TECHNICAL DOCUMENTATION FOR

DOORTRACK

Mobile App

Step 1: Install android studio if its not available in your Computer

Download the android studio https://developer.android.com/studio from this official website of Google and install it in your pc

Step 2: Install Flutter in your Computer

1. Download or clone the following installation bundle to get the latest stable release of the Flutter SDK.

Download link : git clone -b master https://github.com/flutter/flutter.git

2.Create a folder where you can install Flutter.

3.Update your Windows PATH variable

To run Flutter commands in PowerShell, add Flutter to the PATH environment variable.

1. Press Windows + S.
2. Type environment.
3. When Edit the system environment variables displays as the Best match, click Open under Edit the system environment variables.
4. Click About.
5. Click Advanced System Settings.
6. Click Environment Variables…
   The Environment Variables dialog displays.
7. Under User variables for <user> check for the Path entry.
   1. If the entry exists, click Edit….
   2. If the entry doesn't exist, click New….
   3. Click New.
   4. Type <install-directory>\flutter\bin.
   5. Click the <install-directory>\flutter\bin entry.
   6. Click Move Up until the Flutter entry sits at the top of the list.
   7. Click OK.
8. To enable these changes, close and reopen any existing command prompts and PowerShell instances.

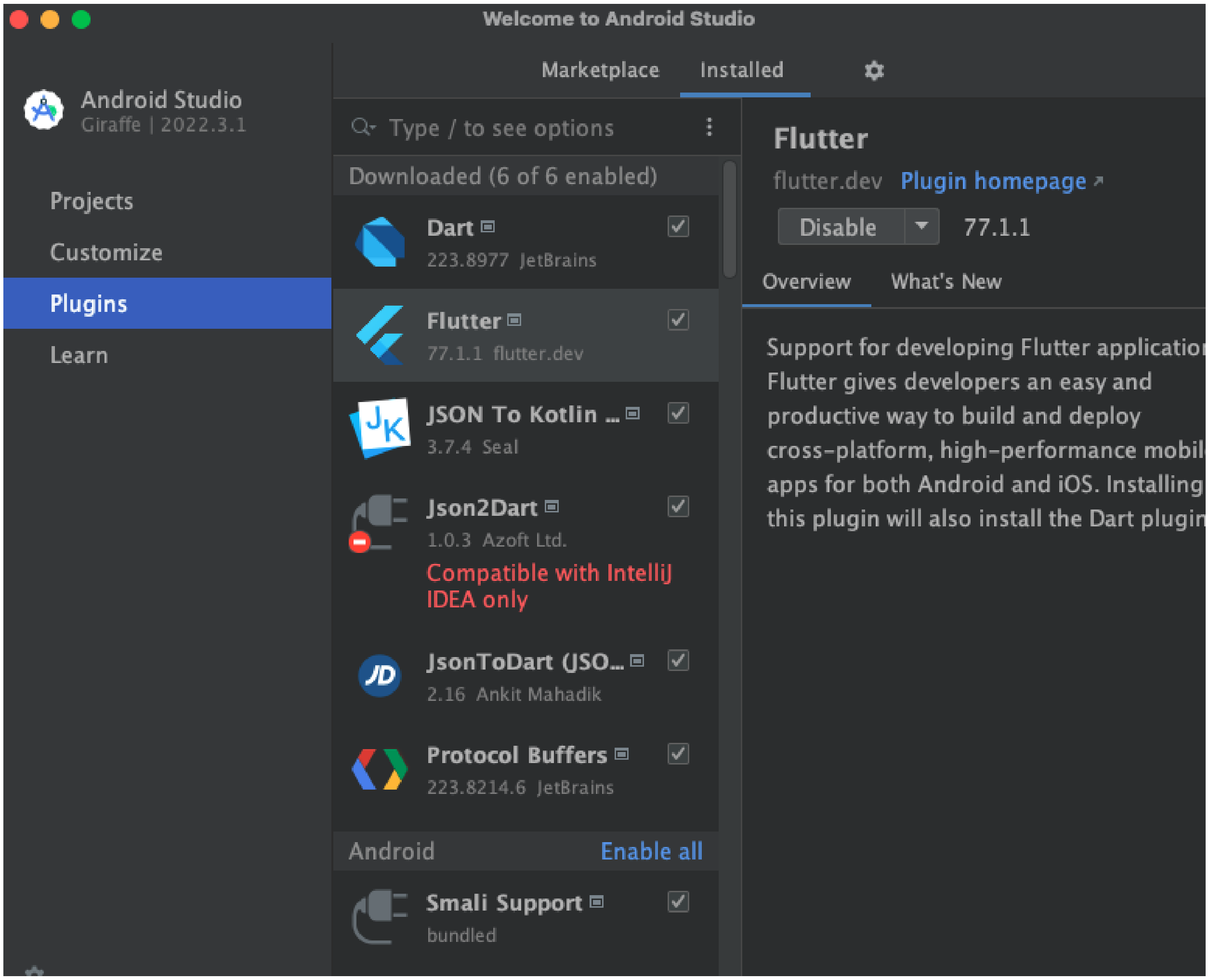
If you have installed all prerequisites and the Flutter SDK, you should be able to start developing Flutter on Windows for mobile.

4. Configure the Android toolchain in Android Studio

Start Android Studio.Follow the Android Studio Setup Wizard.

1. Install the following components:
   - ○ Android SDK Platform, API 33.0.0
   - ○ Android SDK Command-line Tools
   - ○ Android SDK Build-Tools
   - ○ Android SDK Platform-Tools
   - ○ Android Emulator

5. Install Flutter and Dart plugin in Android studio



Downloaded (6 of 6 enabled)

Dart ☑
223.8977 JetBrains

Flutter ☑
77.1.1 flutter.dev

JSON To Kotlin ... ☑
3.7.4 Seal

Json2Dart ☑
1.0.3 Azoft Ltd.
Compatible with IntelliJ IDEA only

JsonToDart (JSO... ☑
2.16 Ankit Mahadik

Protocol Buffers ☑
223.8214.6 JetBrains

Android

Smali Support ☑
bundled

Flutter
flutter.dev
77.1.1

Support for developing Flutter applicatio
Flutter gives developers an easy and
productive way to build and deploy
cross-platform, high-performance mobil
apps for both Android and iOS. Installing
this plugin will also install the Dart plugin

6. Before you can use Flutter and after you install all prerequisites, agree to the licenses of the Android SDK platform.

1. Open an elevated console window.
2. Run the following command to enable signing licenses.

   ```
   flutter doctor --android-licenses
   ```

3. You can skip the next step.
4. Before agreeing to the terms of each license, read each with care.
5. Check flutter installation is running perfectly

   Open command line interface and run the command : `flutter doctor -v`

Step 3: Open flutter Project in Android studio

Step 4: Connect your real device via usb to your computer or create an emulator in Android studio .

STep 5: Build the project in an Emulator or real device from android studio and it's a very easy step .

First run the project and follow the source code functional documentation below.I declared here every fun working process and what is responsible for.

OVERVIEW:

In this app we have used some packages. Here is the list of the packages and usages of the packages.

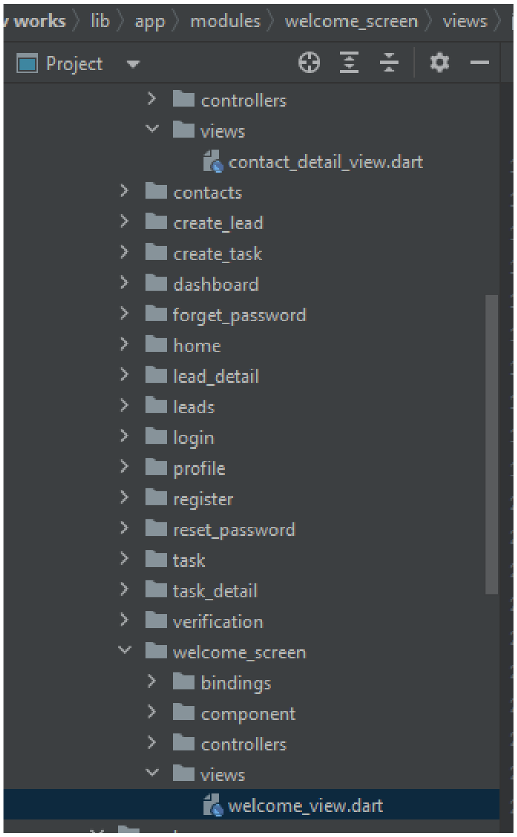
Package Name	Usages
get	State management
shared_preferance	Local storage
http	Network calls
google_fonts	Text styling
carousel_slider	sliding
dots_indicator	Showing the dots
flutter_screenutil	Responsive design
flutter_easyloading	Loading indicator
flutter_native_splash	Showing splash native
flutter_launcher_icon	Updating Flutter app's launcher icon
Cached_network_image	Network image handling
Flutter_rounded_date_picker	For customised date picker
another_flushbar	Showing msg in the app
fluttertoast	Showing msg in the app
Permission_handler	Handling permission request
Flutter_spinkit	Loading indicator
Flutter_svg	Display svg images
Infinite_scroll_pagination	Pagination
Awesome_dialogue	Showing dialogue
Awesome_snackbar_content	Showing snack bar msg
gap	Spacing dynamically
Flutter_country_code_picker	Picking country code
Flutter_calendar_week	Showing calendar week

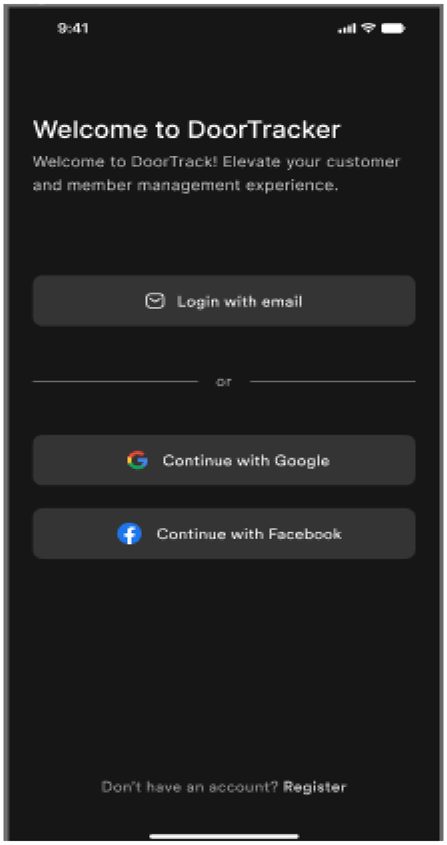
Note: Controller class is responsible for api implementation and functionality and that documentation added after api implementation

After Successful build it will open first and its welcome screen and every screen is described below .

Here are the list and detail all screen

1. Welcome Screen





Welcome to DoorTracker

Welcome to DoorTrack! Elevate your customer and member management experience.

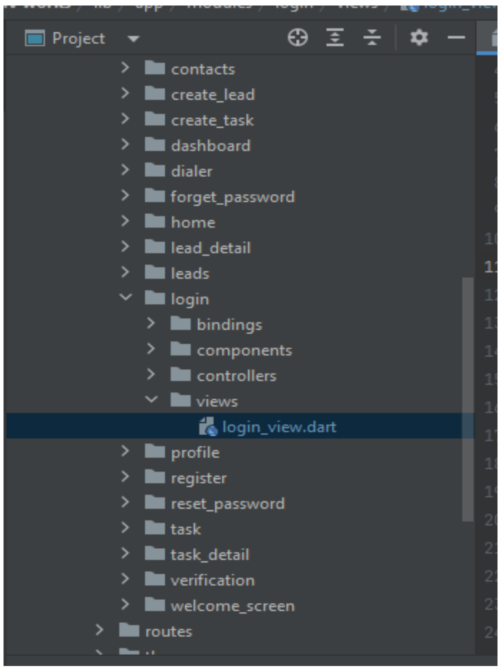
Class name: WelcomeView that extends GetView<WelcomeController> class.

Here are the details about the WelcomeView class. This screen design is divided into three parts. The three methods are stored in **welcome_screen/component**

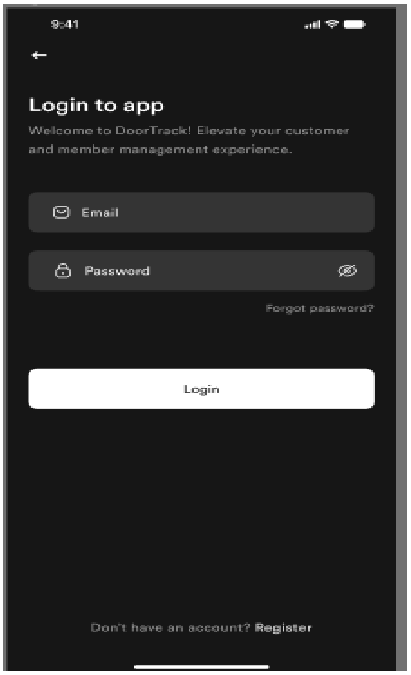
Here is the detail and usages of these methods.

Methods	Description	Api
header()	Welcome text	
loginDescripsionButton()	The buttons are email, google, facebook.	
footerRegistration()	Text button “register”	

2. Login Screen



```
> contacts
> create_lead
> create_task
> dashboard
> dialer
> forget_password
> home
> lead_detail
> leads
v login
    > bindings
    > components
    > controllers
    v views
        login_view.dart
> profile
> register
> reset_password
> task
> task_detail
> verification
> welcome_screen
> routes
```



Login to app

Welcome to DoorTrack! Elevate your customer and member management experience.

Don't have an account? **Register**

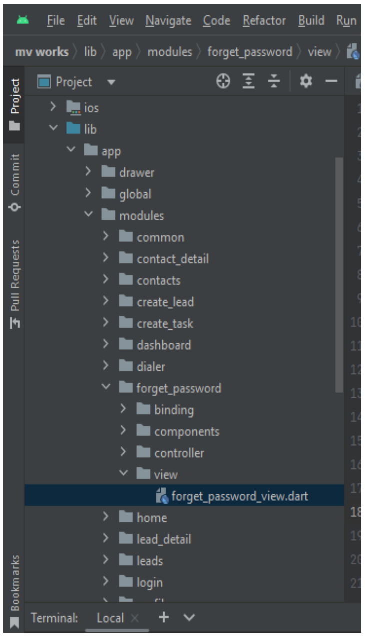
Class name: LoginView that extends GetView<LoginController> class.

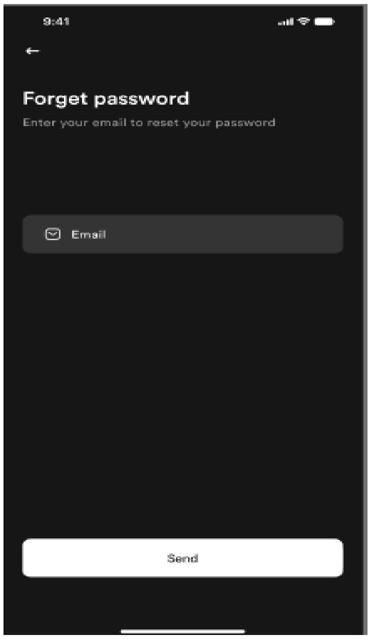
Here is the detail about LoginView class. This screen design is divided into three part. The three methods are stored in **Login_screen/component/**

Here is the detail and usages of these methods.

Methods	Description	Api
header()	Welcome text	
loginFields()	The textfields are email, password. Buttons are forget password, login.	
footerRegistration()	Text button "register"	

3. Forget Password





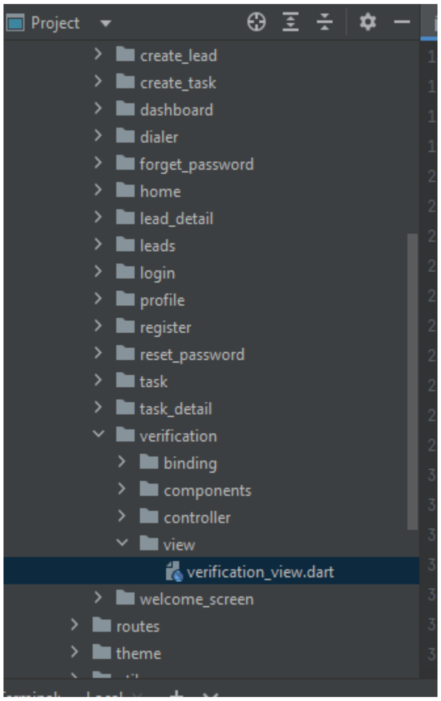
Class name: ForgetPasswordView that extends GetView<ForgetPasswordController> class.

Here are the details about the ForgetPasswordView class. This screen design is divided into three parts. The three methods are stored in **Login_screen/component/**

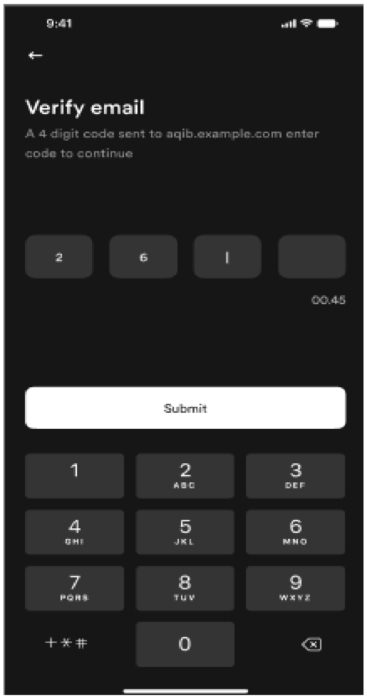
Here is the detail and usages of these methods.

Methods	Description	Api
header()	Forget Password text	
emailField()	The textfields are email.	
Floating action button	Send button. Use to send otp	

4. Verification View



```
Project
  > create_lead
  > create_task
  > dashboard
  > dialer
  > forget_password
  > home
  > lead_detail
  > leads
  > login
  > profile
  > register
  > reset_password
  > task
  > task_detail
  v verification
    > binding
    > components
    > controller
    v view
        verification_view.dart
  > welcome_screen
> routes
> theme
```



Verify email

A 4 digit code sent to aqib.example.com enter code to continue

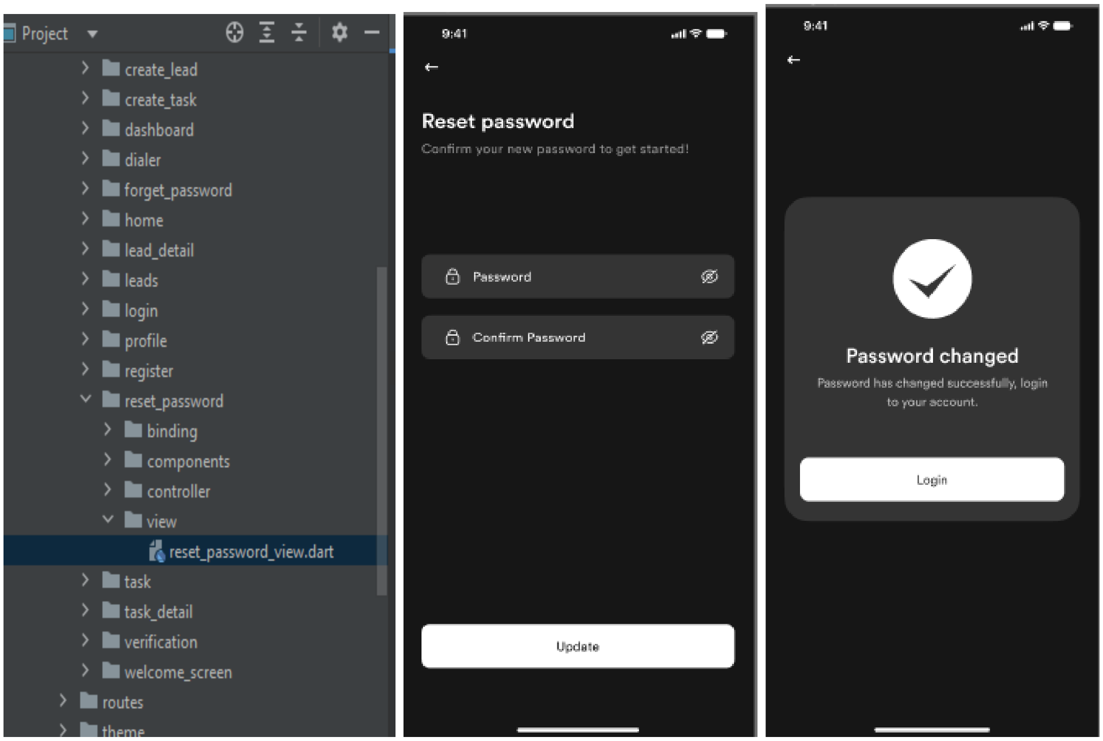
Class name: VerificationView that extends GetView<VerificationController> class.

Here is the detail about the VerificationView class. This screen design is divided into three parts. The two methods are stored in **Verification_screen/component/**

Here is the detail and usages of these methods.

Methods	Description	Api
header()	Verify email text	
otpTextFields()	The text fields are otp text fields.	
Floating action button	Submit button. Verify otp	

5. Reset Password



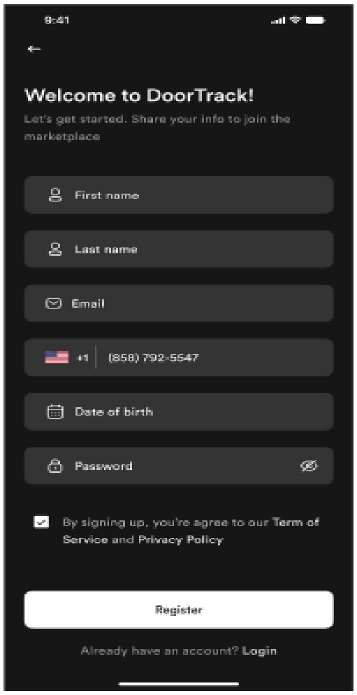
Class name: ResetPasswordView that extends GetView<ResetPasswordController> class.

Here is the detail about the ResetPasswordView class. This screen design is divided into three parts. The two methods are stored in **reset_password/component/**

Here is the detail and usages of these methods.

Methods	Description	Api
header()	Reset passwod text	
reseatPasswordFields()	The text fields are confirm password, confirm again password.	
Floating action button	Update button and success dialogue.	

6. **Registration Screen**

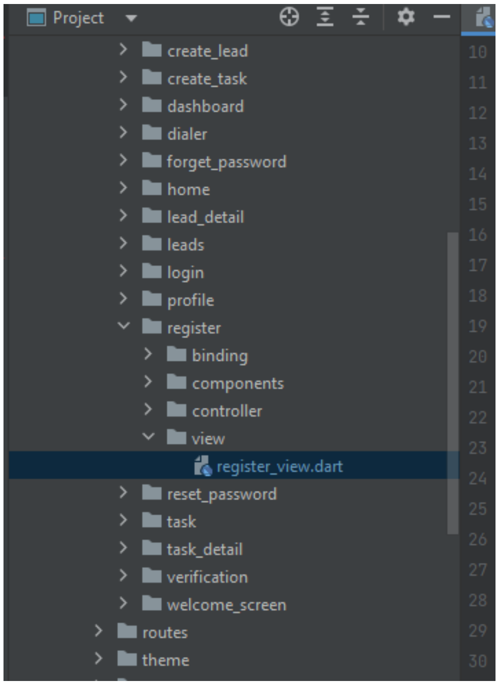


Welcome to DoorTrack!

Let's get started. Share your info to join the marketplace

☑ By signing up, you're agree to our **Term of Service** and **Privacy Policy**

Already have an account? **Login**



```
    > create_lead
    > create_task
    > dashboard
    > dialer
    > forget_password
    > home
    > lead_detail
    > leads
    > login
    > profile
    v register
        > binding
        > components
        > controller
        v view
            register_view.dart
    > reset_password
    > task
    > task_detail
    > verification
    > welcome_screen
> routes
> theme
```

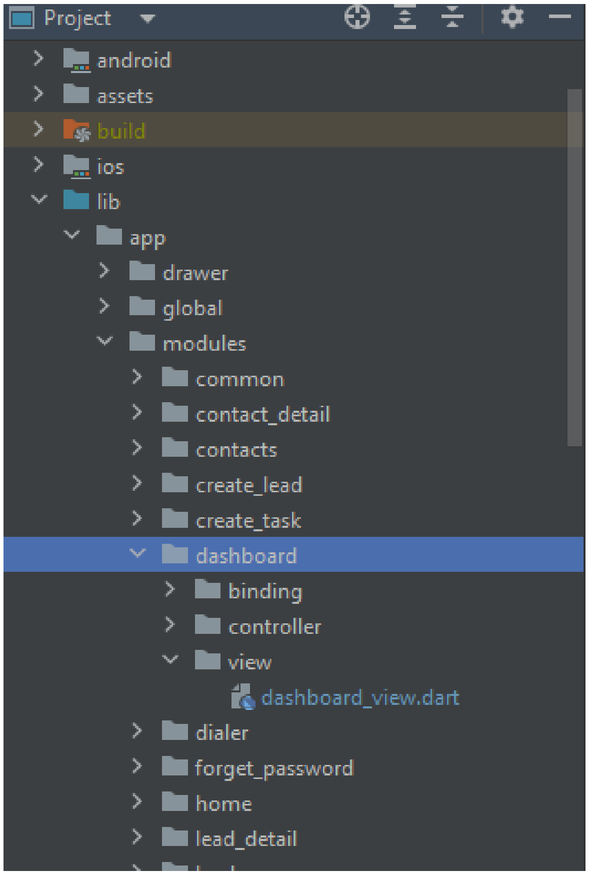
Class name: RegistrationView that extends GetView<RegistrationController> class.

Here are the details about the RegistrationView class. This screen design is divided into three parts. The three methods are stored in **registration/components/**

Here is the detail and usages of these methods.

Methods	**Description**	**Api**
header()	Welcome text	
registrationFields()	The text fields are first name, last name, email, password, date of birth and a register button.	
footerLogin()	"Login" button	

7. Dashboard

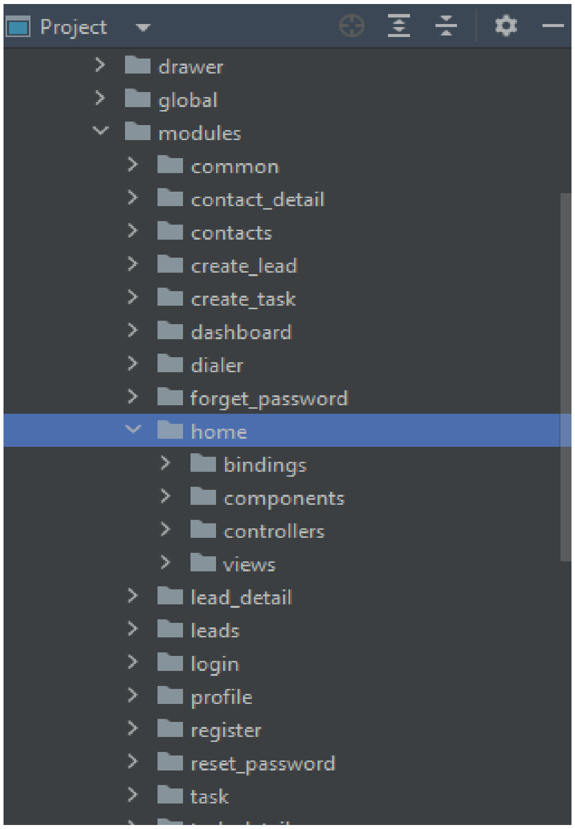


```
Project
> android
> assets
> build
> ios
v lib
  v app
    > drawer
    > global
    v modules
      > common
      > contact_detail
      > contacts
      > create_lead
      > create_task
      v dashboard
        > binding
        > controller
        v view
            dashboard_view.dart
      > dialer
      > forget_password
      > home
      > lead_detail
```

Class name: DashboardView that extends GetView< DashboardController> class.

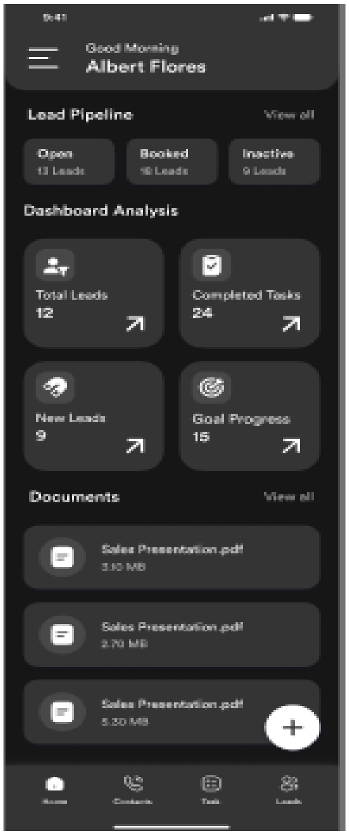
Here are the details about the DashboardView class. In the body we use DashboardController class where we declare all the pages named "bottomPageList" and in bottom navigation all the bottom navigation buttons are declared. Inside BottomNavigationBar class onTap parameter we use DashboardController where we define the "onBottomTap" method it uses to set the tap index of the bottom button.

8. Home



Project

- drawer
- global
- modules
  - common
  - contact_detail
  - contacts
  - create_lead
  - create_task
  - dashboard
  - dialer
  - forget_password
  - home
    - bindings
    - components
    - controllers
    - views
  - lead_detail
  - leads
  - login
  - profile
  - register
  - reset_password
  - task



9:41

Good Morning
Albert Flores

Lead Pipeline View all

Open
13 Leads

Booked
18 Leads

Inactive
9 Leads

Dashboard Analysis

Total Leads
12

Completed Tasks
24

New Leads
9

Goal Progress
15

Documents View all

Sales Presentation.pdf
3.10 MB

Sales Presentation.pdf
2.70 MB

Sales Presentation.pdf
5.30 MB

Class Name:

HomeView that extends GetView<HomeController> class.

Details:

We use a common method “mainAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

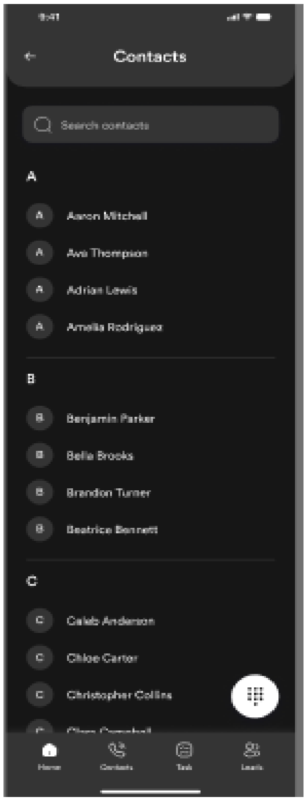
In the drawer method we use MainDrawer class that is located in lib/app/drawer/drawer.dart

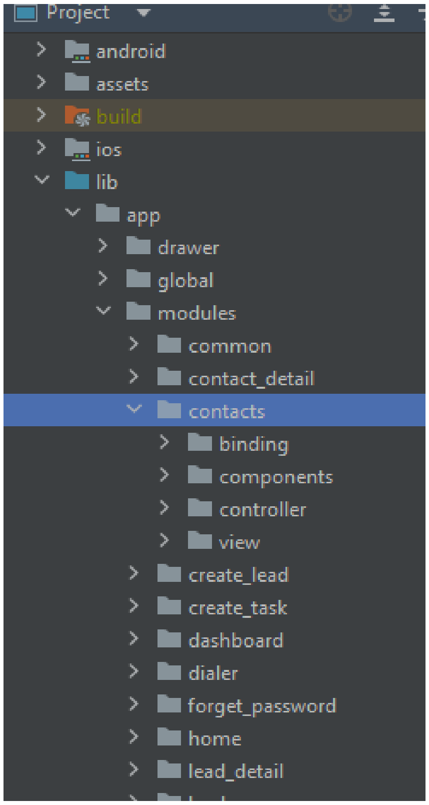
In the floating action button we use a method “getPickedFileDialogue()” which shows the picked file options.

Location of the method is lib/app/utils/success_pop_up.dart

Methods	Descriptions	Api
mainAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	
getPickedFileDialogue()	which shows the pop up dialogue picked file options. lib/app/utils/success_pop_up.dart	
leadPipeline()	It is used to design the lead pipeline items. lib/app/modules/home/components/lead_pipeline.dart	
dashboardAnalysis()	It is used to design the dashboard analysis items. lib/app/modules/home/components/dashboard_analysis.dart	
documents()	It is used to design the document items. lib/app/modules/home/components/documents.dart	

9. Contacts





```
Project
> android
> assets
> build
> ios
v lib
  v app
    > drawer
    > global
    v modules
      > common
      > contact_detail
      v contacts
        > binding
        > components
        > controller
        > view
      > create_lead
      > create_task
      > dashboard
      > dialer
      > forget_password
      > home
      > lead_detail
```

Class Name:

ContactsView that extends GetView<ContactsController> class.

Details:

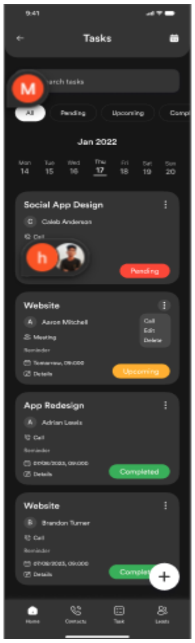
We use a common method “mainAppbar()” in the appbar.

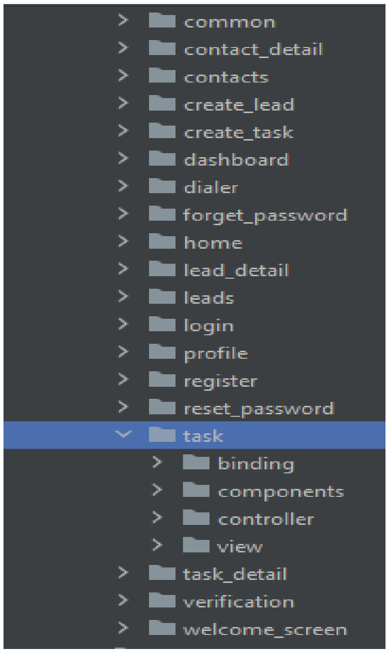
Location of the specified method is lib/app/theme/custom_appbar.dart

In the floating action button we used an svg image of the dialer.

Methods	Descriptions	Api
mainAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	
searchTextField()	which shows the search bar. lib/app/modules/contacts/components/search_text_fields.dart	
leadPipeline()	It is used to design the contacts body items. lib/app/modules/contacts/components/contacts_body.dart	

10. Task





```
> common
> contact_detail
> contacts
> create_lead
> create_task
> dashboard
> dialer
> forget_password
> home
> lead_detail
> leads
> login
> profile
> register
> reset_password
v task
    > binding
    > components
    > controller
    > view
> task_detail
> verification
> welcome_screen
```

Class Name:

TaskView that extends GetView<TaskController> class.

Details:

We use a common method “mainAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

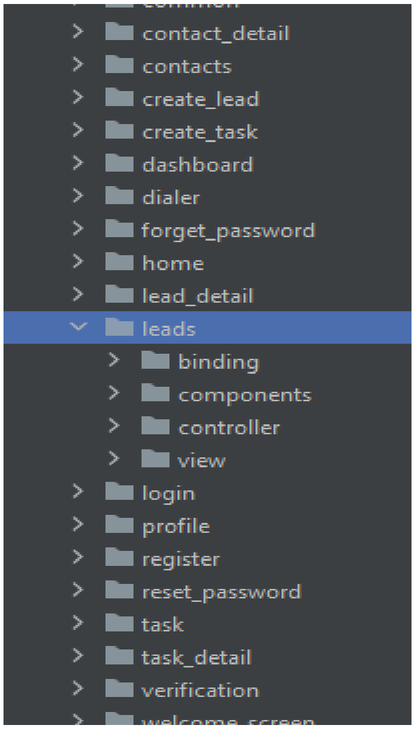
In the floating action button we used an svg image of add.

Methods	Descriptions	Api
mainAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	
searchTextField()	which shows the search bar.	

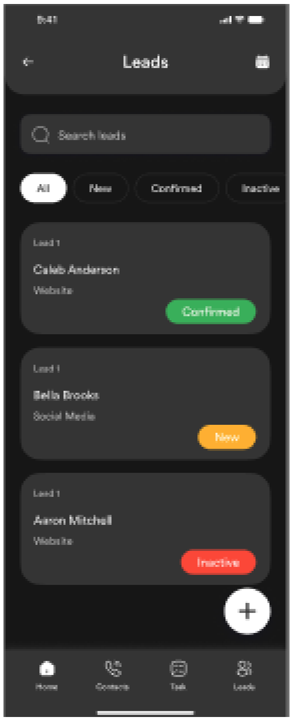
	lib/app/modules/task/components/search_text_fields.dart	
horizontalListButtons()	It is used to design the list of buttons. lib/app/modules/task/components/horizontal_list_button.dart	

weekDay()	It shows the week day calendar view. lib/app/modules/task/components/weekday.dart	
taskItemList()	It shows the task items. lib/app/modules/task/components/task_item_list.dart	

11. Leads



```
  > common
  > contact_detail
  > contacts
  > create_lead
  > create_task
  > dashboard
  > dialer
  > forget_password
  > home
  > lead_detail
  v leads
      > binding
      > components
      > controller
      > view
  > login
  > profile
  > register
  > reset_password
  > task
  > task_detail
  > verification
  > welcome_screen
```



Lead 1
Caleb Anderson
Website
Confirmed

Lead 1
Bella Brooks
Social Media
New

Lead 1
Aaron Mitchell
Website
Inactive

Class Name:

TaskView that extends GetView<TaskController> class.

Details:

We use a common method “mainAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

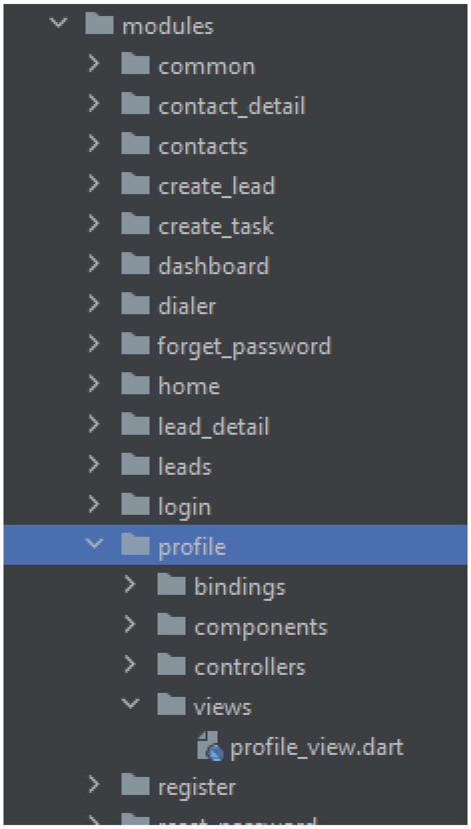
In the floating action button we used an svg image of add.

Methods	Descriptions	Api
mainAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	

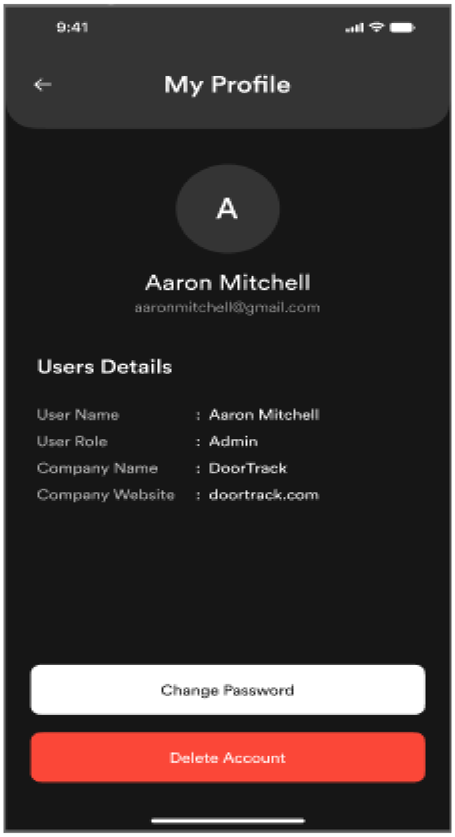
searchTextField()	which shows the search bar. lib/app/modules/lead/components/search_text_fields.dart	
horizontalListButtons()	It is used to design the list of buttons. lib/app/modules/lead/components/horizontal_list_button.dart	

weekDay()	It shows the week day calendar view. lib/app/modules/lead/components/weekday.dart	
leadItemList()	It shows the lead items. lib/app/modules/lead/components/lead_item_list.dart	

12. Profile



```
modules
  common
  contact_detail
  contacts
  create_lead
  create_task
  dashboard
  dialer
  forget_password
  home
  lead_detail
  leads
  login
  profile
    bindings
    components
    controllers
    views
      profile_view.dart
  register
  reset_password
```



Aaron Mitchell
aaronmitchell@gmail.com

Users Details

User Name	:	Aaron Mitchell
User Role	:	Admin
Company Name	:	DoorTrack
Company Website	:	doortrack.com

Class Name:

ProfileView that extends GetView<ProfileController> class.

Details:

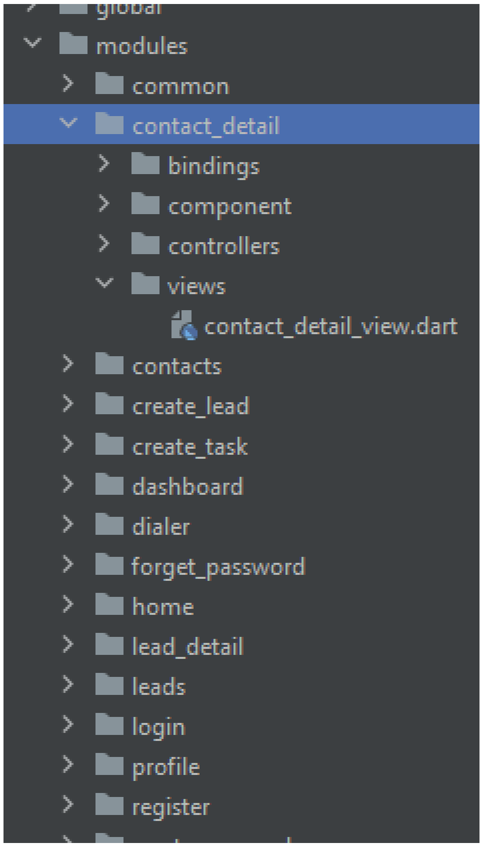
We use a common method "mainAppbar()" in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

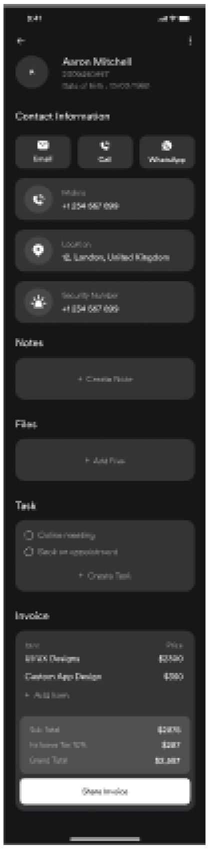
Methods	Descriptions	Api
mainAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	

profileHeader()	Used to show user name, email. lib/app/modules/profile/component/profile_header.dart	
profileDetail()	Used to show user details such as name, role etc. lib/app/modules/profile/component/profile_detail.dart	
changePassDeleteAccount()	Used to show change pass and delete account buttons. lib/app/modules/profile/component/change_pass_delete_account.dart	

13. Contact Detail



```
global
modules
  common
  contact_detail
    bindings
    component
    controllers
    views
      contact_detail_view.dart
  contacts
  create_lead
  create_task
  dashboard
  dialer
  forget_password
  home
  lead_detail
  leads
  login
  profile
  register
```



Class Name:

ContactDetailView that extends GetView<ContactDetailController> class.

Details:

We use a common method “secondaryAppbar()” in the appbar.

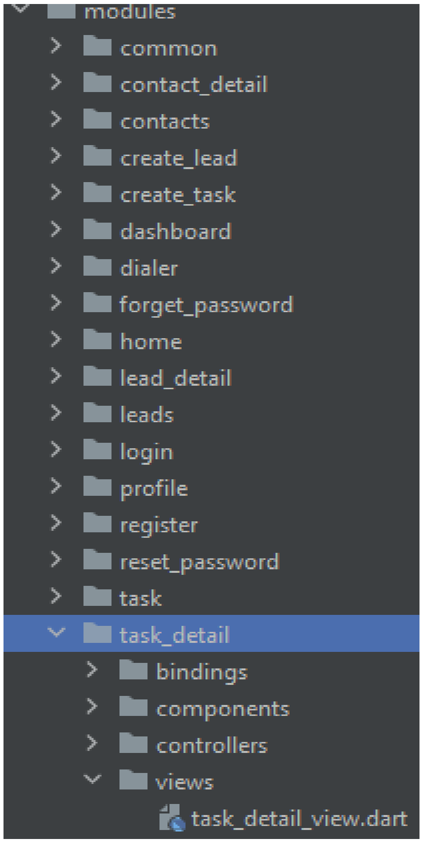
Location of the specified method is lib/app/theme/custom_appbar.dart

Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar.	

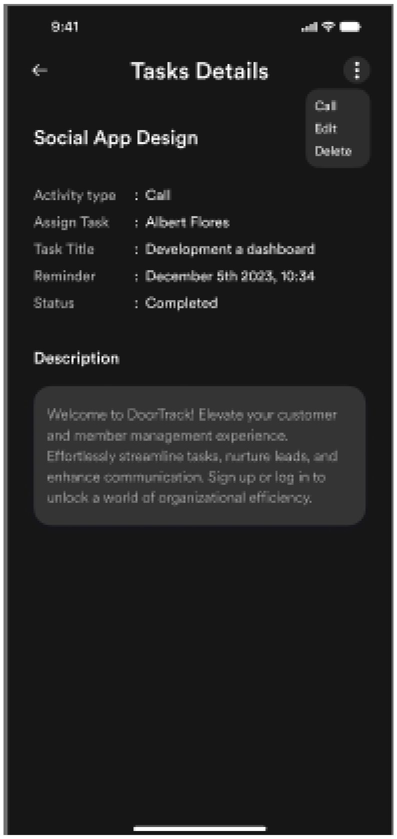
	lib/app/theme/custom_appbar.dart	
profileHeader()	Used to show user name, date of birth etc. lib/app/modules/contact_detail/component/profile_header.dart	
rowItem()	Used to show contacts info button such as email, whatsapp, call. lib/app/modules/contact_detail/component/row_item.dart	
verticalItem()	Used to show contacts info such as mobile, location, security number. lib/app/modules/contact_detail/component/vertical_item.dart	
notes()	Used to show buttons such as notes, files, tasks. lib/app/modules/contact_detail/component/notes.dart	
invoice()	Used to show invoice detail and share invoice button.	

	lib/app/modules/contact_detail/component/invoice.dart	

14. Task Detail



```
modules
  > common
  > contact_detail
  > contacts
  > create_lead
  > create_task
  > dashboard
  > dialer
  > forget_password
  > home
  > lead_detail
  > leads
  > login
  > profile
  > register
  > reset_password
  > task
  v task_detail
      > bindings
      > components
      > controllers
      v views
          task_detail_view.dart
```



Tasks Details

Social App Design

Activity type : Call
Assign Task : Albert Flores
Task Title : Development a dashboard
Reminder : December 5th 2023, 10:34
Status : Completed

Description

Welcome to DoorTrack! Elevate your customer and member management experience. Effortlessly streamline tasks, nurture leads, and enhance communication. Sign up or log in to unlock a world of organizational efficiency.

Class Name:

TaskDetailView that extends GetView<TaskDetailController> class.

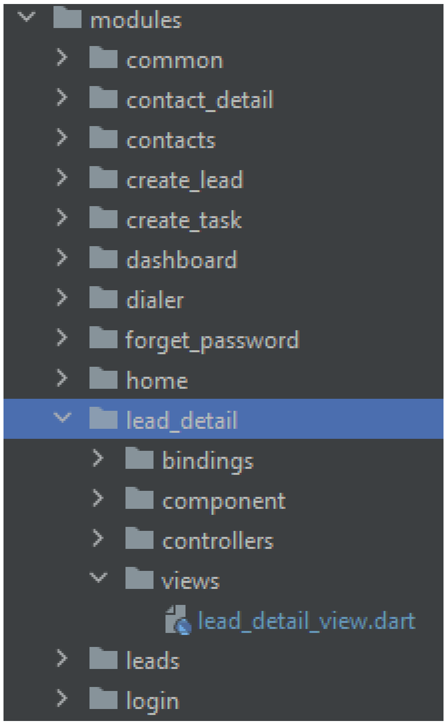
Details:

We use a common method “secondaryAppbar()” in the appbar.

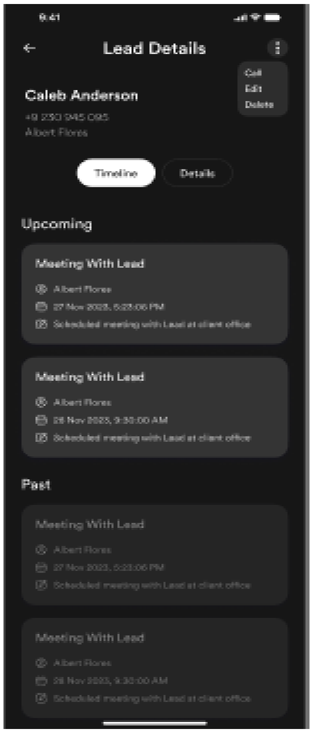
Location of the specified method is lib/app/theme/custom_appbar.dart

Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	
taskHeader()	Used to show user social app design. lib/app/modules/task_detail/component/task_header.dart	
taskDescription()	Used to show a text field that shows the description. And the text field is read only. lib/app/modules/task_detail/component/task_description.dart	

15. Lead Detail



```
modules
  common
  contact_detail
  contacts
  create_lead
  create_task
  dashboard
  dialer
  forget_password
  home
  lead_detail
    bindings
    component
    controllers
    views
      lead_detail_view.dart
  leads
  login
```



9:41

Lead Details

Call
Edit
Delete

Caleb Anderson

+9 230 945 095

Albert Flores

Timeline Details

Upcoming

Meeting With Lead

Albert Flores

27 Nov 2023, 5:23:06 PM

Scheduled meeting with Lead at client office

Meeting With Lead

Albert Flores

28 Nov 2023, 9:30:00 AM

Scheduled meeting with Lead at client office

Past

Meeting With Lead

Albert Flores

27 Nov 2023, 5:23:06 PM

Scheduled meeting with Lead at client office

Meeting With Lead

Albert Flores

28 Nov 2023, 9:30:00 AM

Scheduled meeting with Lead at client office

Class Name:

LeadDetailView that extends GetView<LeadDetailController> class.

Details:

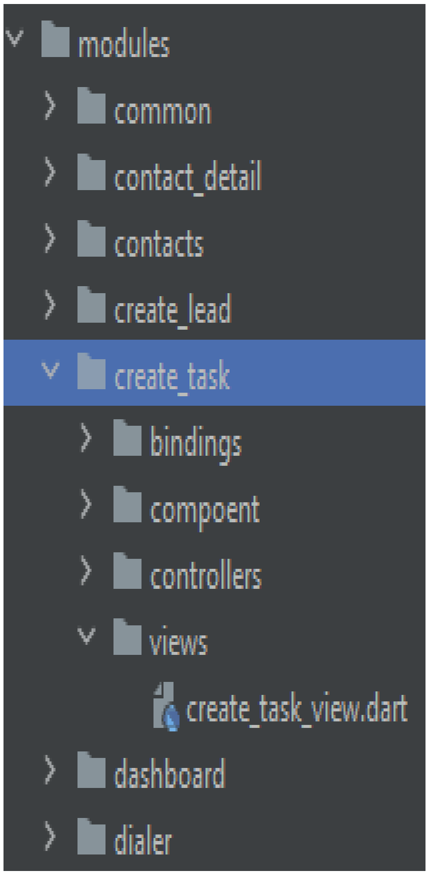
We use a common method “secondaryAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

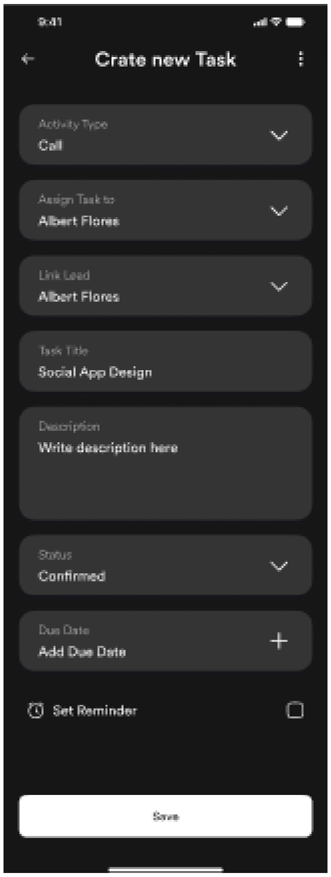
Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	
leadDetailHeader()	Used to show user name,	

	phone etc. lib/app/modules/lead_detail/component/lead_detail_header.dart	
primaryButton()	Used to show buttons, timeline and details. lib/app/theme/button_theme.dart	
getView()	This method is located inside the class. It shows two methods: timeline and detail.	
note()	Used to show notes. lib/app/modules/lead_detail/component/note.dart	

16. Create Task



```
modules
  common
  contact_detail
  contacts
  create_lead
  create_task
    bindings
    compoent
    controllers
    views
      create_task_view.dart
  dashboard
  dialer
```



9:41

Crate new Task

Activity Type: Call

Assign Task to: Albert Flores

Link Lead: Albert Flores

Task Title: Social App Design

Description: Write description here

Status: Confirmed

Due Date: Add Due Date +

Set Reminder □

Save

Class Name:

CreateTaskView that extends GetView<CreateTaskController> class.

Details:

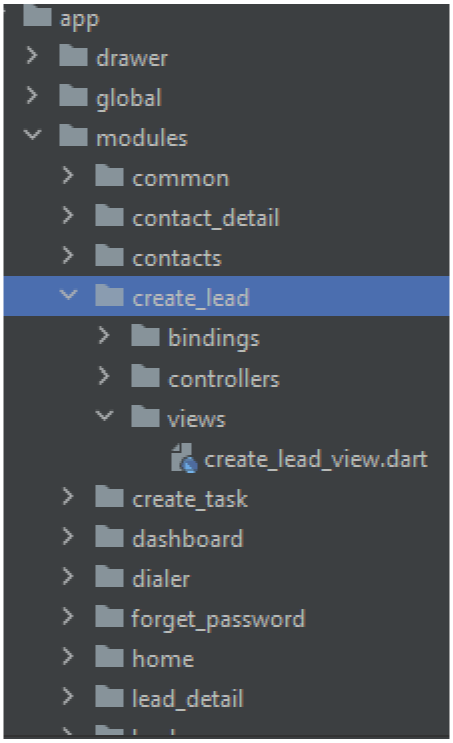
We use a common method “secondaryAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

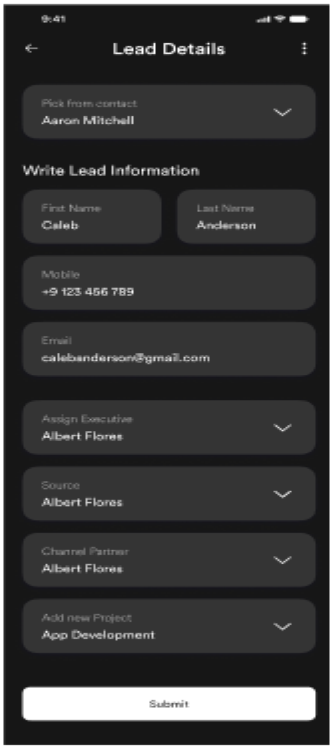
Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar.	

	lib/app/theme/custom_appbar.dart	
dropdownField()	Used to show dropdown. lib/app/modules/common/dropdown.dart	
textfieldWithTitle()	Used to show text fields. lib/app/modules/common/text_field_with_title.dart	
reminderField()	Used to show reminders. lib/app/modules/create_task/component/reminder_text_field.dart	
primaryButton()	Used to show button save. lib/app/theme/button_theme.dart	

17. Create Lead



```
app
  > drawer
  > global
  v modules
    > common
    > contact_detail
    > contacts
    v create_lead
      > bindings
      > controllers
      v views
          create_lead_view.dart
    > create_task
    > dashboard
    > dialer
    > forget_password
    > home
    > lead_detail
```



Pick from contact: Aaron Mitchell

Write Lead Information

First Name: Caleb

Last Name: Anderson

Mobile: +9 123 456 789

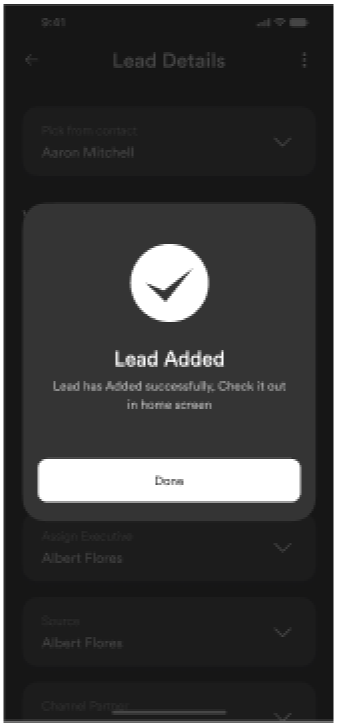
Email: calebanderson@gmail.com

Assign Executive: Albert Flores

Source: Albert Flores

Channel Partner: Albert Flores

Add new Project: App Development



Pick from contact: Aaron Mitchell

Lead Added

Lead has Added successfully, Check it out in home screen

Assign Executive: Albert Flores

Source: Albert Flores

Channel Partner

Class Name:

CreateLeadView that extends GetView<CreateLeadController> class.

Details:

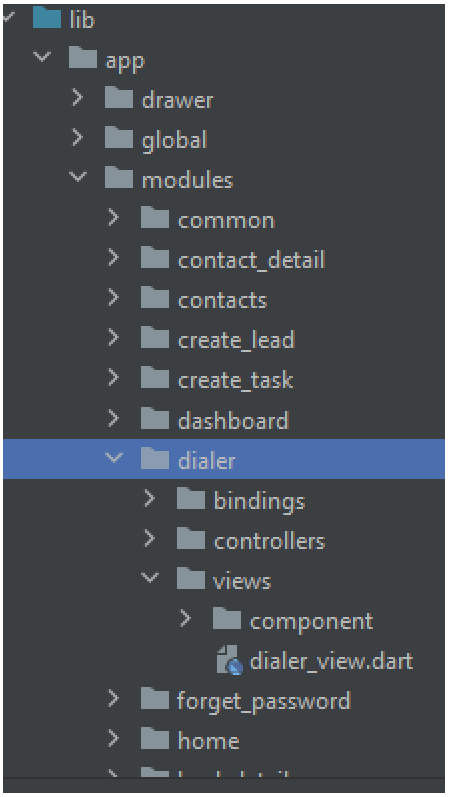
We use a common method “secondaryAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

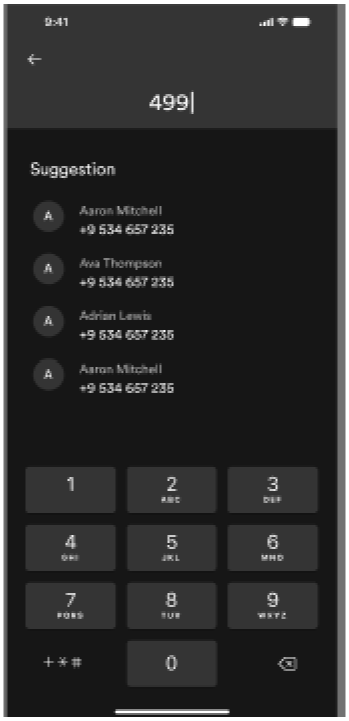
Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar. lib/app/theme/custom_appbar.dart	

dropdownField()	Used to show dropdown. lib/app/modules/common/dropdown.dart	
textfieldWithTitle()	Used to show text fields. lib/app/modules/common/text_field_with_title.dart	
primaryButton()	Used to show button save. lib/app/theme/button_theme.dart	

18. Dialer



```
lib
  app
    drawer
    global
    modules
      common
      contact_detail
      contacts
      create_lead
      create_task
      dashboard
      dialer
        bindings
        controllers
        views
          component
          dialer_view.dart
      forget_password
      home
```



Class Name:

DialerView that extends GetView<DialerController> class.

Details:

We use a common method “secondaryAppbar()” in the appbar.

Location of the specified method is lib/app/theme/custom_appbar.dart

Methods	Descriptions	Api
secondaryAppbar()	Used in the appbar.	

	lib/app/theme/custom_appbar.dart	
header()	Used to show header items. lib/app/modules/common/header.dart	
dialerItem()	Used to show dialer items. lib/app/modules/dialer/component/dialer_item.dart	