1. Add menus (Bar Buttons).
2. Add Sort functionality.
3. Add Sharing of Notes via email etc.
4. Add export project as spreadsheet connect to google drive
5. Add ad support.

Task 5:Polish UI

1. Add split pane support for tablet
2. Add any missing ui elements graphs etc.
3. Add polish using material design guidelines.

Task 6:Test and create a release build

1. Unit test application functionality
2. Unit test UI.
3. Test using on the device .
4. Test again.
5. Create a release build signed certificates etc.