How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: "**Capstone_Stage1**"
3. Replace the text in green

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**"

GitHub Username: MangeshKadam

Devotional Aartis & Kathas

Description

This app displays the images of different gods. In detail view you can see the list of different prayers(aartis) specific to that god. When you click on any aarti name, details aarti will be displayed.

Intended User

This app is useful for all tthe people who use paper for reading prayers and kathas.

Features

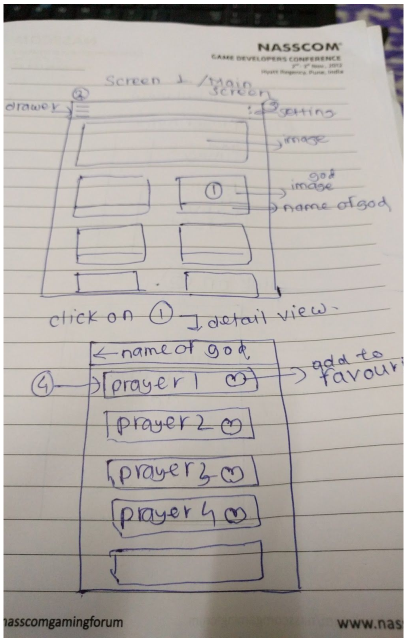
List the main features of your app. For example:

- You can mark 3 favourite aartis.
- Collection of more than 100 aaris
- Aartis are displayed in native language font. I.e marathi or hindi.
- Works in complete offline mode.

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 Photoshop or Balsamiq.

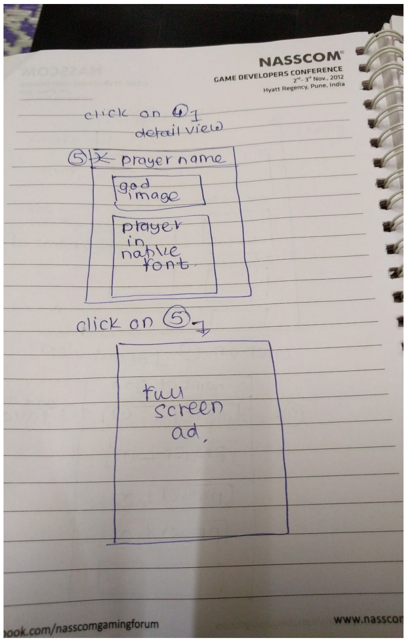
Screen 1



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]

Main Screen and screen 2- having setting drawer and list of gods with their names. When clicked on 1 in main screen, screen 2 will display list of Aartis.

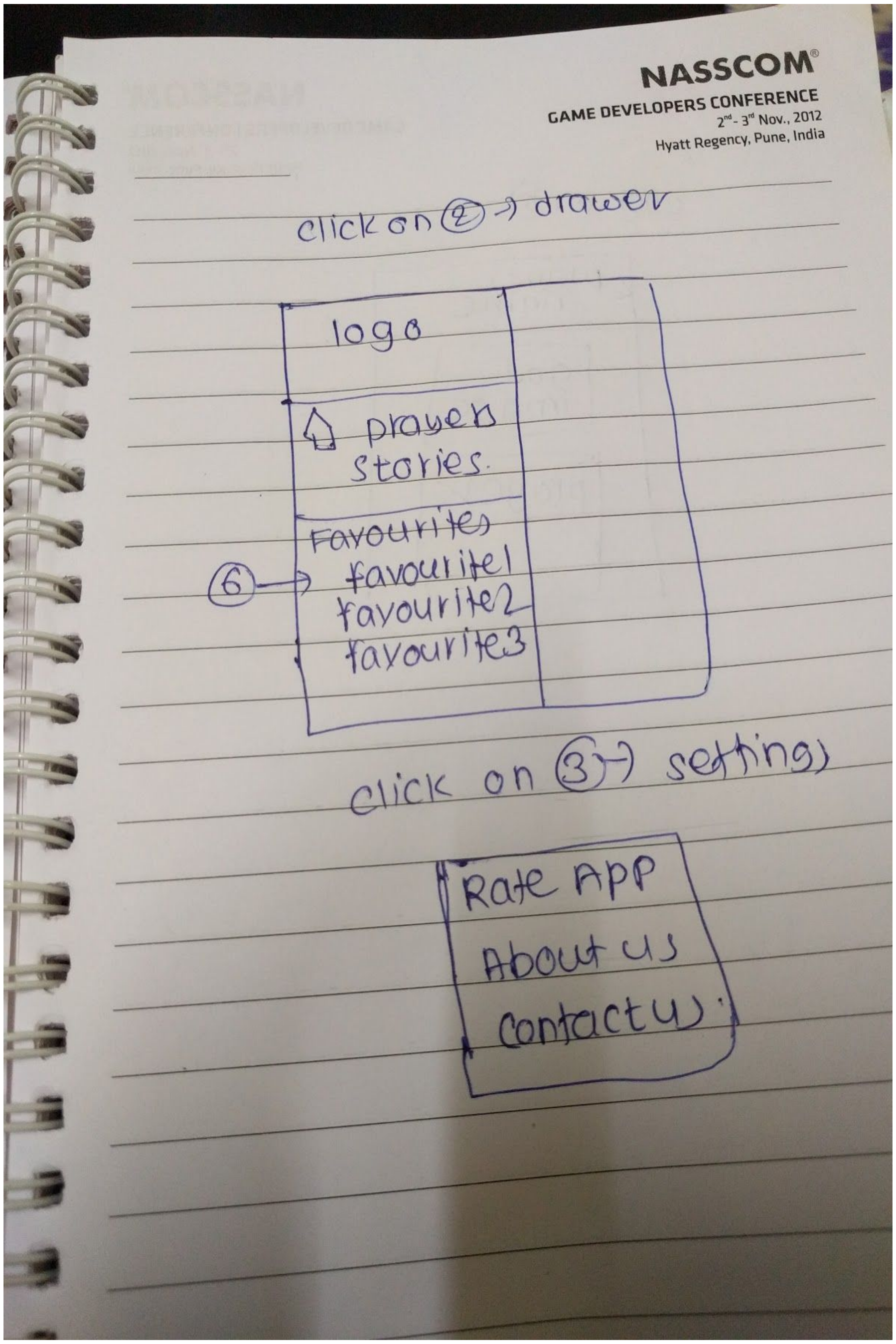
Screen 2



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]

When clicked on 4 in screen 2, detailed aarti or katha will be displayed. When user clicks on 5 in screen 3 - full screen ad will be displayed.

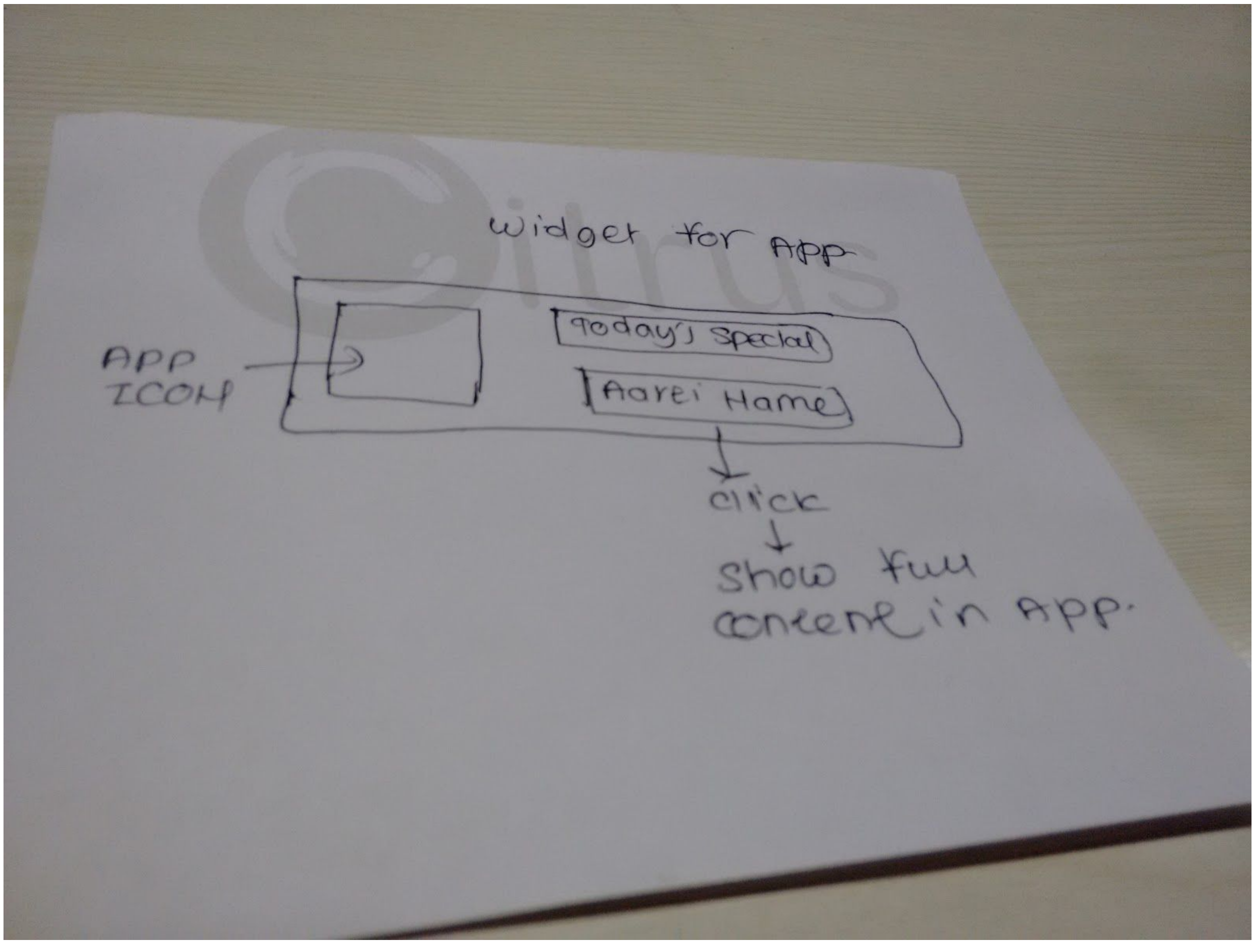
Screen 3



When user clicks on 2 in screen 1, drawer will be displayed. Drawer will have sub menus - aartis and kathas. Favourite aartis will be displayed here.

When user clicks on 3 in setting, Rate App, About us and Contact us will be displayed.

Screen 4



App Widget - Random Aarti name will be displayed every day. When user clicks on the widget detail aarti contents will be displayed.

Key Considerations

How will your app handle data persistence?

Shared preference will be used to save favourite item. Aartis and kathas will be in .txt file inside assets folder. Content-Provider will be used to read those txt file.

Describe any corner cases in the UX.

If user click on favourite item , aarti will be displayed. When user click on back button he will returned to main - menu instead of aarti's list.

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

Butterknife - to bind the views
Card view - to show items in card.
Firebase messaging - to send the notifications
Firebase database - to read JSON file to check app update.
Play-services ad - to display the full screen ad

Describe how you will implement Google Play Services.

Play Services Ad - to display the full screen ad.
Google Play Services analytics - to measure user activity and mostly visited aarti/katha.

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

Write out the steps you will take to setup and/or configure this project. See previous implementation guides for an example.

Configure 3rd party libraries
Configure firebase for App

Task 2: Implement UI for Each Activity and Fragment

1. Build UI for drawer
2. Build UI for setting

Task 3: Your Next Task

1. Build UI for Main Activity.
2. Build UI for for fragment with recycler view in main activity.
3. Build UI for Aarti list and detail activity for selected aarti.
4. Collect all the aartis and kathas , images of god.

Task 4: Your Next Task

1 .Build App rater dialog.

2. Build contact-us and about us activity.

Task 5: Your Next Task

1. Add firebase notification
2. Add admob code to show full screen ad.
3. Add google analytics for selcted aarti
4. Add firebase-storage code for reading config json to check if there is any update available or not.
5. Add fonts and use custom fonts to display aarti and kathas in native font.
6. Build add to favourite functionality.
7. Build widget and its functionality.
8. App usage AsyncTask to read the content of txt file.

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

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**”