GitHub Username: dilpreet96

Devents

Description

Devents helps a user to get notified about any tech related news or about any technical event near him.This app also provides a platform to promote your technical event (hosted on facebook) with a single click ,all the developers or programmers will be notified about any new event added.

Intended User

Devents is meant for all the developers who want to stay up to date about any technical news and about all the events or workshops which will be happening near them.

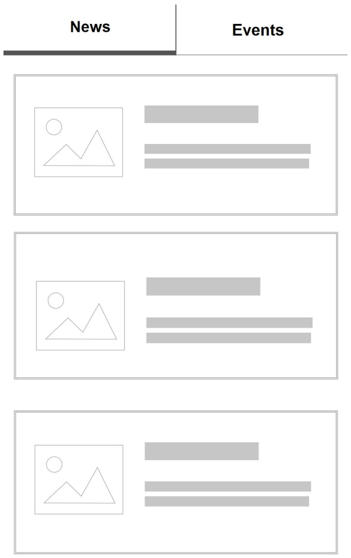
Features

Features of Devents :

- Shows Tech news from Hacker News
- Shows nearby Tech events with complete information
- Share event just by copying it’s facebook link
- Reminder about any upcoming event
- Share event by manually adding information
- Details about an event

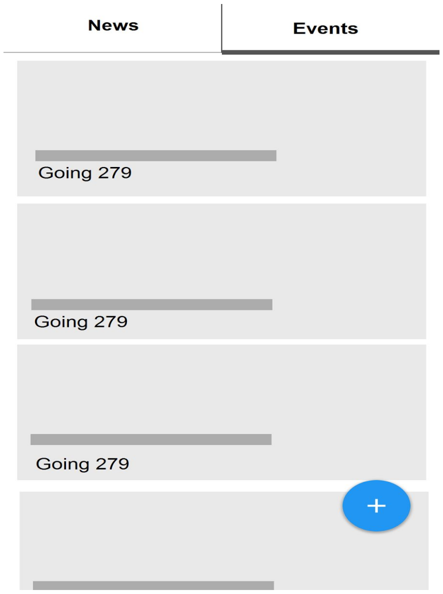
User Interface Mocks

Screen 1



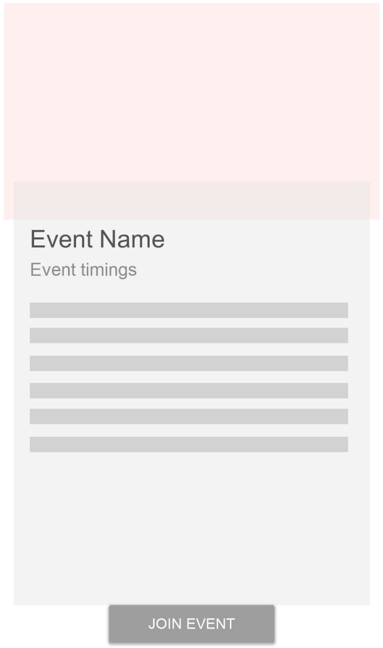
This screen show all the technical news fetched from News API .

Screen 2



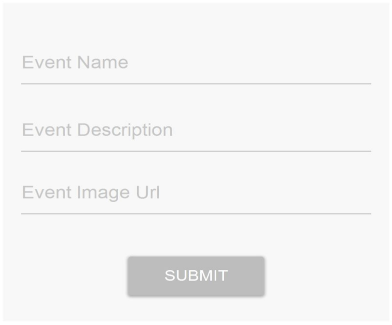
This will contain all the events which are near to user and this is updated when user recives a push notification ,the app will check events location and user's location and notify accordingly

Screen 3



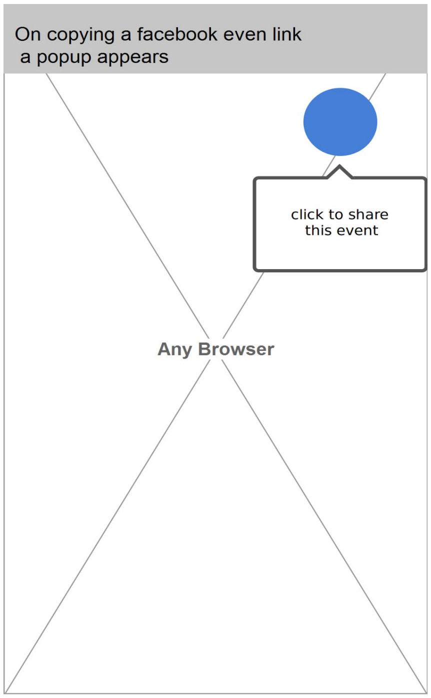
This screen will display complete information about that particular event along with a button which will take the user directly to the event signup page.

Screen 4



This screen allows user to manually add an event to share it among other users whose complete info will be sent to my database using my api and sent to every device as a push notification.

Screen 5



Making a service using clipboard manager to get fb event link and allowing user to share the event using 1 click.

Key Considerations

How will your app handle data persistence?

Content providers would be used in order to provide an offline support by caching the events and news.

Describe any corner cases in the UX.

User can swipe right to see nearby events and click on any event to get more information about it.

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

- Glide :- For image downloading and caching
- Retrofit :- For connecting to REST Api's
- Butterknife :- For avoiding boilerplate code.

Describe how you will implement Google Play Services.

Google play service will be used for implementing google login and google Admob to show advertisements in application

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

-Create project in android studio
-Adding all the required libraries in the gradle

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for MainActivity
- Build UI for NewsListFragment
- Build UI for EventsListFragment
- Build UI for EventsAddActivity
- Build UI for LoginActivity
- Build UI for EventsInfoScreen

Task 3: Implementing News API and Content Providers

-Implementing News API to fetch Tech news peroidically using Broadcast Receivers and Alarm Manager or the news may be fetched pushed using fcm
-Implementing Content Providers and Loaders

Task 4:Implementing Google Login,Google Location API ,FCM and API's for Events

-Implementing Location API to fetch users location for displaying events near the user
-Making django api for storing all events on a server and making apis to interact with it
-Implementing a Google login for those who want to share an event
-Setting up FCM for push notifications about new events added
-Storing events info in a local storage for offline support
-Implementing events info activity

Task 5: Implementing ClipBoardManager and Manual Entry

-Implementing clipboard manager to automatically get a facebook event link and allowing the user to share the event just with a single click .
-Allowing user to manually add about an event they want to share with other developers

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"