
GitHub Username: github.com/maps-apps-n

EpInfo

Description

This app displays information on mosquito-borne diseases, symptoms, and how to avoid them. The app's widget displays news on possible risk areas.

Intended User

Students, health professionals, travelers, and individuals who want to stay informed and up-to-date.

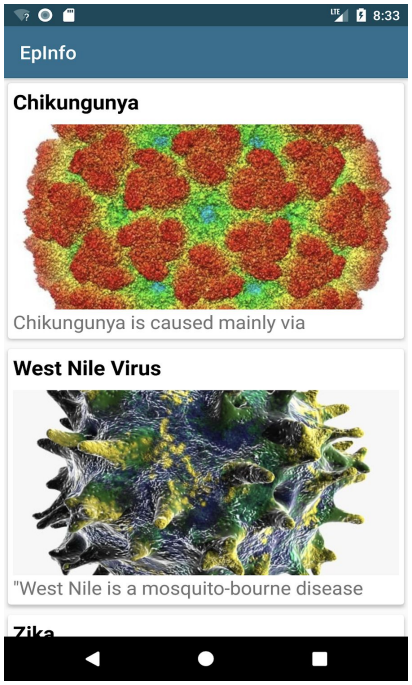
Features

- Displays Information.
- Displays news alerts via app widget.

User Interface Mocks

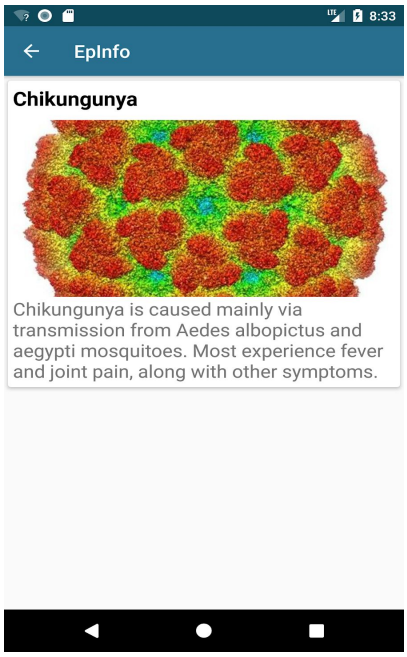
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1

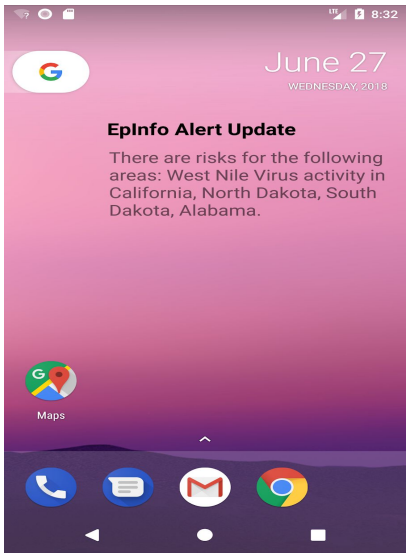


Home Screen featuring a RecyclerView containing CardViews with images and a preview of each description.

Screen 2



Details class displays a details page with the image and the rest of the description.



AppWidget displays alerts with updates regarding disease risk locations.

Key Considerations

How will your app handle data persistence?

This app uses the Firebase Realtime Database to call for titles, images, and descriptions. AsyncTask is used for data requests.

Describe any edge or corner cases in the UX.

User can navigate back to the home screen from the details page using the back arrow in the app bar.

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

This app uses Picasso to handle loading of images
('com.squareup.picasso:picasso:2.71828').
Firebase is used for the database
('com.google.firebase:firebase-database:11.8.0', 'com.firebaseui:firebase-ui-database:0.4.0').
RecyclerView and Cardview are used for loading and displaying of data.
('com.android.support:recyclerview-v7:1.',
'com.android.support:cardview-v7:27.1.1')

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

Firebase is used to handle data from the Realtime Database.

Next Steps: Required Tasks

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

Task 1: Project Setup

Tasks to complete:

- Add permissions to manifest.
- Add dependencies for Firebase, Picasso, CardView and RecyclerView to gradle.

Task 2: Implement UI for Each Activity and Fragment

- Add RecyclerView, CardView, and details activity layouts.
- Add ViewHolder, Model, and DetailsActivity.java

Task 3: Your Next Task

- Setup Realtime Database on the Firebase console.
- Include images and descriptions in the Data file within the console.

Task 4: Your Next Task

- Add Widget java class and layout.