

CAPSTONE PROJECT

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[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.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

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

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: abilaashsai@gmail.com

Sai Kailash

Description

Sai Kailash is the official application of Sri Sathya Sai Organization Ooty(Non-profit organization). I am also on the track of developing a website and it is half way through, the link for the organization is [here](#). Currently there is neither website nor mobile app for the organization for easy communication and subscription to events.

Sai Kailash mobile app will help to engage more people willing to serve the society. The advanced features that enable user engagement and easy communication are: to notify the events and announcements instantly to the users and permanently make available on the mobile app. All the information is digitized, which helps the users to stay updated with latest happenings in the organization.

Intended User

Irrespective of religion, age, gender, national, it is intended to people who are willing to serve the society with love and passion.

Features

- Events updates- Future and past
- Blogs and Articles
- Video of events
- Location of organization
- Notification of latest happenings
- Photos of event

User Interface Mocks

1. Home Screen

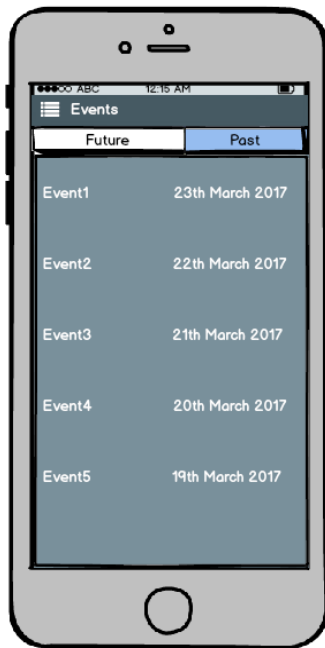
2. Hamburger menu

3. Events Screen(Future)



All the information in **Home Screen** and **Events Screen** are retrieved from Firebase.

4. Home Screen(Past)



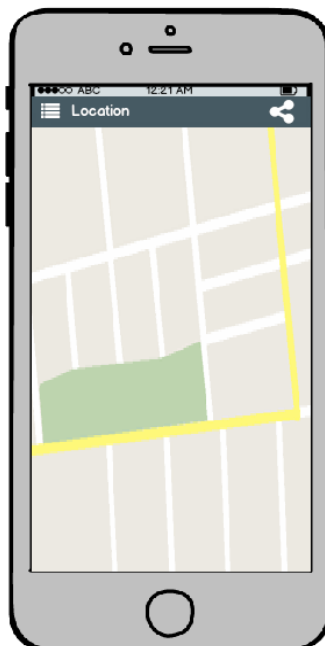
5. Article Screen(List View)



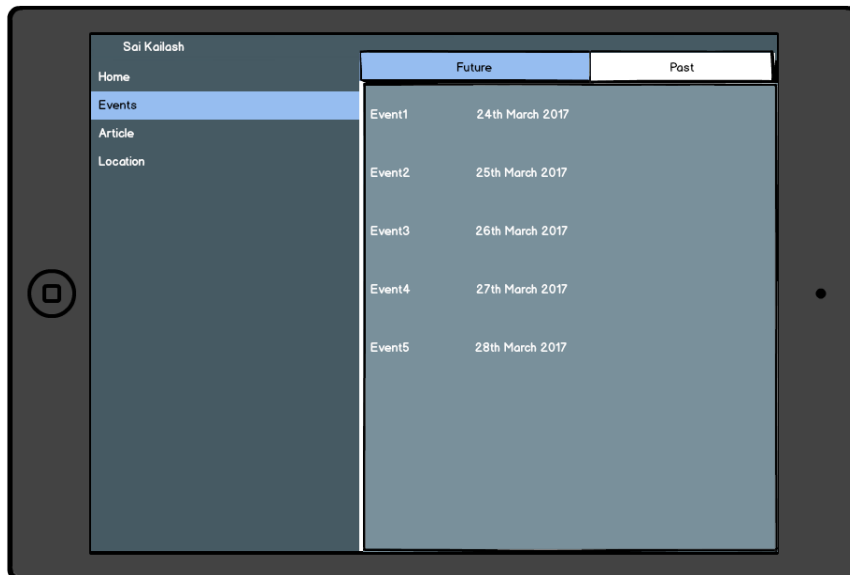
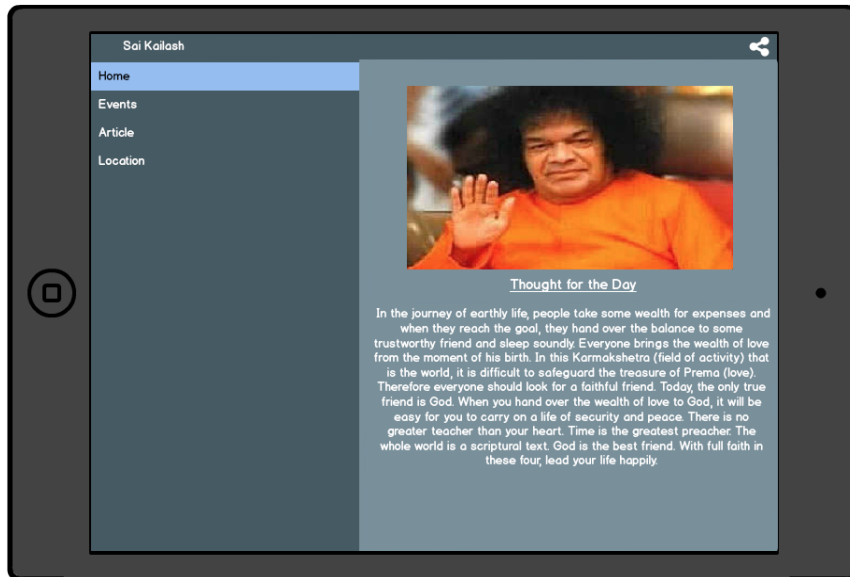
6. Article Screen(Detail View)

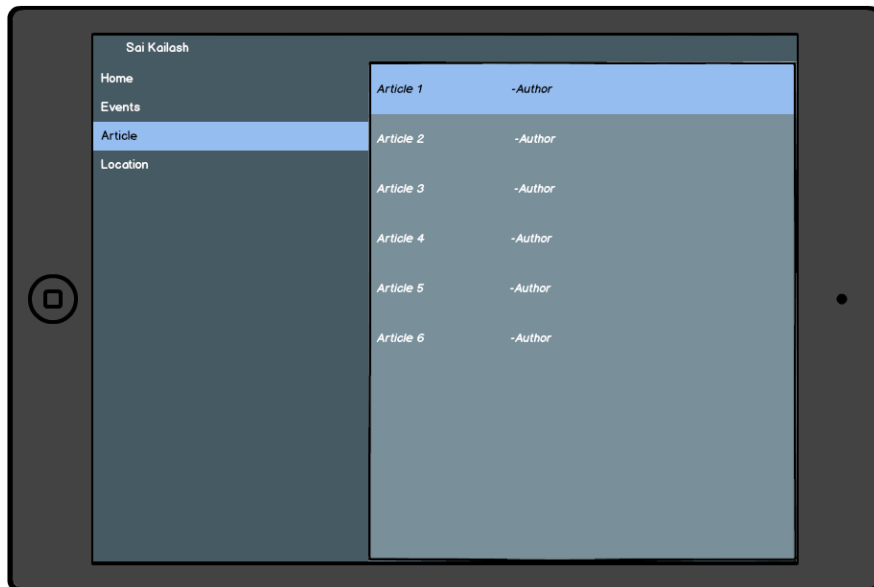


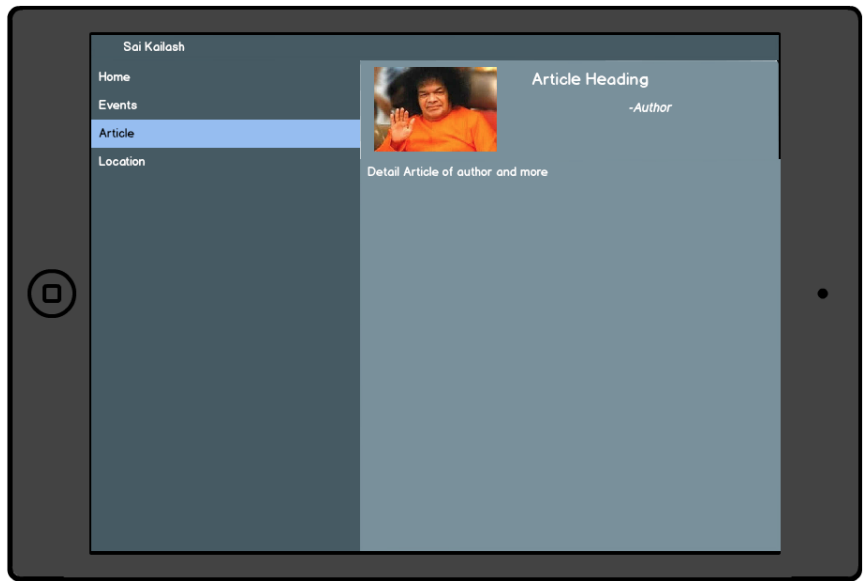
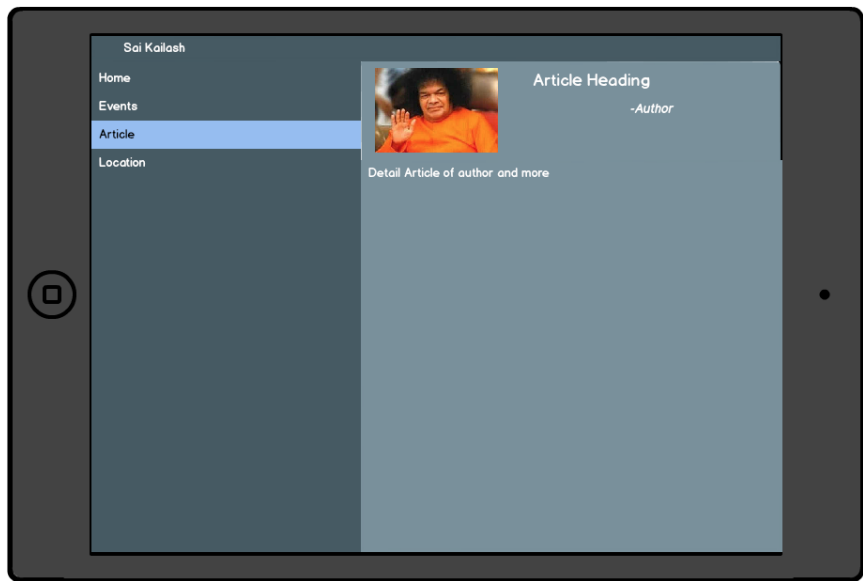
7. Location Screen



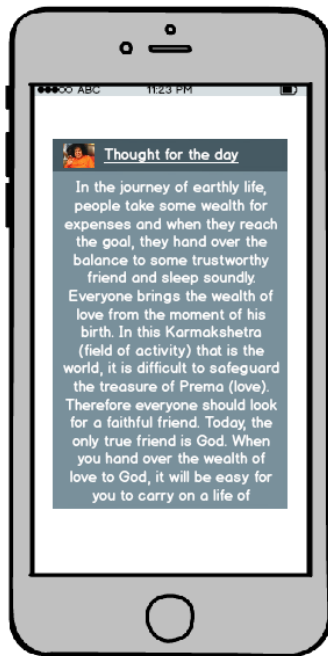
Tablet View







Widget



Key Considerations

How will your app handle data persistence?

It uses Firebase Database to fetch data. When in offline it makes use of local storage(SQLite) using Content Providers and Loaders to load data from SQLite.

Describe any corner cases in the UX.

Hamburger menu is used to navigate to any part of the application, in addition back button is used to come out of the flow.

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

- Picasso for image loading
- Butterknife to inflate the views
- Junit for testing

Describe how you will implement Google Play Services.

- Firebase Real-Time Database for syncing data
- Firebase Notification to receive notification on events
- Firebase Storage for receiving images that are stored

Next Steps: Required Tasks

Task 1: Hamburger Menu

Develop main base framework for building the features and functionalities on it

- Build UI on fragment for MainActivity
- Creating Layouts for Fragments
- Setup hamburger menu with options

Task 2: Implement Home Screen by using Firebase

- Design Main Home screen
- Fetch data from Firebase Realtime Database
- Display data from Firebase to home screen
- Using content providers to store data and Loaders to load from database
- SyncAdapter to update the data when changed

Task 3: Design Events List View

- Creating Feature Session and Upcoming Session Dividers
- Listview to display future events and past events
- Using content providers to store data and Loaders to load from database
- SyncAdapter to update the data when updated

Task 4: Design Article Section and Share option

- Listview to display Article Title
- Clicking on it should go to Detail View
- Detail View should contain whole article
- Share button in Detail View to share articles

Task 5: Google Maps for Direction

- Import and add Google map services permission
- Locate the organization
- Sharing option to share the location

Task 6: Building Notification Functionality

- Import Firebase notification library
- Enable Firebase Notification feature