GitHub Username: DhayanandBaskar

Hackathon App

Description

The Hackathon App is a one stop shop for all the upcoming hackathons that are about to take place. Its helps search for hackathons, have a detail view about it and also lets you bookmark them. You can share the upcoming hackathon event to your friends. You can have a widget for all the upcoming events.

Intended User

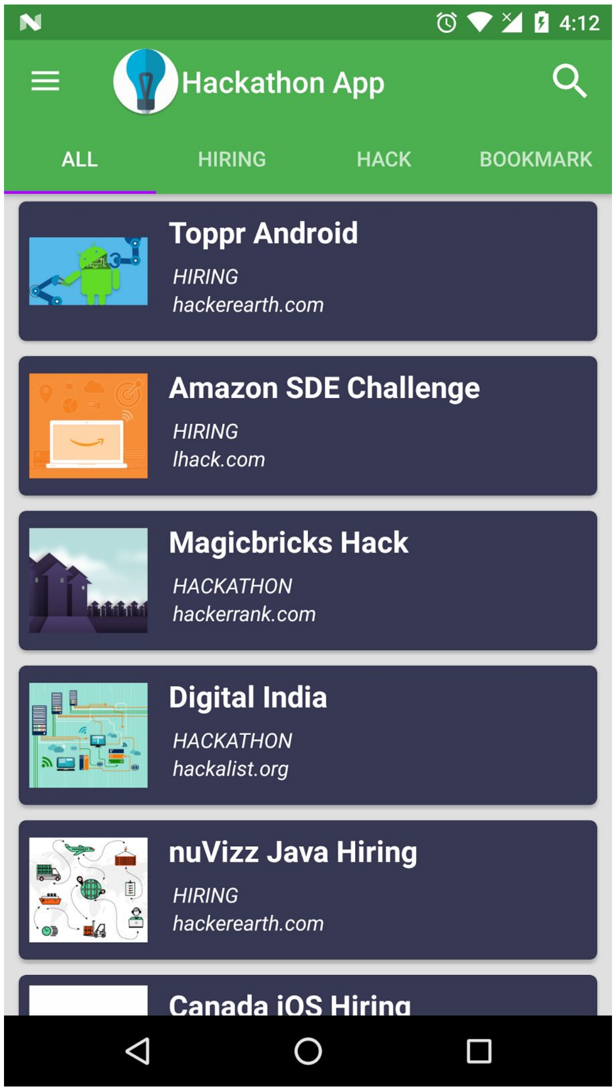
Students, Programmers, Software Developers

Features

- Sort event into tabs based on category and bookmark
- Bookmark events
- Detail Widget for the home screen
- Custom search suggestions
- User Login
- Share events with friends
- Works offline even in places where the user can't access internet

User Interface Mocks

Screen 1 (Main Activity)



4:12

Hackathon App

ALL HIRING HACK BOOKMARK

Toppr Android
HIRING
hackerearth.com

Amazon SDE Challenge
HIRING
lhack.com

Magicbricks Hack
HACKATHON
hackerrank.com

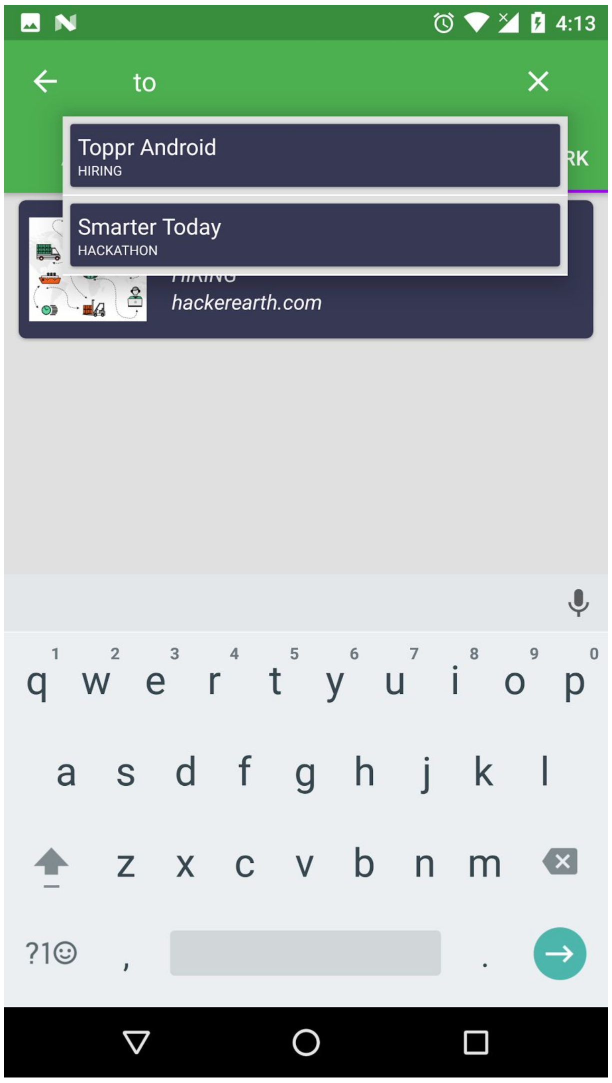
Digital India
HACKATHON
hackalist.org

nuVizz Java Hiring
HIRING
hackerearth.com

Canada iOS Hiring

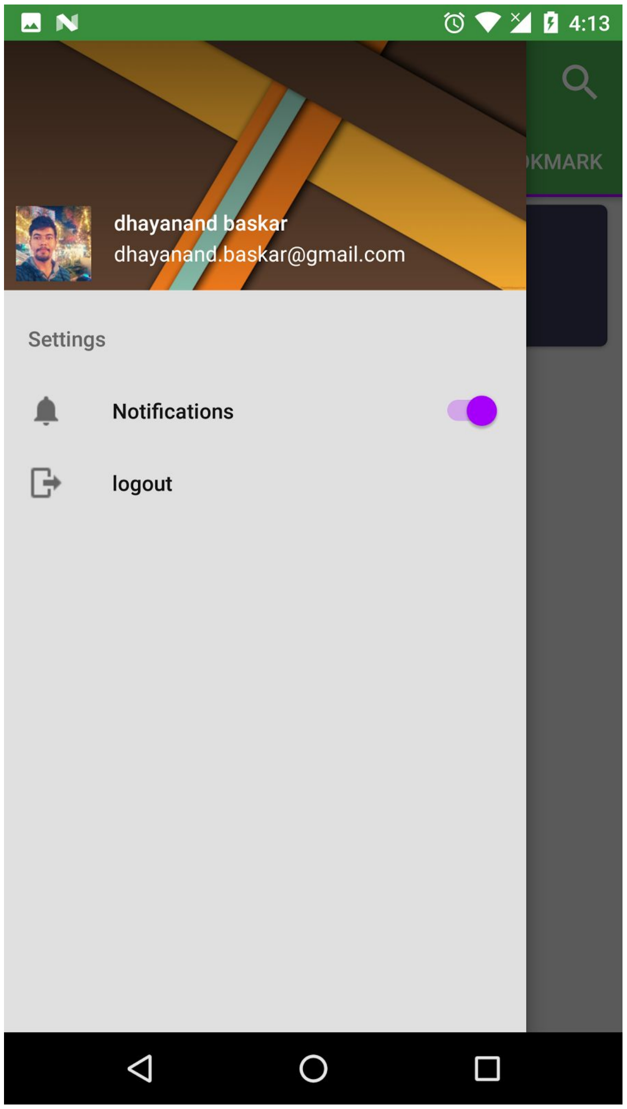
This is the main view when the user opens the app, the screen shows the hackathon event that are about to take place. The user can click on them to go to the details view.

Screen 2 (Custom Search suggestion)



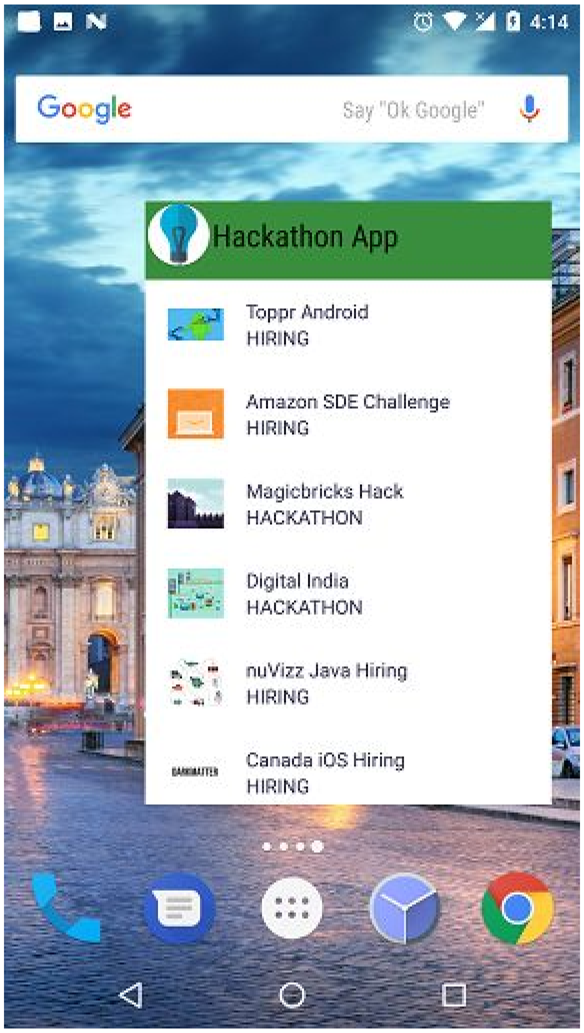
The user can search for event sand when clicking on them it directly takes the user to the details view.

Screen 3 (Navigation view)



The navigation view displays user info and also lets user switch on and off notifications and the sign out option is also available here.

Screen 4 (widget)



Hackathon App

Toppr Android
HIRING

Amazon SDE Challenge
HIRING

Magicbricks Hack
HACKATHON

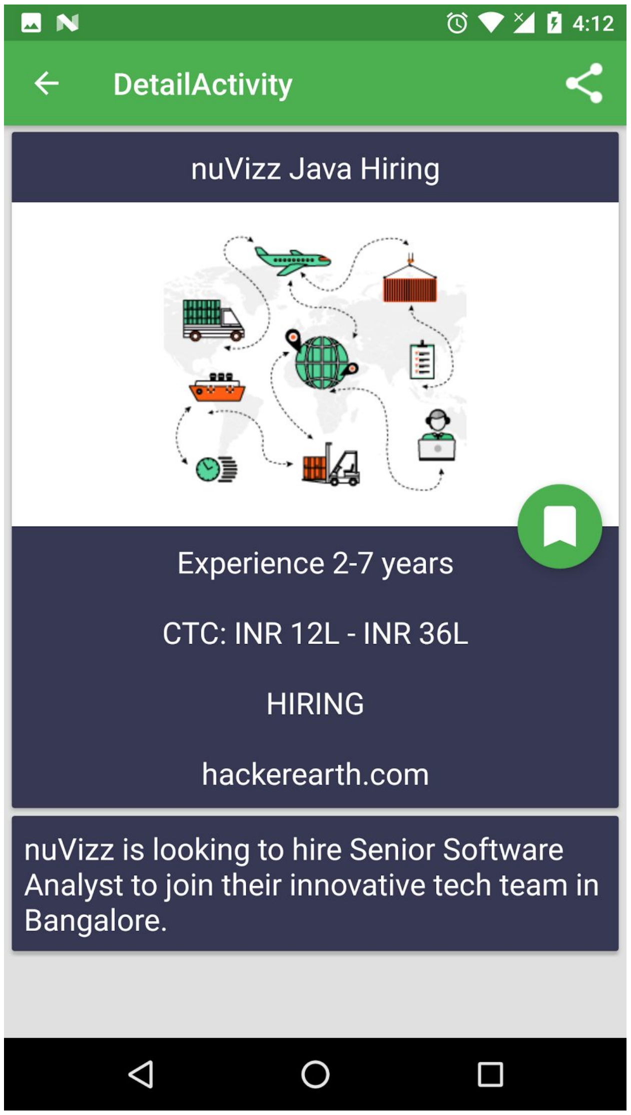
Digital India
HACKATHON

nuVizz Java Hiring
HIRING

Canada iOS Hiring
HIRING

Users can have a quick access widget which lets the users directly land on the details view by clicking on the app and into the app by clicking on the title.

Screen 5 (Details view)



nuVizz Java Hiring

Experience 2-7 years

CTC: INR 12L - INR 36L

HIRING

hackerearth.com

nuVizz is looking to hire Senior Software Analyst to join their innovative tech team in Bangalore.

The Details view provides detailed description and information about the event and the user can click on the Bookmark FAB button to book mark the event and also the user can share events. The bookmarked events will appear in the bookmarks tab.

Key Considerations

How will your app handle data persistence?

Events data will come from an external API. The app will save some data internally using SQLite and a Content Provider.

Describe any corner cases in the UX.

When the user scrolls to an event and presses the event to view the detail view, the position in the recyclerview is retained and then the user presses back button the app returns to the exact position where the user left by saving and restoring the view.

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

Picasso (http://square.github.io/picasso/) : for loading images from the network in the background thread, used both in the main Activity and details activity.

Retrofit (http://square.github.io/retrofit) : for fetching the Json object from the network, the Json is used to populate the view with information and storing it in the content provider.

Okhttp and retrofit gson converter are used since retrofit depends on them for its operations for fetching and converting the Json to model objects.

Butterknife (http://jakewharton.github.io/butterknife) : is used for binding views with their objects instead of doing it manually and also presents a cleaner look to the code.

Firebase core is used for firebase analytics.

Firebase auth and ui auth for user authentication

Firebase messaging for push notifications

Glide for getting the bitmap images from the network to set into widgets.

Describe how you will implement Google Play Services.

I am implementing Play services using Firebase. Features such as:

- Identity authentication using firebase auth and providing a navigation view for the user with basic information and profile pic populated.
- Firebase Messaging for push notification using firebase console
- Firebase core is used for analytics

Next Steps: Required Tasks

Task 1: Project Setup

- Set up libraries
- Structure the project in packages: view, data, model, rest, etc. Separate pure Java classes in separate packages

Task 2:Gather data using custom API

We need a way to gather all the event data from an external API.

- Write a helper class using Retrofit
- Use a Service (that can be bind to an activity) to do the job

Task 3: Implement UI for the Search Activity

- Build UI for the Search Activity.
- Handle the logic custom Search suggestion with the help of the data from content provider

Task 4: Implement UI for the main Activity and the Fragments

- Build UI for the MainActivity & Following Tabs
- Build UI for the Detail activity
- Build UI for the Navigation view

Task 5: Data persistence

We need to store the following:

- Basic information regarding Hackathons
- BookMarked Events

Following, Pending or Seen Tasks:

- Finish to decide the design of the database.Keep in mind that only what we store will be available when we are offline
- Create the database and the Content Provider using Schematic
- Use Loaders to access the data

Task 6: Pull data

- Implement a SyncAdapter to update stored information, if necessary pulling data daily

Task 7: Collection Widget for Following Shows

- Create a widget that shows a list all events
- Add intents to the events hence when clicked takes the user directly to the details view

Task 8: Implement Fire Base

- Track crashes and other stuff Analytics give us for free
- Track if users uses Notifications (if they are implemented)
- Track if users shares something (if it is implemented)

Task 9: Notifications

- Periodic notifications when the new data is sync from the api.
- Push notifications using firebase messaging.

Task 10: Share Action

- Feature for the user to share the event information.