React Native Assignment 1 – Home Budget Mobile App

## Problem Statement

Create a mobile app for managing the monthly home budget. App should allow user to save the monthly planned and actual expenditure per item. As part of this assignment, you only need to create the save module of the app. Below are the detailed requirements.

## Requirements

The mobile app should have the following:

|  |  |  |
| --- | --- | --- |
| **Screen** | **Use case** | **Functional requirement** |
| Budget entry screen | User can enter per item budget details | 1. Screen has the title ‘Budget entry’ 2. Following are the input fields: 3. ***Name*** of the item like Sports, Education, Grocery, Entertainment, etc. 4. ***Planned amount*** *like 1000* 5. ***Actual******amount*** *like 200* 6. **Save** button to save the item in memory-based list. 7. **Show items** button to navigate to the ‘Budget entries listing’ screen. |
| Budget entry listing screen | User can see all his saved budget entries | 1. Screen has a title ‘Budget entry listing’ 2. List of the entries saved in the memory-based list 3. Fields in the list are ***Name*** of the items like Sports, Education, etc. |

## Technical Requirements

1. Use Redux.
2. The data should persist even after the app is closed
3. Use of material UI only for styling purpose.

***Note: Take relevant assumptions during the application development wherever either requirements are not clear to you or the information required is not provided in this case study.***

React Native Assignment 2 – Contacts App

## Problem Statement

Create a mobile app for managing the contacts. App should allow user to save the contact details in local database along with the photo. User can see contact list and can modify/delete any contact. Also, user can favorite/unfavorite contacts and can see list of favorite contacts. Below are the detailed requirements.

## Requirements

The mobile app should have the following:

|  |  |  |
| --- | --- | --- |
| **Screen** | **Use case** | **Details** |
| Contact list screen | User can see list of contacts in ascending order by name | 1. Screen has the title ‘Contact List’ 2. List of contacts with contact name and photo in ascending order 3. This screen will have a ‘add’ button at the bottom right of the screen 4. Pressing this button will take user to ‘create new contact screen’ 5. In contact list, by clicking on any contact, user will navigate to ‘Update contact screen’ |
| Create new contact screen | User can add a new contact | 1. Screen has the title ‘Add New Contact’ 2. Following are the input fields: 3. **Name** **of person** 4. **Mobile phone number** 5. **Landline number** 6. **Take/Browse photo of person** 7. **Favorite** button to mark/unmark the contact as favorite 8. **Save** button to save the contact details |
| Update contact screen | User can update contact details | 1. Screen has the title ‘Update Contact’ and will show details of selected contacts. 2. Following are the input fields: 3. **Name** **of person** 4. **Mobile phone number** 5. **Landline number** 6. **Take/Browse photo of person** 7. **Update** button to update the contact details 8. **Favorite** button to mark/unmark the contact as favorite 9. **Delete** button to delete the contact details |
| Favorite contact list screen | User can see list of contacts marked as favorite | 1. Screen has the title ‘Favorite Contact List’ 2. List of favorite contacts in ascending order |
| App Navigation | User can navigate to contact list screen and favorite contact list screen | 1. App will have a sliding drawer (using master-detail page) with two options: ‘Contact list screen’ and ‘Favorite contacts.’ 2. Clicking on any of these options will navigate user to corresponding screen |

## Sample Screens

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

Contact list screen Add new contact screen.

## Technical Requirements

1. Use React Navigation
2. Add a search input box to filter the contacts based on the name
3. Use local database for CRUD operations.
4. Each contact in the list should be in swiper item to show edit/delete button when swiped from the right side (<https://www.npmjs.com/package/react-native-swipe-list-view> or any other library)

***Note: Take relevant assumptions during the application development wherever either requirements are not clear to you or the information required is not provided in this case study.***