GitHub Username: DINESHKARPE

TurnByTurn

Description

TurnByTurn is location based app which tell real-time location & do the various action for the following stakeholders
Parent get real time location of their child school bus.
Parent get real time update of their child for get up into the school bus & safety majors like hash break, over speeding, breaking geo-fence.
Parent have in app facility to send push notification to driver about child status (Coming or Not)

Driver get navigation for pick & drop. In emergency driver have SOS facility which informs emergency to school as well as parents in real time. Driver or Bus owner have in app invoicing facility.

Intended User

Bus Driver
Parent

Features

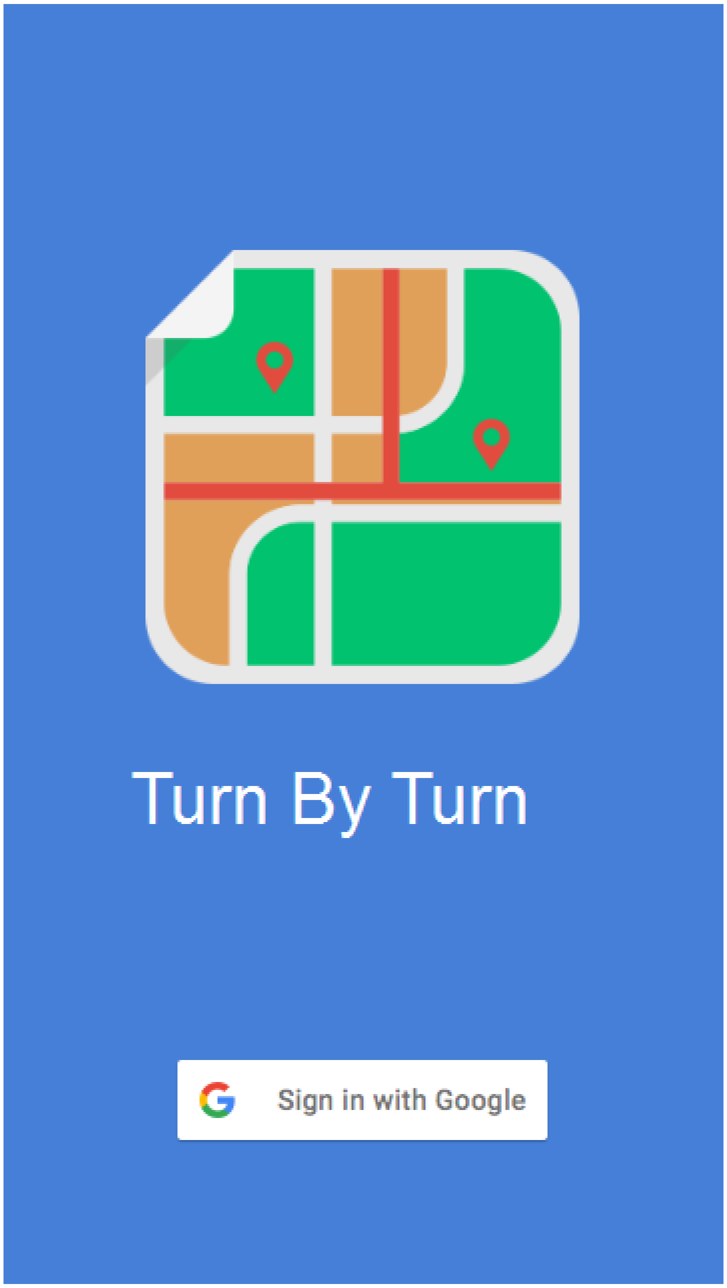
List the main features:

- Parents get real time update their child.
- Parent get alert before pick & drop (when bus is near to stop).
- Parent send notification about child status (Coming or Not)

- Driver get navigation for pick & drop.
- Driver create SOS alert any emergency.
- Driver manage all invoicing.

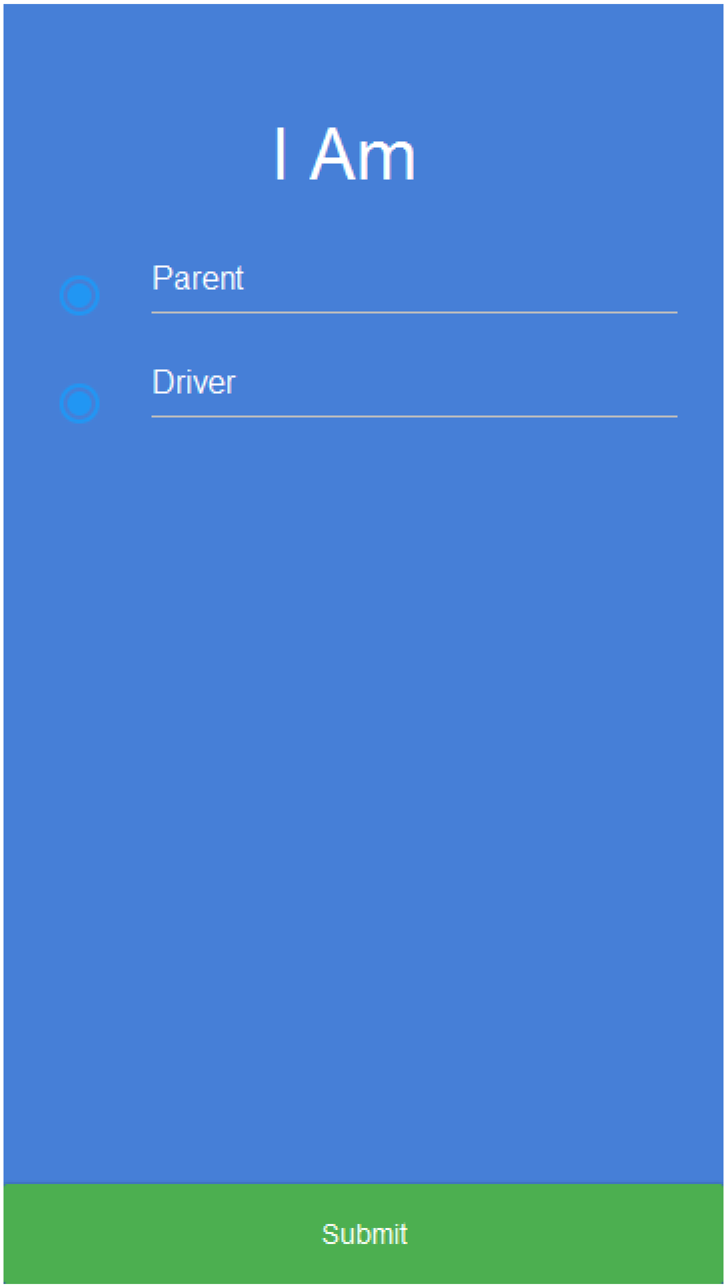
User Interface Mocks

Screen 1



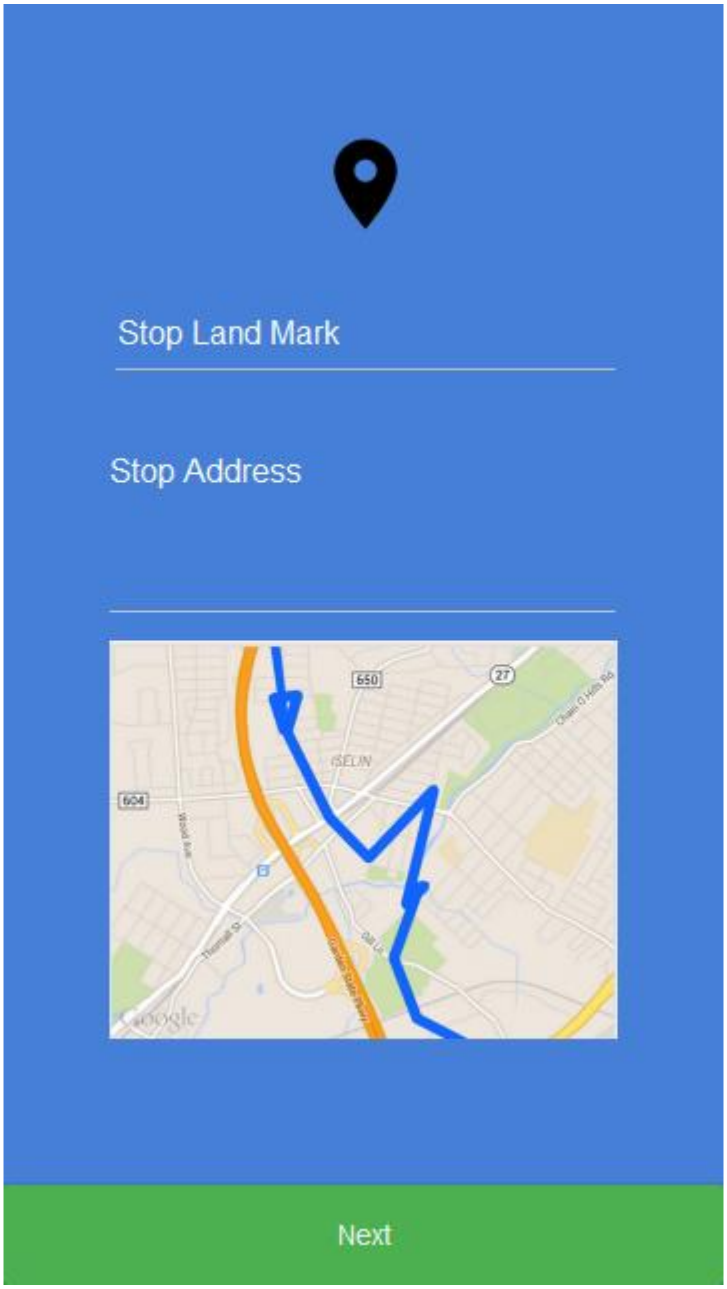
Home screen, Sign In Using Google Account.

Screen 2



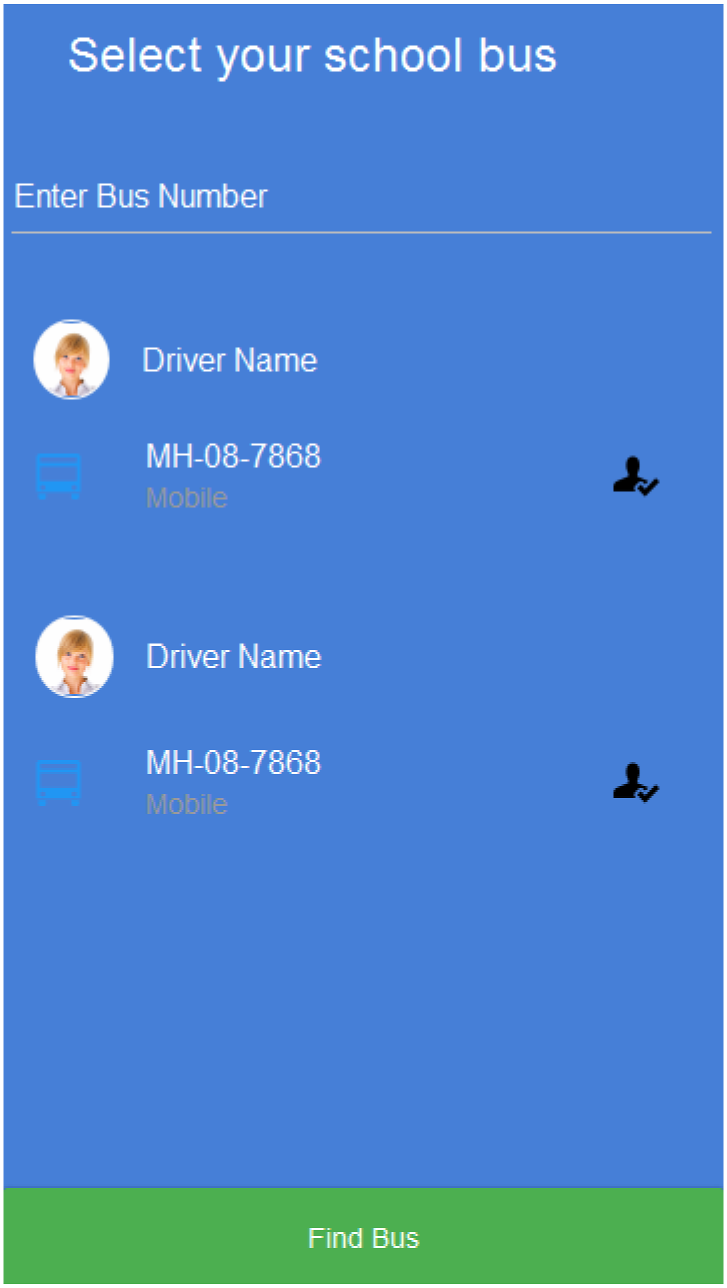
Profile Type Selection

Screen 3



Parent Location Auto Load here & parent put landmark

Screen 4



Select your school bus

Enter Bus Number

Driver Name

MH-08-7868

Mobile

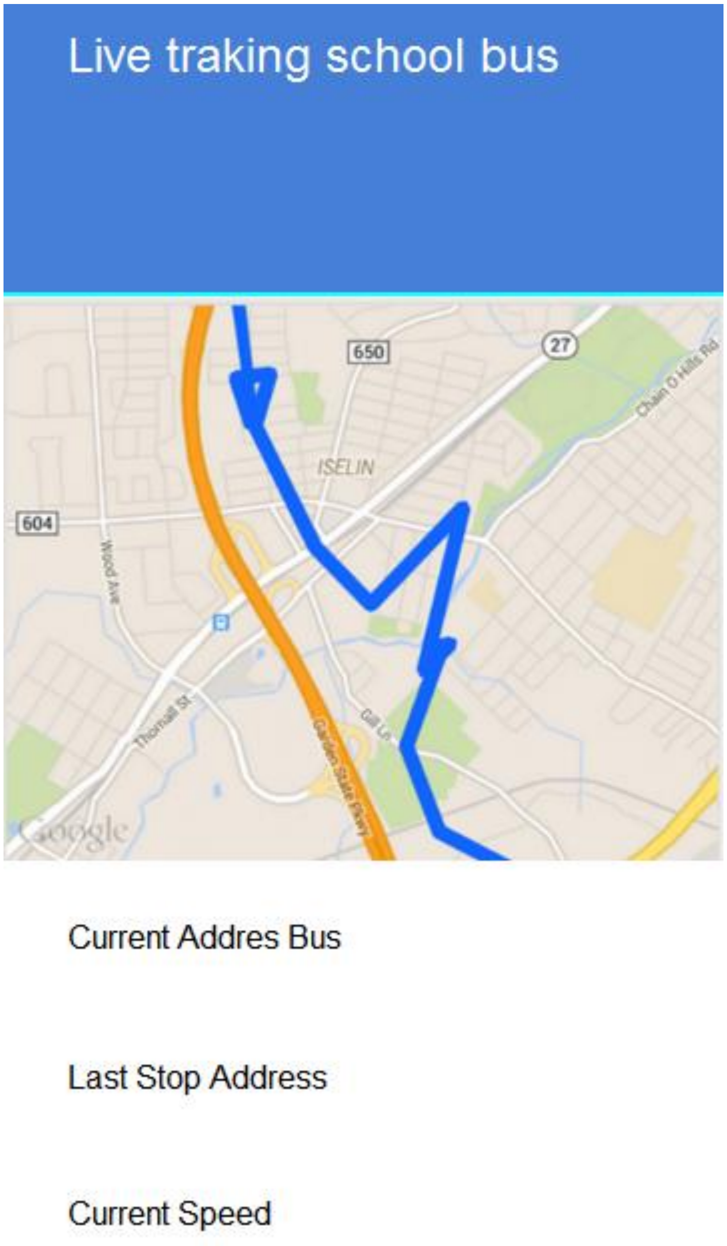
Driver Name

MH-08-7868

Mobile

Parent Search school bus using bus number & send request to Driver as subscriber

Screen 5



Live traking school bus

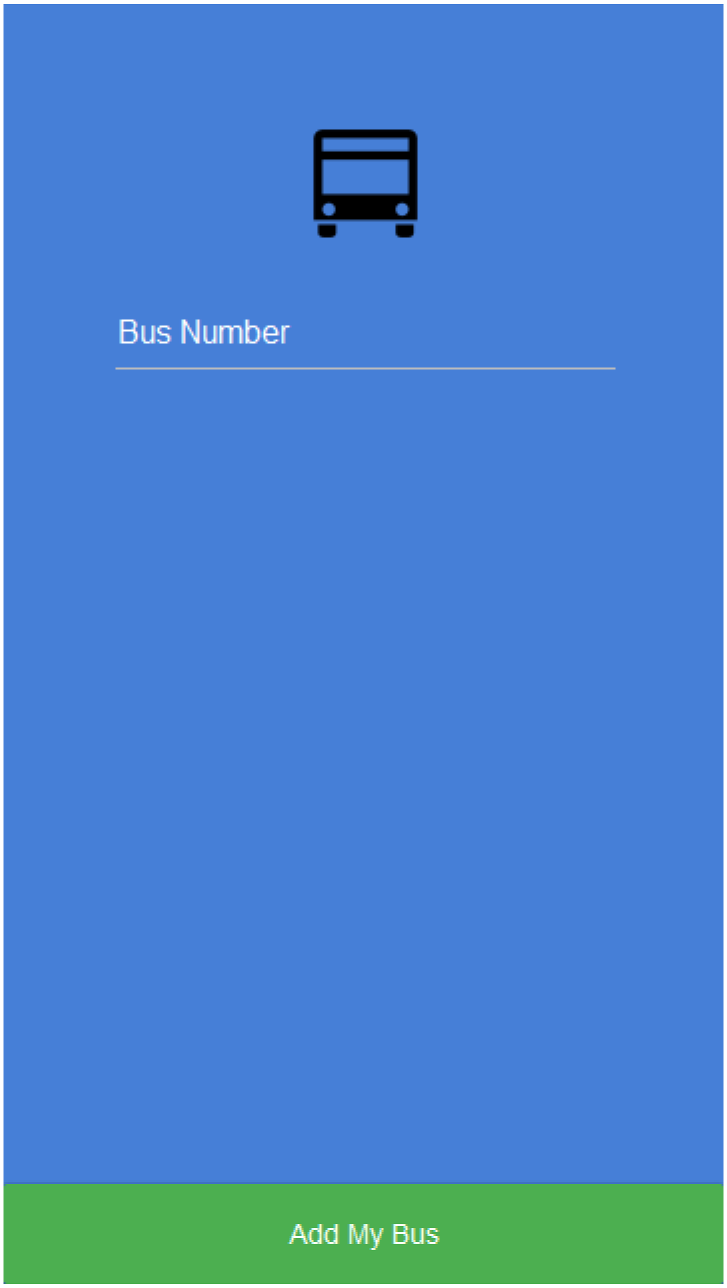
Current Addres Bus

Last Stop Address

Current Speed

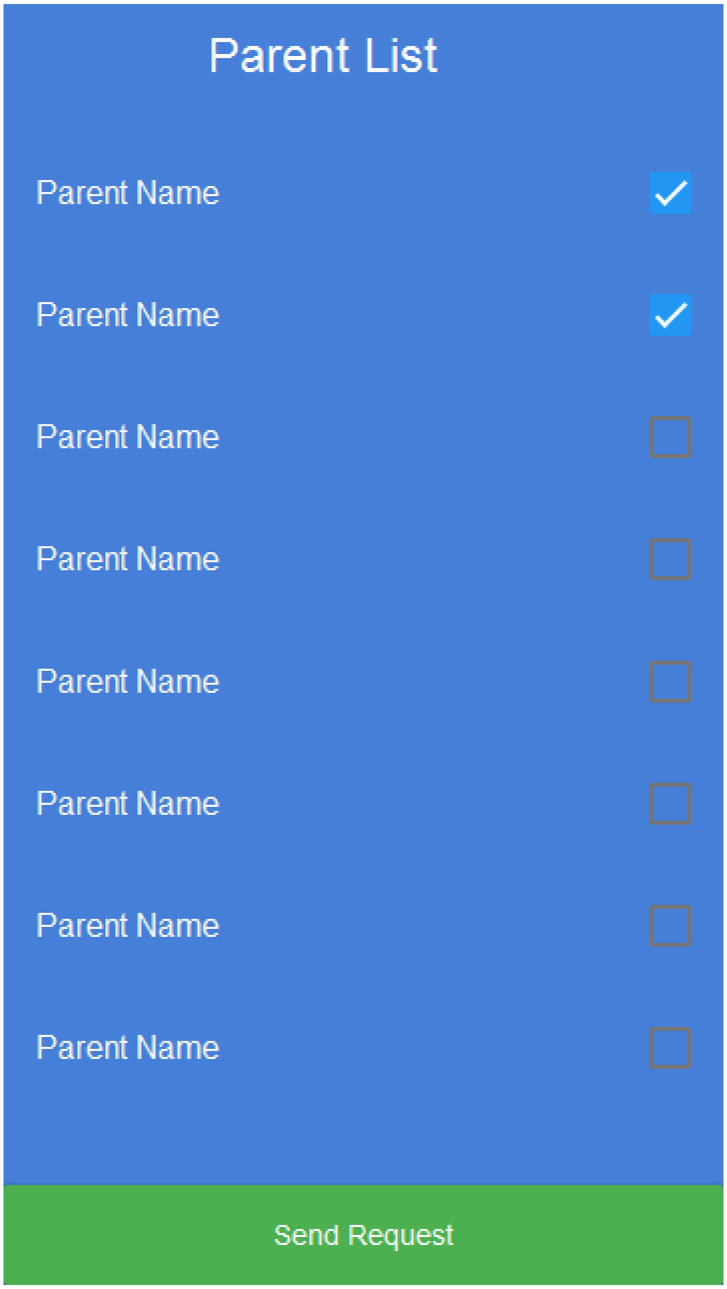
School Bus Live Location for parent

Screen 6



Here Driver Add own Bus using Bus Number

Screen 7


Parent List

Parent Name ☑

Parent Name ☑

Parent Name □

Parent Name □

Parent Name □

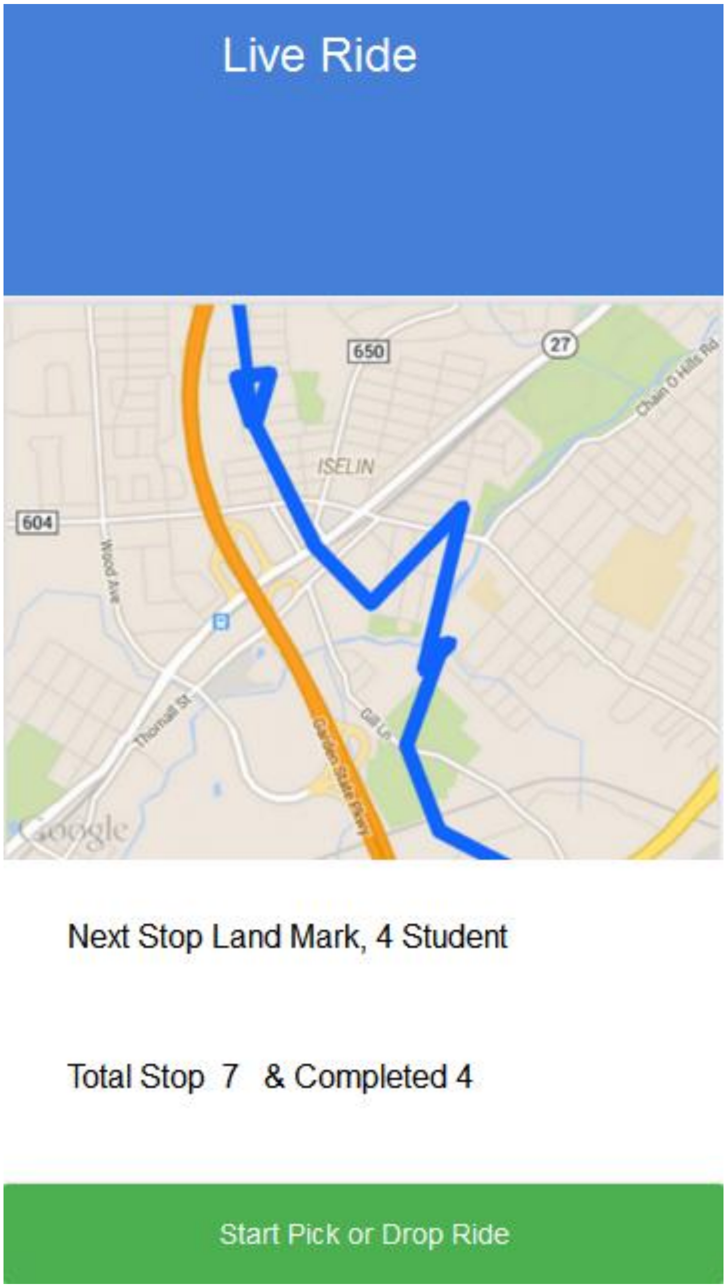
Parent Name □

Parent Name □

Parent Name □

Load All Contact using contact API & Driver send invitation request

Screen 8



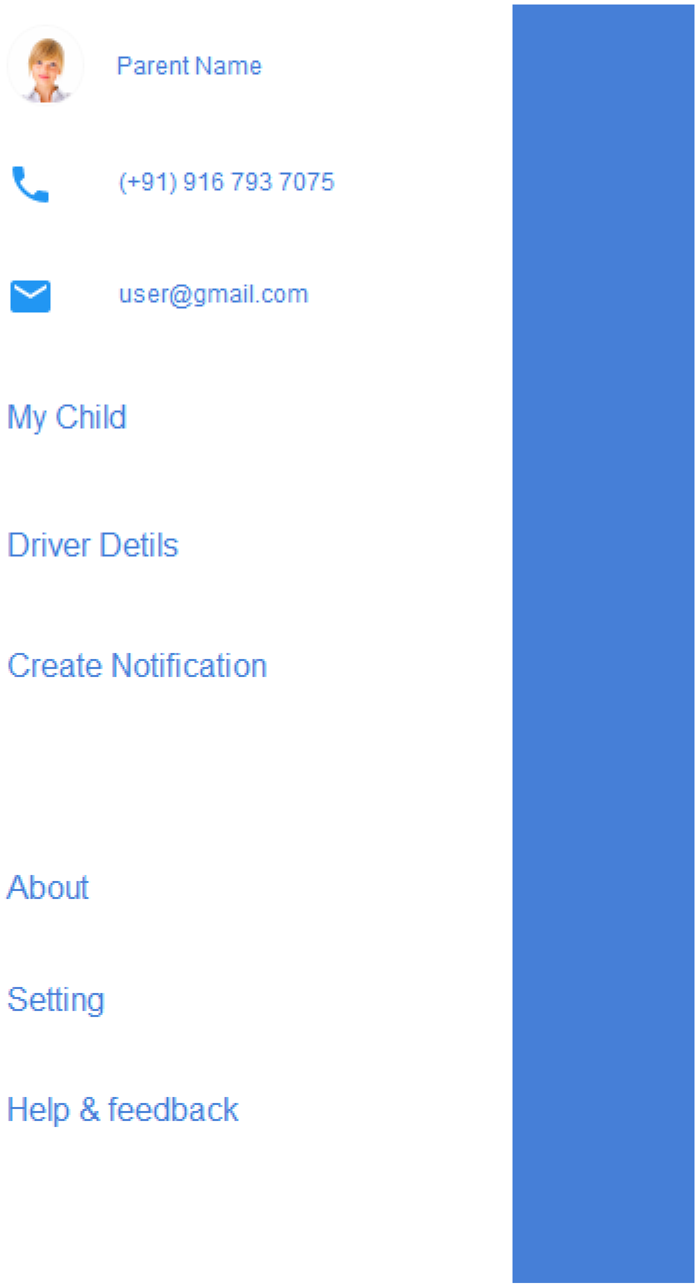
Live Ride

Next Stop Land Mark, 4 Student

Total Stop 7 & Completed 4

Here Driver Start & Stop Pick and Drop ride.

Screen 9



Parent Name

☎ (+91) 916 793 7075

✉ user@gmail.com

Navigation Drawer with Navigation Items for exploring the different features of the app & User Profile

Key Considerations

How will your app handle data persistence?

All Data will be on server & in app SQLite will be use. Data sync event based with server according user interaction with app.
With in App Data will be access using content provider. Each user access only own Data.

All pick & drop location data will be uploaded to server & according driver fetch all data & store into SQLite database using Content Provider
Describe how your app with handle data. (For example, will you build a Content Provider or connect to an existing one?)

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

com.github.bumptech.glide:glide:3.6.1'
Retrofit:retrofit
Retrofit:converter-gson:2.0.0-beta2
Butterknife

Google Vision API
Google Navigation API
Google Play Services for Analytics, Location API etc.

Next Steps: Required Tasks

Task 1: Project Setup

- Create a new project
- Configure libraries
- Add models for data

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Main Activity
- Build UI for Parents Account
- Build UI for Driver Account

Task 3: Implement tasks associated with backend

- Add Parent, Driver Sing in & Profile Create features
- Implement Navigation API & Vision API
- Implement SOS, Add Stops Features
- Implement AsyncTask for real time data update.
- Implement SyncAdapter for Day to Day Data update with Server and app

Task 4: Handle exceptions and improve UI

- Handle exceptions and add progress bars wherever required
- Implement widget For Bus Driver next ride schedule, one widget for parent regarding bus status
- Add Activity transition animations for lollipop and above
- Add runtime user permissions for Android M and above
- Add licenses and terms

Task 5: Make Application compatible

- Implement Material design elements
- Check string resources for localization
- Check for right to left alignment

Task 6: Get feedback from code reviewer and friends

- Get feedback from the code reviewer and friends
- Implement suggested feedback
- Publish on Google Play Store