GitHub Username: surazverma

Train Journey Reminder

Description

This application will remind a user about the upcoming (Indian railway) train journey. All the user have to do is to feed the PNR (Passenger Name Record) number, save the information and the application will remind the user at a desired time which is chosen by user and on the date of journey. Also this application have a widget where the user can see the information about the upcoming journey. The user can check the reservation status by refreshing the reminder. All the information regarding the user and the reservation status is provided by the RailwayAPI.

Intended User

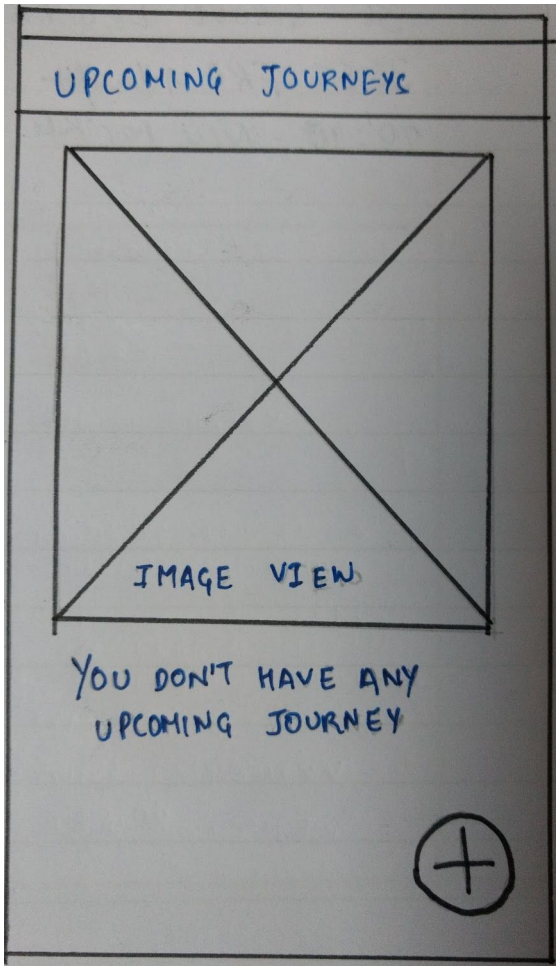
This application is intended for the people who travel from Indian Railway who want to get reminded about their upcoming journey and also wants to check the reservation status.

Features

- Search passenger information
- Save passenger information as a reminder
- Update passenger information
- Get notification reminder about the upcoming journey
- Delete Reminder

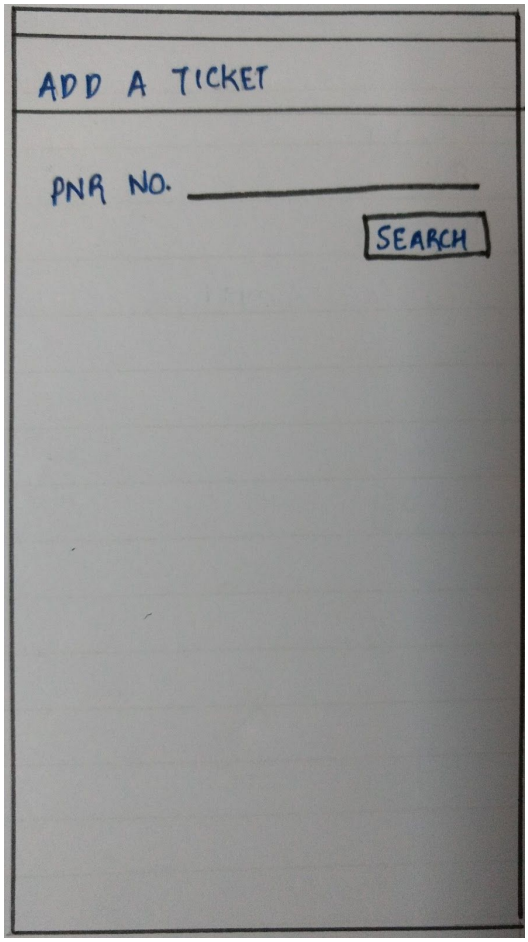
User Interface Mocks

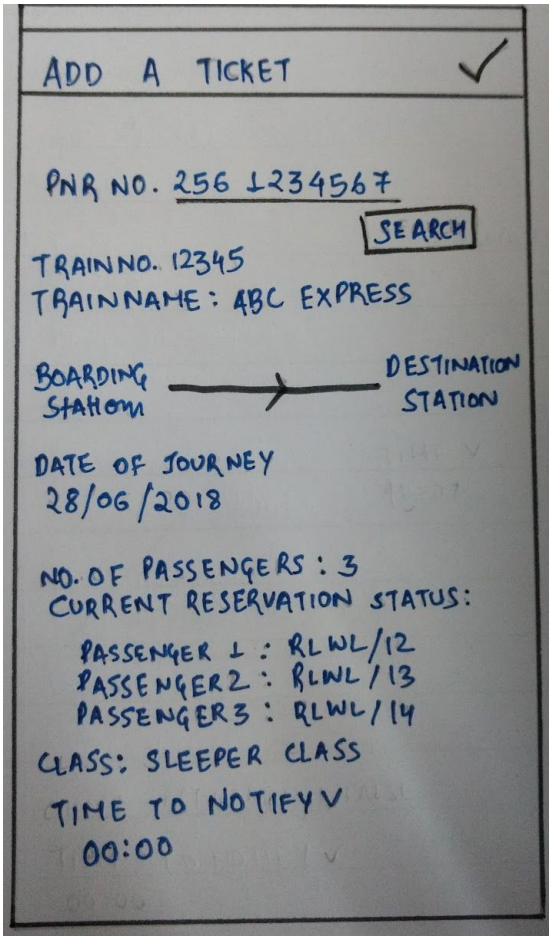
Upcoming Journey Screen (Empty)



This is the starting screen as the user starts the application. This screen represents that there is no upcoming journey and to set a journey reminder the user have to press the floating action button.

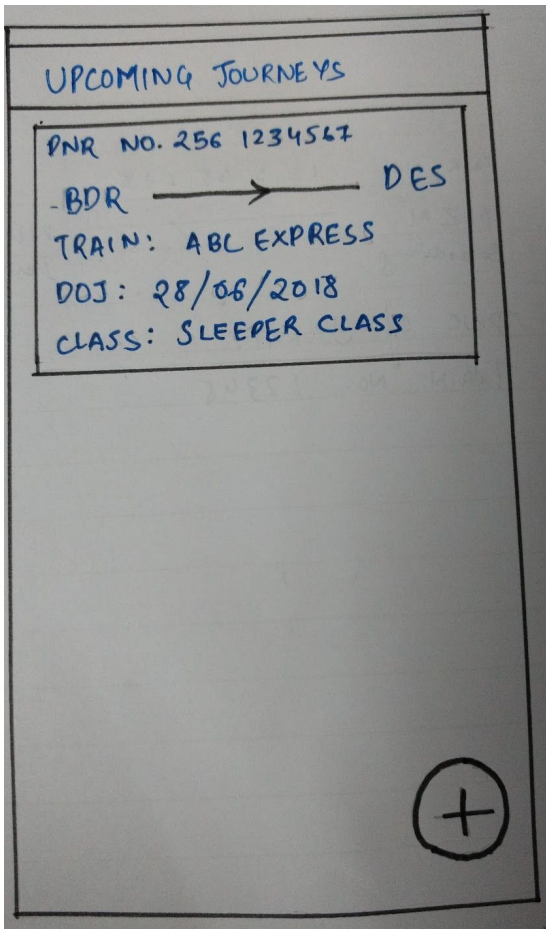
Add A Ticket Screen





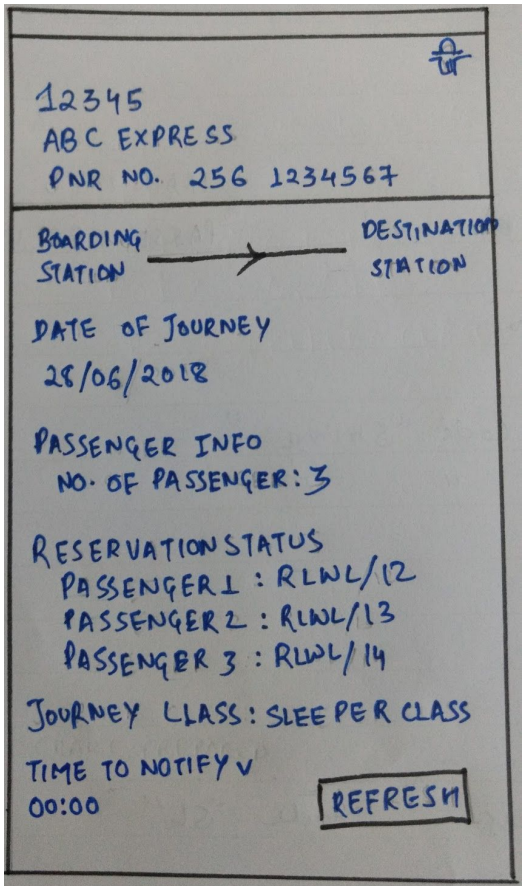
After pressing the floating action button the user will have this screen where the user have to input the PNR() number and press the search button. After pressing the button the screen will update with upcoming journey information as shown in the second image at right. After the data is updated the user have to choose a desired time to be reminded about the journey and the date will be the same as the date of journey. Once the user is done with time and journey information the user can simply save the information by tapping on the tick mark present in the app drawer.

Upcoming Journey Screen (With Reminder)



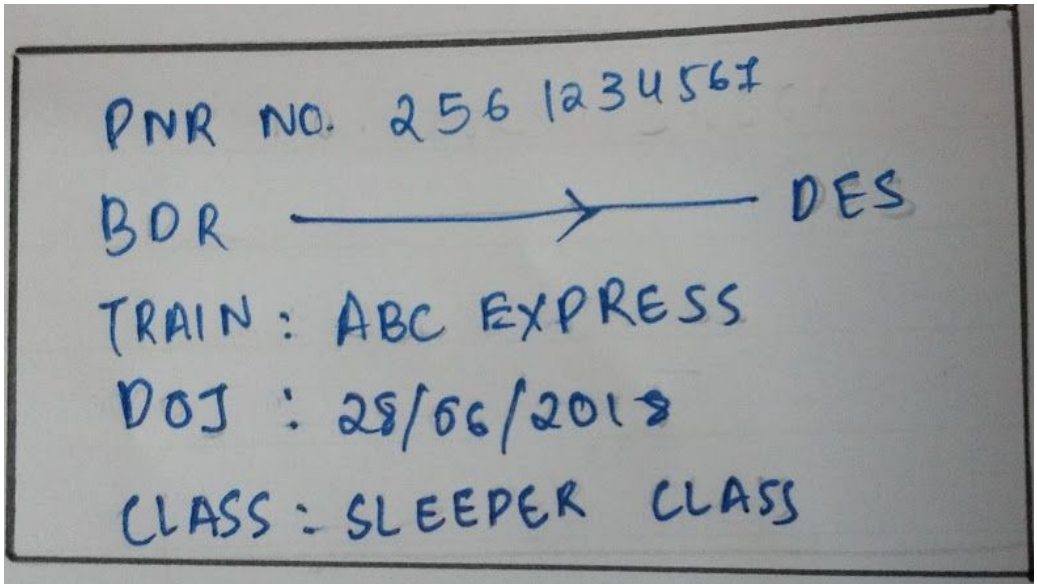
After the user have saved the information about the journey, the user will be back on the Upcoming journey screen with a upcoming journey card which will have some of the journey information.

Ticket Information Screen



When the user Clicks on the journey card another screen will appear with full passenger information. Here the user can update the information by clicking the refresh button also the user can edit the reminder time by clicking the time to notify.

Journey Information widget



This widget will display the small information about the upcoming journey.

Tools and Libraries Version.

This whole application is written solely in java programming language.

- Java

The libraries which are used are :

- Retrofit 2 : v2.4.0
- Butterknife: v8.8.1
- Firebase Crash Report : 16.0.2
- Google Analytics : 16.0.1
- Google mobile ads : 15.0.1

Gradle Version 3.1.3 is used in this application.

Key Considerations

How will your app handle data persistence?

The application will have a content provider which will save and provide information about the train journey.

Describe any edge or corner cases in the UX.

- Orientation changes of device should not affect the application state being in any state.
- Popping up an alert dialogue when a user forget to save the reminder.
- Providing proper error when there is a network error.
-

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

- Butterknife library is used for boilerplate code reduction.
- Firebase is used for the crash reports.
- Retrofit library is used for api network request.

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

- Google admob will be implemented to provide ads on the app
- Firebase Crash Reports and Google analytics to monitor the application.

Tasks

Task 1: Project Setup

Create and setup the project. This step requires these subtasks

- Create the new project in android studio and provide a name to the project
- Add the required libraries to the application.

Task 2: Add the libraries

- Add the Butterknife library to reduce the boilerplate code reduction
- Add the Retrofit 2 library for the API request

Task 3: Data model class

Create a data model class which handle all the data response coming from the RailwayAPI.
Required classes:

- PNR info
- Boarding point
- Reservation upto
- From station
- To station
- Journey Class
- Passenger
- Train info

Task 4: API

Implement an interface for the API which retrieves the data from the Railway API

Task 5: Database Persistence

Add a Content provider and data helper class to handle all the locally stored data in the application.The following steps are required to achieve this

- Add a content provider
- Add a database helper
- Implement the database access function using the loaders and created content provider.

Task 6: Setup the UI

Implement all the necessary fragments and activities
Following UI parts will be created :

- Upcoming Journey activity
    - Setup the empty screen.
    - Add a recycler view.
- Add a ticket screen
    - Edit text field
    - Search button
    - Text View for display
    - Time picker dialogue
- Ticket Info Screen
    - Text View for display information
    - Refresh button
    - Delete button
    - Time picker dialogue
- Widget card

Task 5: Notification Service

Add a broadcast receiver for providing notification about the reminder which the user have set.

Task 6: Manifest

Add the broadcast receiver, parent activity, fragments and the content provider to the manifest file.

Task 7: Google Play services

Add all the google play services mention above that is the firebase crash reports and the Google analytics. Also add the Google admob to the application to display the ads in the application

Task 8: Make application production ready

- Use material design to improve the visuals of the application
- Provide RTL support to the layout
- Plan for localization by moving all the strings to strings.xml

Task 9 : Prepare for the release

Create app icon and Remove any debug messages