

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1. Scan the ticket](#)

[Screen 2. Ticket is found](#)

[Screen 3. Ticket isn't found](#)

[Screen 4. Navigation](#)

[Screen 5. History](#)

[Screen 6. History details](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Handle error cases](#)

GitHub Username: volnyansky

Ticket checker app

Description

There is Vedanta yoga center in my city. They organize different yoga trainings and other programs. This app will help them to verify tickets which was sold online.

Intended User

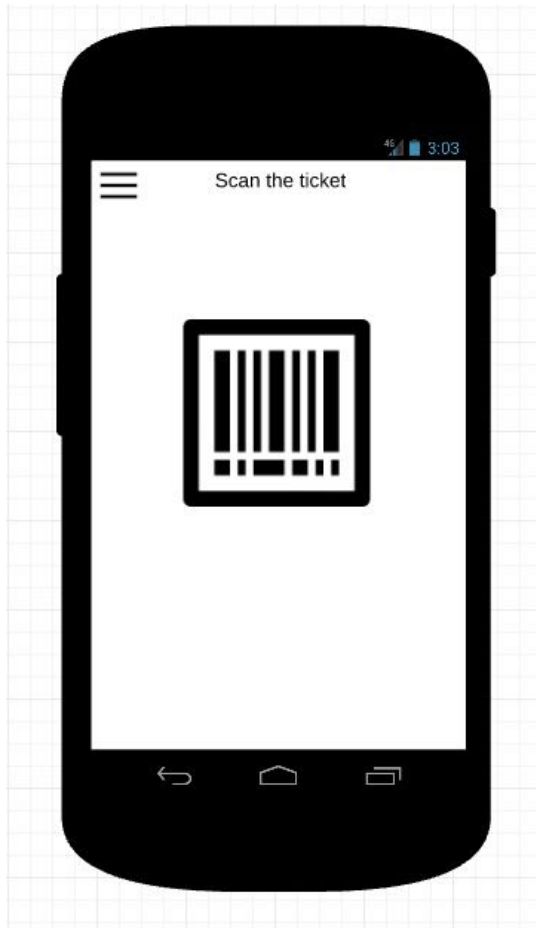
Event support staff.

Features

- Scans barcodes from tickets
- Saves info on server
- Saves scan history on device

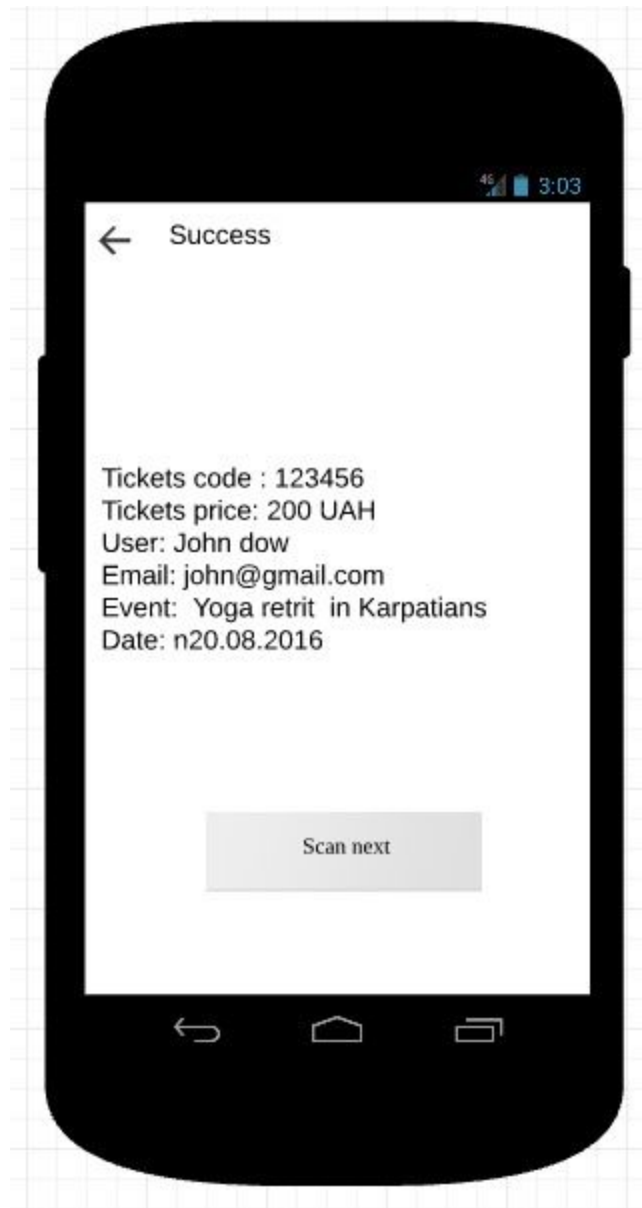
User Interface Mocks

Screen 1. Scan the ticket



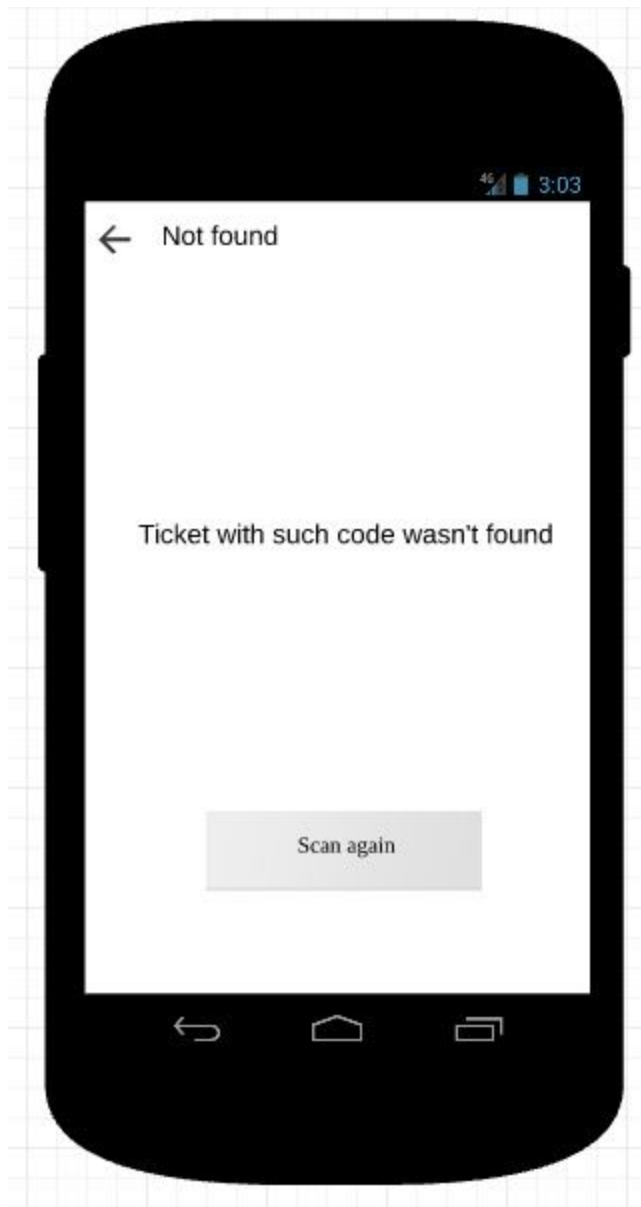
When user start app he can start scan tickets. He should place ticket barcode in front of phone camera.

Screen 2. Ticket is found



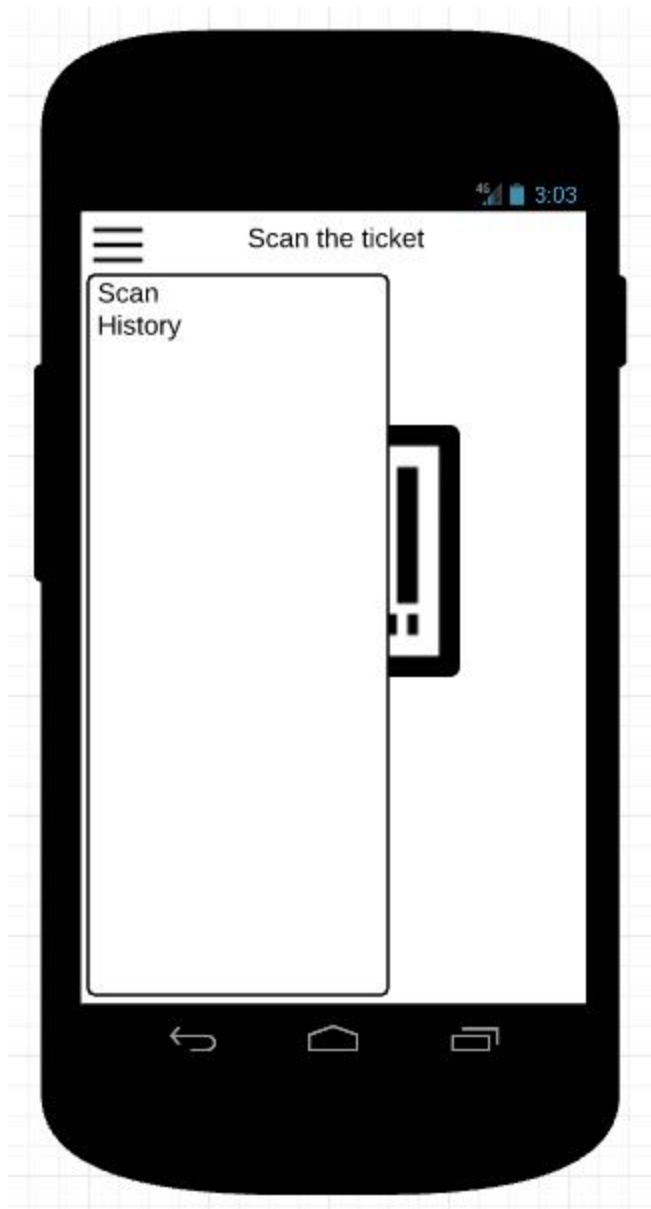
If ticket with given barcode was found then user is taken ticket info screen.

Screen 3. Ticket isn't found



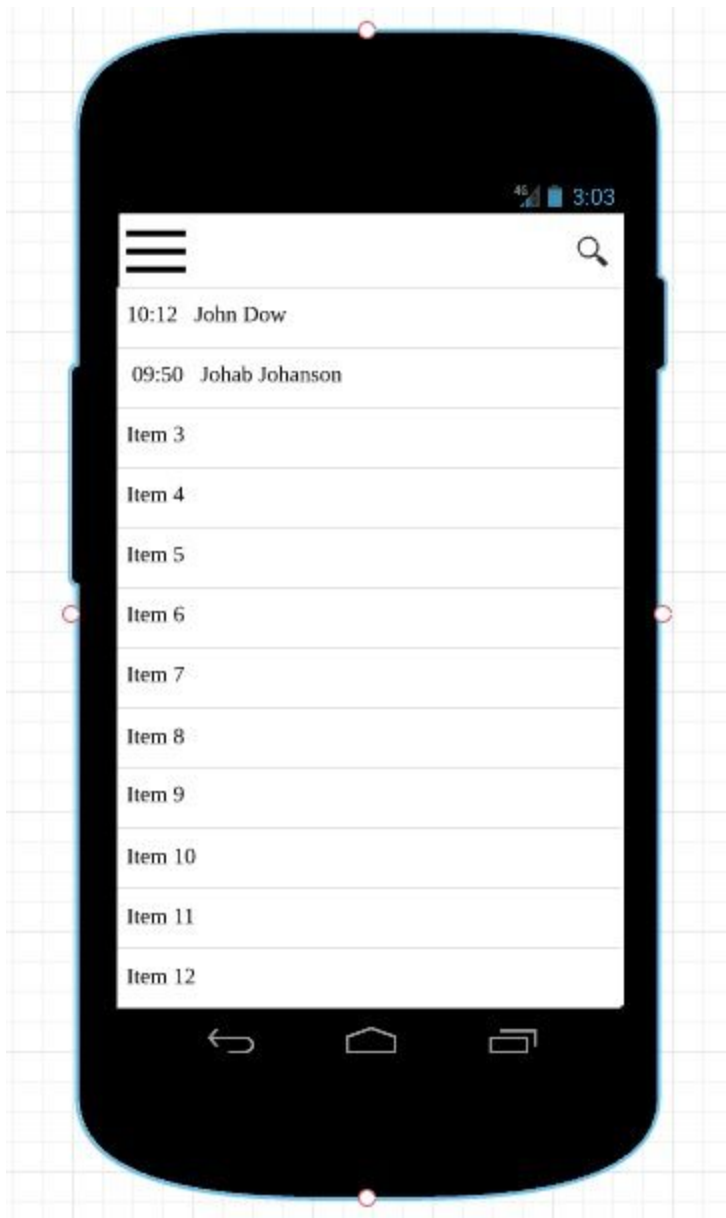
If ticket is not found in db then app shows warning to user.

Screen 4. Navigation



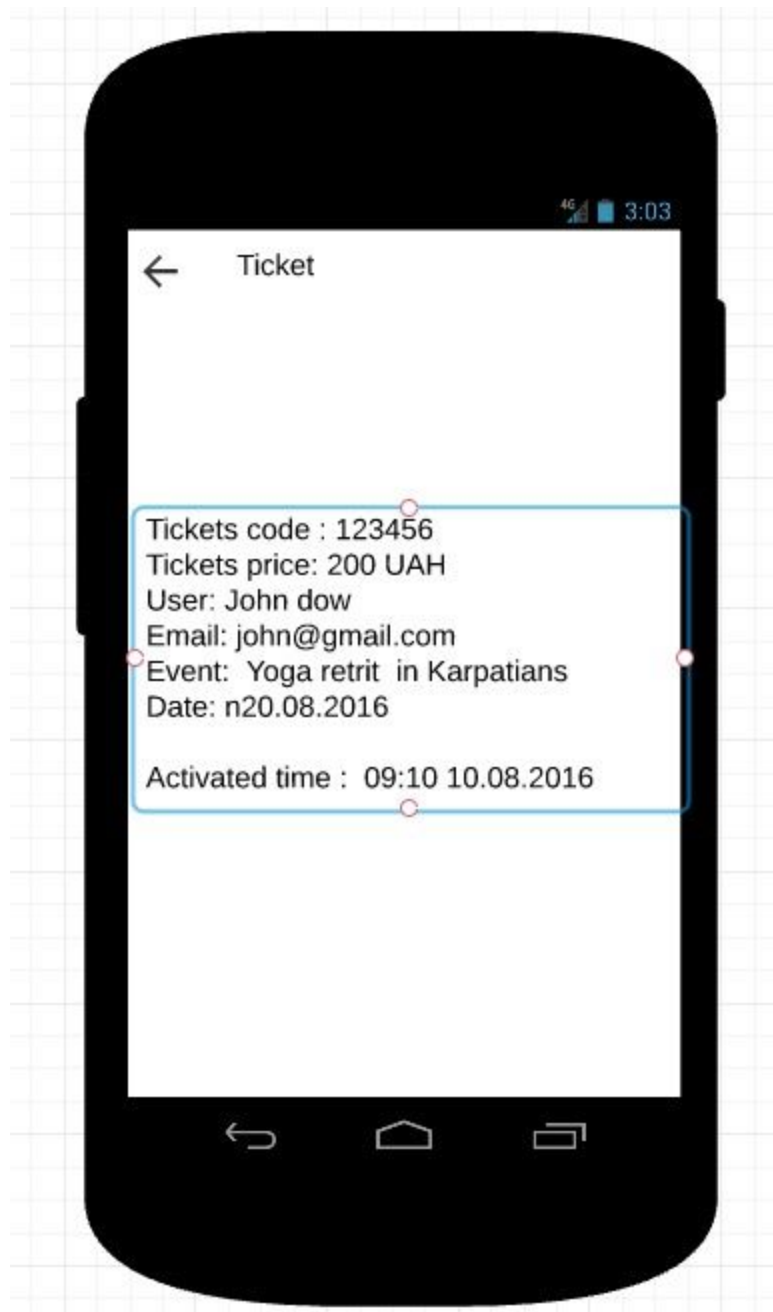
There are 2 main screen in app - scan screen and history screen . App uses navigation drawer to switch between them.

Screen 5. History



History screen contains info about scanned

Screen 6. History details



When user taps on history item, he goes to the details screen. This screen almost the same like screen 2 except activation time.

Key Considerations

How will your app handle data persistence?

App uses local db to store data for verified tickets, and it uses content provider to get this data.

Describe any corner cases in the UX.

App uses navigation drawer to switch between scan and history screen.

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

The app will use following libs:

- Retrofit - data exchange between app and server
- Google play services - barcode scanning
- Greendao - local db and content provider.
- Android data binding - display info and handle ui events

Next Steps: Required Tasks

Task 1: Project Setup

- Add ticket generation component to the server side.
https://www.drupal.org/project/commerce_event_ticket
- Create a test event and assign tickets to it

Task 2: Implement UI for Each Activity and Fragment

- Build UI main activity (navigation drawer + fragment container)
- Build UI for Scan fragment
- Build UI for ticket found activity
- Build UI for Ticket not found activity
- Build UI for history fragment
- Build UI for history details activity
- Implement REST API client
- Implement local db and content provider

Task 3: Handle error cases

- Handle internet connection errors
- Handle app foreground/background transitions issues

