GitHub Username: cingh-jasdeep

Gurudwara Time!

Description

[Notes for grader: are in square brackets]

Waheguru Ji Ka Khalsa Waheguru Ji Ki Fateh ji
[This is a Sikh greeting. It means: *"Pure souls belong to God and Victory belongs to God"*]

This app automatically puts your phone in **silent mode** when you are at a **Gurudwara**!
[Gurudwara is Sikh place of worship and also a community center]

No one likes ringing phones when you are singing God's praises.
It is disrespect to the Guru and to the sangat.
So avoid embarrassment and make your phone sangat approved :)

Bonus: Quick navigation to nearby Gurudwaras.
Please Note: The silent mode will activate approximately 1 minute after you enter the Gurudwara.

Intended User

This is an app for people who visit Gurudwaras often
and want to automatically silent their phones.

In most places it is advisable to put the phone on silent mode, to avoid distraction.

Features

The main features of the app are:
- Automatically put phone in silent mode when at a Gurudwara
- Quick navigation option to nearby Gurudwaras using Google maps
- Change auto silent mode settings to "vibrate" or "silent"
- Exclude a Gurudwara from automatic triggering
- Add a User-defined place to Include for automatic triggering

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 - Welcome
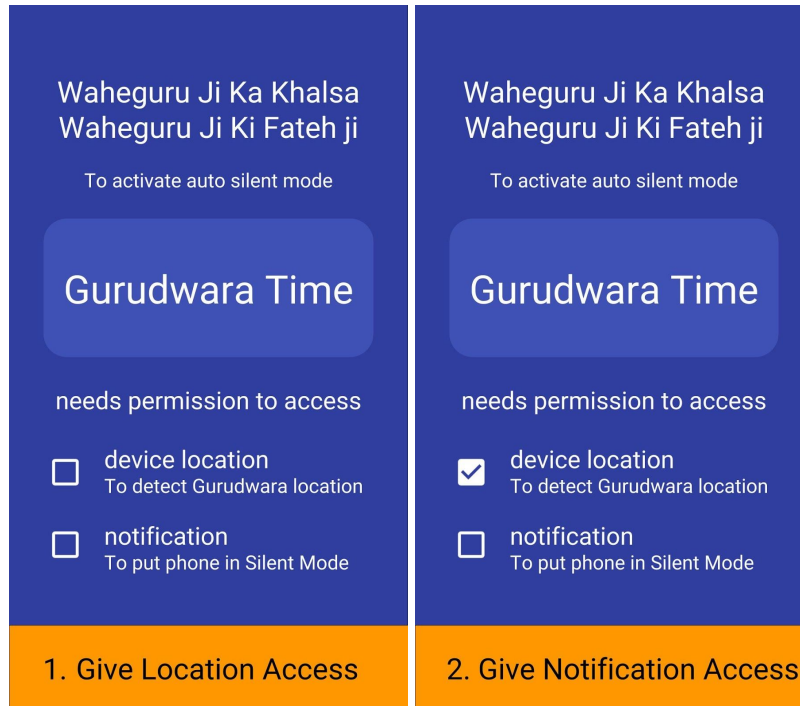
This screen is acts as Welcome screen and Permissions setup screen.
As these permissions are required for app functioning, this screen will show up initially and whenever permissions are not available from the user.
User can use checkboxes or bottom button to give permissions to app.
Screen controls help user give permission access to the app in a appropriate way i.e.

1. Permission dialog
2. Settings Screen

Note: notification permssions are needed only for Api level > 23 and will not be displayed when not needed.

Screen 2 – Auto Silent State

This screen presents user with the

1. current state of Auto Silent mode,
   (no location detected, at a detected location or off)
   (from left to right)

2. actions for detected location,
   if available (undo silent or never silent here)

3. Settings for Auto Silent

4. a Button attached to bottom which will launch a new
   activity to show nearby Gurudwaras.

5. A context menu to launch include exclude places

Screen 3 - Nearby Gurudwaras List

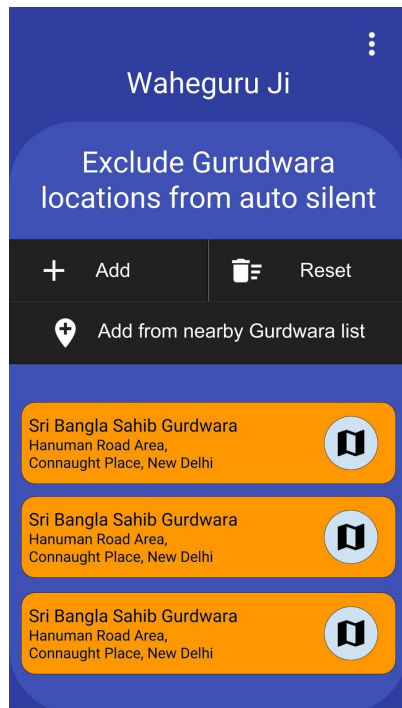| Waheguru Ji	Waheguru Ji
Nearby Gurudwara List	Nearby Gurudwara List
Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi	Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi
Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi	Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Never Auto Silent Here
Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi	Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi
Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi	Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi
Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi	Sri Bangla Sahib Gurdwara Hanuman Road Area, Connaught Place, New Delhi

This screen presents user with the
1. List of 60 nearby Gurudwaras sorted by distance from the current location (refresh on launch)
2. Swipe to refresh
3. Quick actions in list item to
    a. Open location in google maps
    b. Navigate to location in google maps
4. Long press item to open context menu
    a. to exclude Gurudwara from auto silent
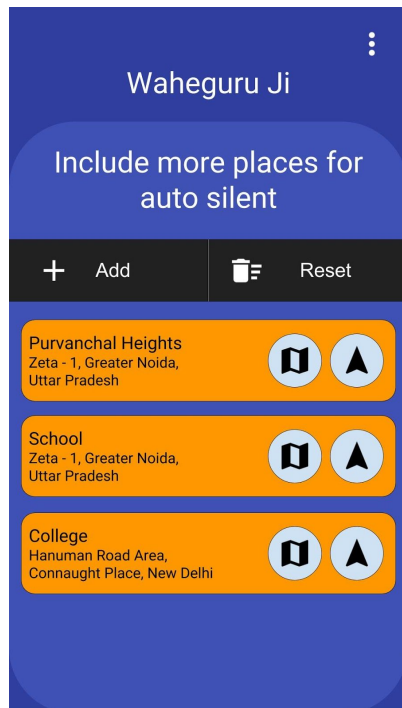
Screen 4 - Include More Places

This screen presents helps user add more places (which are not Gurudwara)
For auto silent

Waheguru Ji

Include more places for auto silent

+ Add   🗑 Reset

Purvanchal Heights
Zeta - 1, Greater Noida,
Uttar Pradesh

School
Zeta - 1, Greater Noida,
Uttar Pradesh

College
Hanuman Road Area,
Connaught Place, New Delhi

Screen 5 - Exclude List

This screen presents helps user exclude Gurudwara places from auto silent

Waheguru Ji

Exclude Gurudwara locations from auto silent

+ Add   🗑 Reset

📍 Add from nearby Gurdwara list

Sri Bangla Sahib Gurdwara
Hanuman Road Area,
Connaught Place, New Delhi

Sri Bangla Sahib Gurdwara
Hanuman Road Area,
Connaught Place, New Delhi

Sri Bangla Sahib Gurdwara
Hanuman Road Area,
Connaught Place, New Delhi

Screen 6 - App Widget

Waheguru Ji	Waheguru Ji	Waheguru Ji
You are at		
Gurudwara Bangla Sahib New Delhi	Phone will auto silent in Gurudwara	Phone will not auto silent.
☑ Auto Silent On	☑ Auto Silent On	☐ Auto Silent Off

Widget shows the current Auto silent status and setting for auto silent.

Key Considerations

How will your app handle data persistence?

App will store settings and app state in preferences,
places data in a room database.

Preference Settings:
Silent Mode setting (vibrate or silent)

App State:
Auto silent status status (to enable location tracking, and enable silent mode in geofence)
Last geofencing creation time (remember 24 hours expiry)
Last geofencing creation location (to check if need to get new geofences on location update, or if need to display nearby gurudwaras)
Current Geofence location (location data if currently inside a Gurudwara geofence)
Audio mode state before auto silent trigger (to detect manual changes by user)

Place data in Room for
Nearby: place ids, place names, place vicinity, updated_at
(cached data to quickly show, will refresh with network update when used)

Additional place ids: to monitor (non-Gurudwaras to monitor for auto silent)
Exclude place ids: to ignore monitor (Gurudwaras to exclude for auto silent)

Describe any edge or corner cases in the UX.

Whenever a User opens "Nearby List" a force refresh of nearby locations will be made according to the current location on map,
A cached copy of results from the last fetch location will be shown for faster navigation.

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

Retrofit
To execute GET request calls to Places API

Design library & cardview
To use material design components

Firebase JobDispatcher
This is used to schedule background jobs (with backward compatibility)
to fetch nearby places

Android Architecture Components (Room, Livedata and ViewModel)
For data persistence and UI update, storing place ids, names, vicinity and update time

Stetho
For debugging database and network call easily

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

play-services-location
FusedLocationApi
For location tracking every 30 minutes

Places API
Nearbysearch
To fetch a list of 60 nearby Gurudwaras
(post api call check to confirm type `place_of_worship`)

play-services-places
PlacePicker
For "include" additional places, "exclude" places.

GeoDataApi
For getPlaceById, get latest place information for saved ids.
(Note: if place type changes then adjust data)

GeofencingApi
For registering and unregistering geofences of "nearby" and "included" locations,
but not "excluded" ones

What programming language will you use?

App will be written solely in the Java Programming Language

What versions of all libraries, Gradle, and Android Studio will you use?

Dev Environment Components	Versions
Android Studio	3.2.1
Gradle	4.6

Library	Library Version
com.android.support:appcompat-v7	28.0.0
com.android.support:design	28.0.0
com.android.support:cardview-v7	28.0.0
com.android.support:preference-v7	28.0.0
com.android.support.constraint:constraint-layout	1.1.3
com.google.android.gms:play-services-location	16.0.0
com.google.android.gms:play-services-places	16.0.0
com.firebase:firebase-jobdispatcher	0.8.5
com.squareup.retrofit2:retrofit	2.5.0
com.squareup.retrofit2:converter-gson	2.5.0
android.arch.persistence.room:runtime	1.1.1
android.arch.persistence.room:compiler	1.1.1
android.arch.lifecycle:extensions	1.1.1
android.arch.lifecycle:compiler	1.1.1
com.facebook.stetho:stetho	1.5.0
com.facebook.stetho:stetho-okhttp3	1.5.0

Where will app keep strings and will app enable RTL layouts?

App keeps all strings in a `strings.xml` file.
Yes, app enables RTL layout switching on all layouts.

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

Setup new project with min sdk 14 and target sdk 18, with name "Gurudwara Time"
- Add stable gradle dependencies for play services and external libraries
- Name the first activity "WelcomeActivity"

Task 2: Implement UI for Each Activity and Fragment

- Build UI for WelcomeActivity
- Build UI for StatusActivity
- Build UI for NearbyActivity
- Build UI for StatusWidget
- Build UI for ExcludeActivity
- Build UI for IncludeMoreActivity

Task 3: Setup Welcome Screen
- Implement welcome screen with
- permissions request for location and notification access (for api >2 3)

Note: Always return to permissions screen in case permission are not available
(i.e. while accessing location or ringer mode)

Task 4: Setup Periodic Location Updates

- Implement background location updates with FusedLocationApi every 30 minutes
- Make sure location updates work even after app kill and reboot
- Turn off background location updates if auto silent setting is off.

Task 5: Setup StatusActivity

- Show status for initial load
- Show status for no location, at location, turned off
- Store current state in shared prefs
- Change state using UI contols

Task 6: Setup Room Database and repository

- Create places entity
- Implement places dao
- Implement places database
- Implement data repository to access dao

Task 7: Refresh places data in location updates

Implement a on-demand job to be called in
- location update and nearbyactivity
  - to fetch nearby places and save to db
    - Retrieve nearby places from api (check stale location, or force refresh)
    - Save data in DB
    - update geofence (just add todo comment) (done later task)

Task 8: Show places list in NearbyActivity

Setup NearbyActivity to show places list using paging library.
- Setup viewmodel and create paged list
- Implement adapter
- Implement pull to refresh places data for the current location
- Auto refresh for nearby places in onresume, even if auto silent setting is off.
  - Show loading status, use refresh-places job

Task 9: Implement Geofences

Setup Geofencing class and broadcast receiver
- Implement Geofencing class like Shushme project
- Setup updateGeofences method for places when
  - auto silent setting is On and places data is updated
- Setup job to updateGeofences every 23 hours (24 hours is geofence expiry)
  - Start this job in Main Application (implement Application class)

Task 10: Implement Silent mode on Geofence Transitions
- Setup Silent mode in geofence broadcast receiver
  - Use DWELL and EXIT transitions
  - Check auto silent setting
  - Send a notification to identify current state
  - Save ringer mode before changing to restore later

Task 11: Implement App widget
- Setup code for app widget to show current app status
- Update app widget whenever app status changes i.e.
  in initial load, geofence transitions, silent mode switch on/off.

Task 12: Implement Place exclude
- Implement ExcludeActivity
- Add place picker
  - Make sure only Gurudwaras are added to exclude list with place picker
- Save excluded places to db (update flag if exists)
- updateGeofences
- Setup context menu for long press item in nearbyactivity to exclude place

Task 13: Implement Place include
- Implement IncludeMoreActivity
- Add place picker
  - Make sure Gurudwaras are not added to include list with place picker
- Save included places to db (update if exists)
- updateGeofences

Bonus Task: Implement about page
- Tell about this project and nanodegree

Add as many tasks as you need to complete your app.

Submission Instructions
- After you've completed all the sections, download this document as a PDF [File →
  Download as PDF]
  - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal
If using GitHub:
- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"