

# Introduction of POS SAAS WhatsApp Marketing panel

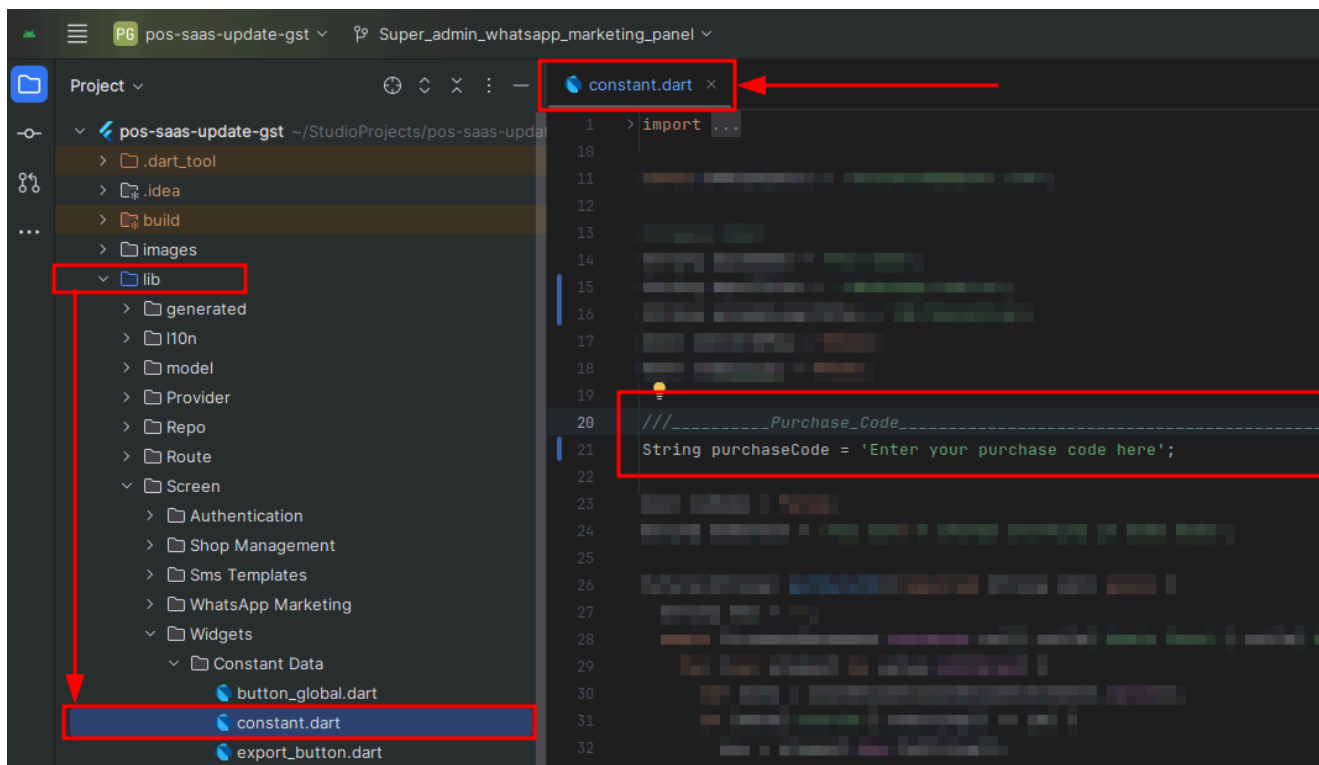
Thank you for purchasing our code. We are delighted to have you as a valued customer and are committed to providing you with exceptional service. If you experience any issues with the code or documentation, please do not hesitate to contact our Support Team through [Team](#) or [email](#).

We appreciate your trust in Acnoo and look forward to assisting you. 😊

## Prerequisite

- Make sure you have purchase our main product [POS SAAS -Flutter POS Billing Inventory Software with HRM App+Web+Super Admin](#)
- Flutter & Dart SDK (We have used Flutter version 3.24.3)
- Anyone IDE [Android Studio](#) (Recommended), [Visual Studio Code](#) or [IntelliJ IDEA](#)
- To edit this project you must have Flutter and Dart installed and configured successfully on your computer.
- Set up your editor – Install the [Flutter and Dart plugins](#).
- If you have got Android SDK installed and configured, to install Flutter you only need to:
  - Download Flutter SDK from the official website and extract it.
  - Add path to previously extracted SDK to your PATH variable
  - Run the **flutter doctor** tool to check if everything is configured correctly.
  - All above steps are mentioned here: <https://flutter.dev/docs/get-started/install/>

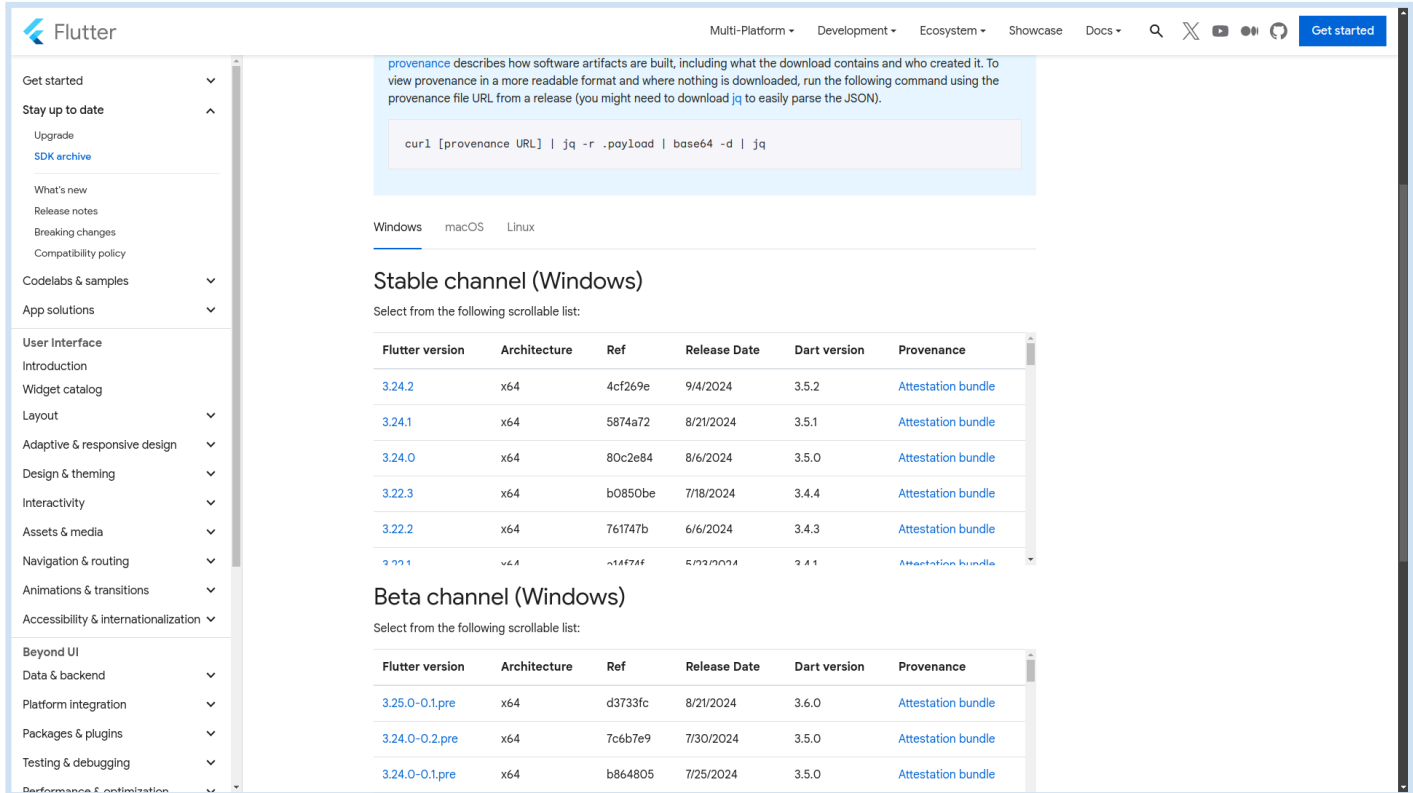
Make sure you have added the purchase code inside the app.



Directory: `/pos-saas-update-gst/lib/Screen/Widgets/Constant Data/constant.dart`

## Setup Flutter

Download the Flutter 3.24.3 version from the Flutter SDK Archive [link](#).



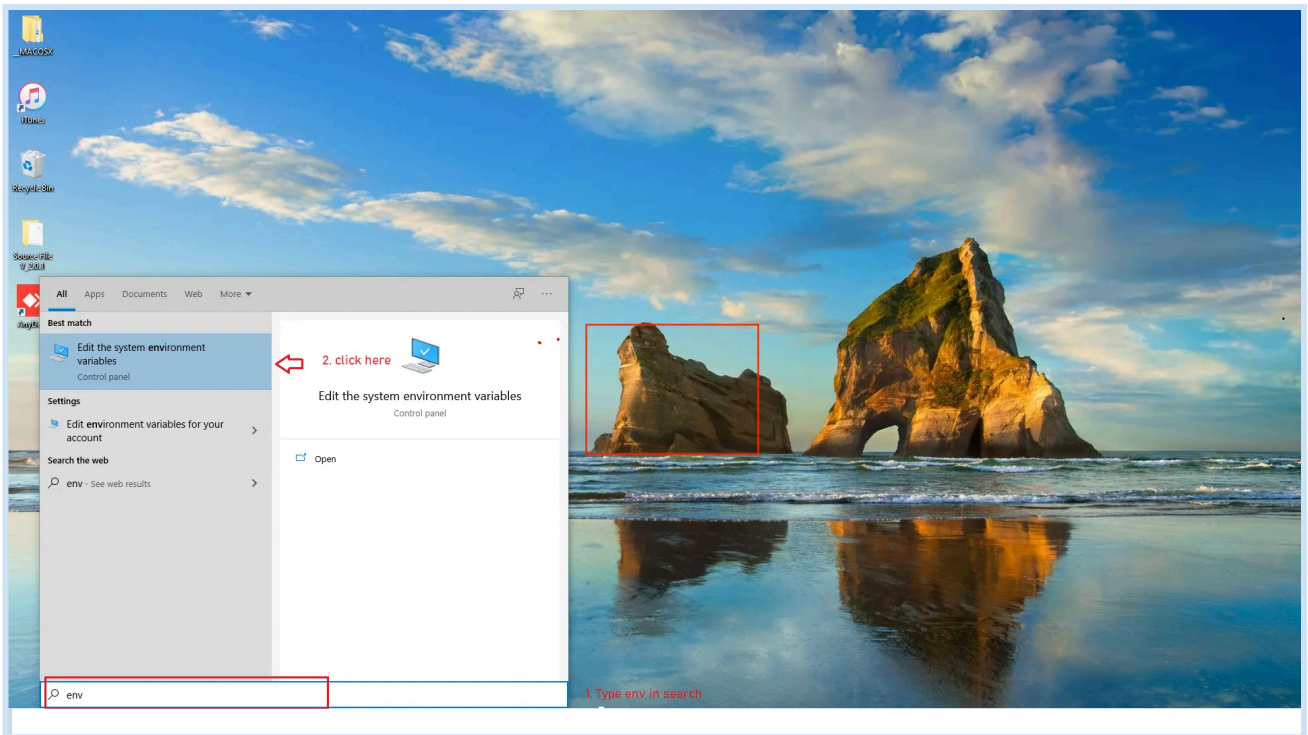
The screenshot shows the Flutter SDK Archive website. The main content area displays the "Stable channel (Windows)" section, which includes a table of Flutter versions and their corresponding Dart versions and provenance. The table is as follows:

Flutter version	Architecture	Ref	Release Date	Dart version	Provenance
3.24.2	x64	4cf269e	9/4/2024	3.5.2	<a href="#">Attestation bundle</a>
3.24.1	x64	5874a72	8/21/2024	3.5.1	<a href="#">Attestation bundle</a>
3.24.0	x64	80c2e84	8/6/2024	3.5.0	<a href="#">Attestation bundle</a>
3.22.3	x64	b0850be	7/18/2024	3.4.4	<a href="#">Attestation bundle</a>
3.22.2	x64	761747b	6/6/2024	3.4.3	<a href="#">Attestation bundle</a>
3.22.1	x64	~14F74E	5/22/2024	3.4.1	<a href="#">Attestation bundle</a>

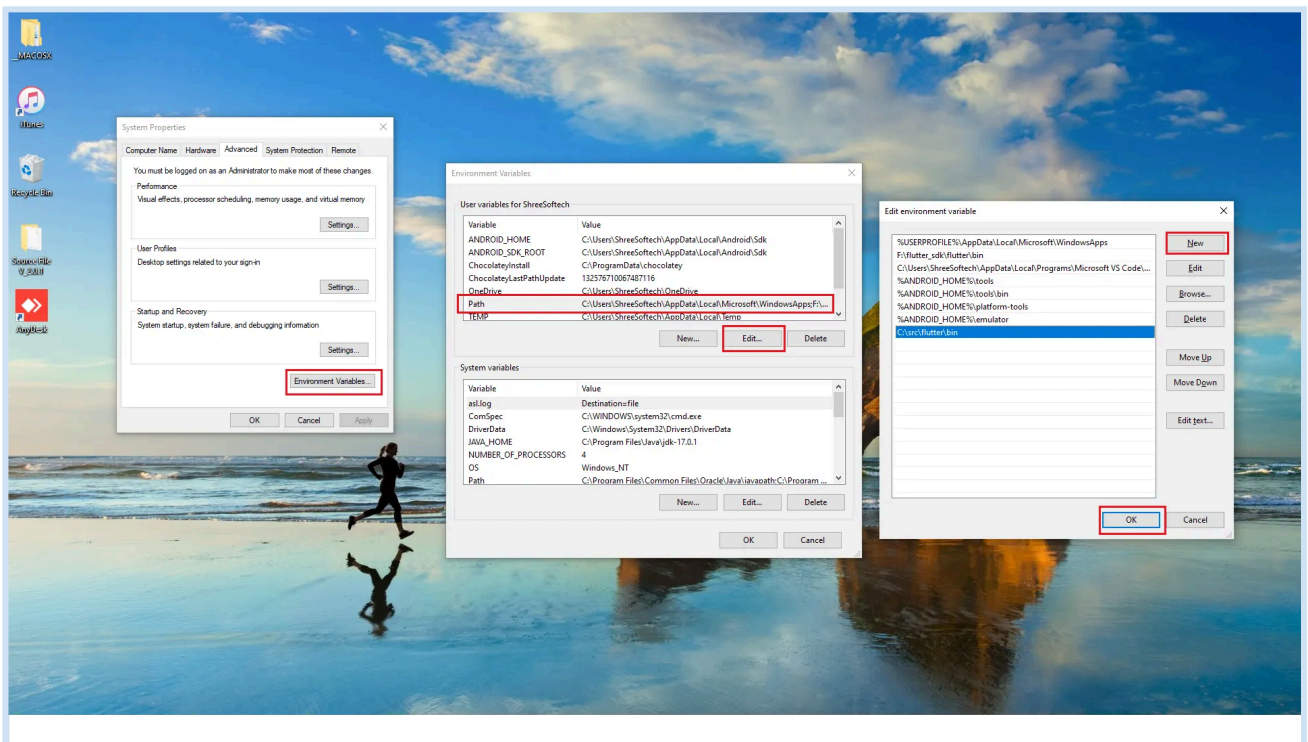
Below the stable channel table, the "Beta channel (Windows)" section is also visible, showing a table of beta versions:

Flutter version	Architecture	Ref	Release Date	Dart version	Provenance
3.25.0-0.1.pre	x64	d3733fc	8/21/2024	3.6.0	<a href="#">Attestation bundle</a>
3.24.0-0.2.pre	x64	7c6b7e9	7/30/2024	3.5.0	<a href="#">Attestation bundle</a>
3.24.0-0.1.pre	x64	b864805	7/25/2024	3.5.0	<a href="#">Attestation bundle</a>

1. Please unzip the downloaded file and move the Flutter folder to your desired installation directory for the Flutter SDK (e.g., `C:\flutter`). We recommend avoiding locations like `C:\Program` Files to prevent potential permission issues.
2. To add Flutter to your environment path, start by searching for "Environment" in the Start menu and select "Edit the system environment variables." In the System Properties window that opens, click on "Environment Variables." Under the "System variables" section, find and select the "Path" variable, then click "Edit." In the Edit Environment Variable window, click "New" and add the path to the Flutter `bin` directory (e.g., `C:\flutter\bin`). Once added, click "OK" to save the changes and close all the windows. This process will ensure that Flutter commands can be executed from the command line.



1. Within the "User variables" section, locate the Path environment variable and click on "Edit." In the dialog box that appears, click "New" and paste the path to the `\bin` directory of your Flutter installation (e.g., `C:\flutter\bin`). After adding the path, click "OK" to close each of the dialog boxes and apply the changes. This will complete the setup of the Flutter path in your environment variables.



To verify that everything is installed correctly, open the Command Prompt and run the command **flutter doctor**. This command will check your Flutter installation and provide a report on the status of your environment. Ensure that there are no critical errors, and follow any suggested steps to resolve warnings or issues.

```
Last login: Thu Sep 5 10:02:14 on ttys003
[tahmidtarongo@Tahmids-MacBook-Air ~ % flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.24.1, on macOS 14.6.1 23G93 darwin-arm64, locale en-BD)
[✓] Android toolchain - develop for Android devices (Android SDK version 35.0.0)
[✓] Xcode - develop for iOS and macOS (Xcode 15.4)
[✓] Chrome - develop for the web
[✓] Android Studio (version 2024.1)
[✓] Connected device (4 available)
[✓] Network resources

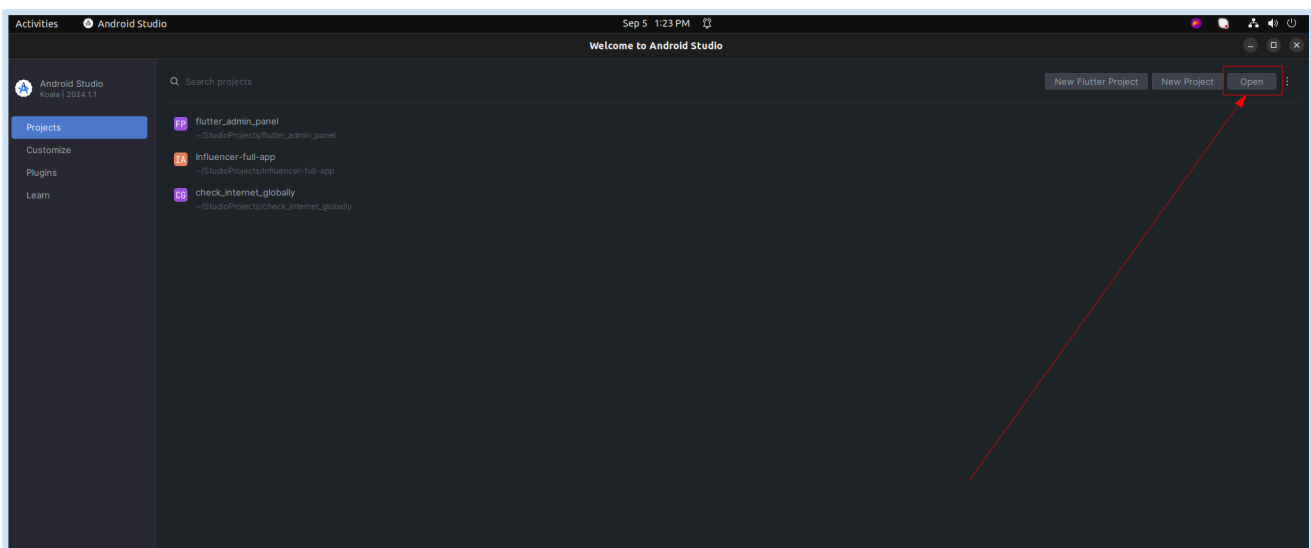
• No issues found!
tahmidtarongo@Tahmids-MacBook-Air ~ %
```

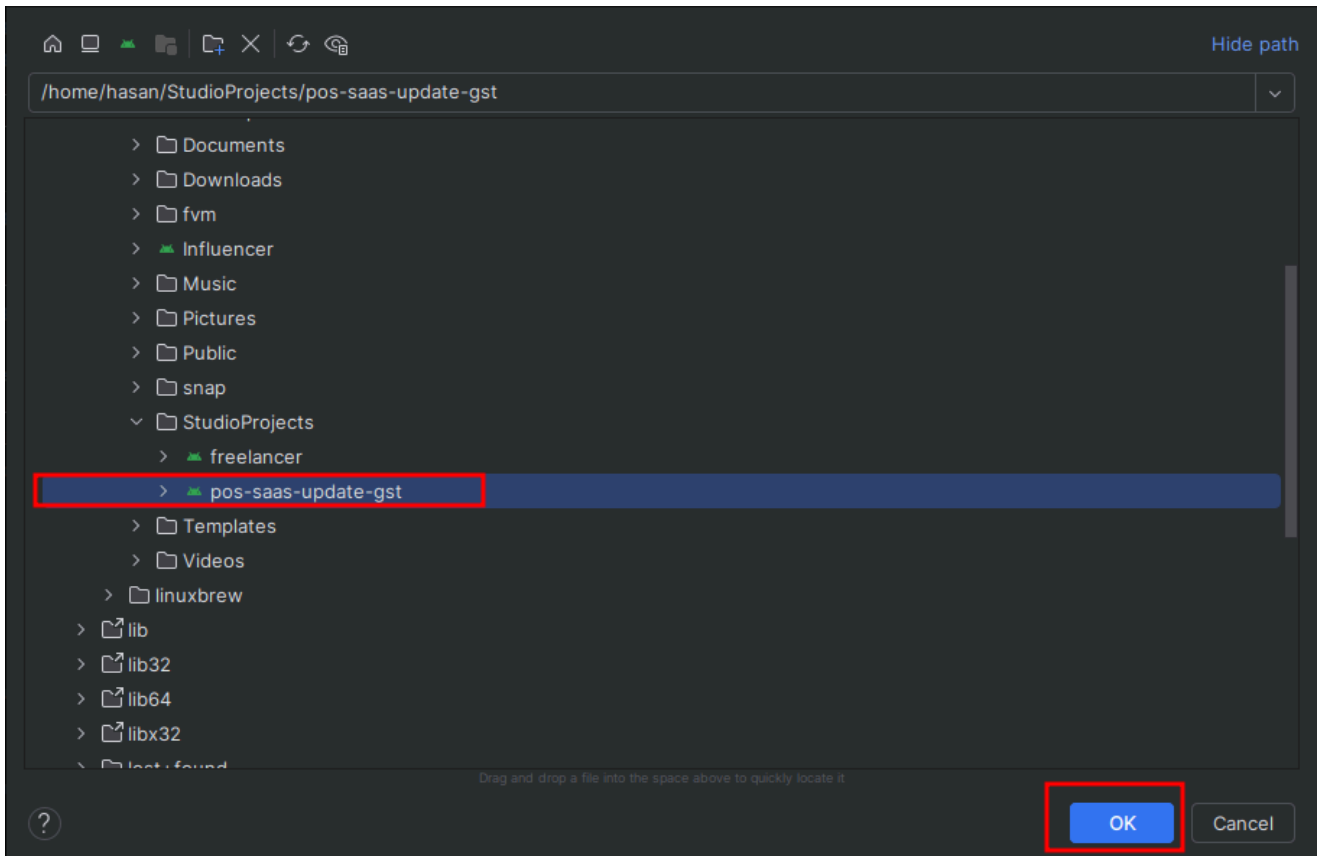
- Explore the official Flutter website at <https://docs.flutter.dev/get-started/install> for a complete installation guide.
- For video tutorials, Check out the playlist:

📺 How to Install and Setup Flutter for App Development on Mac - Part 2

## Setup Android Studio

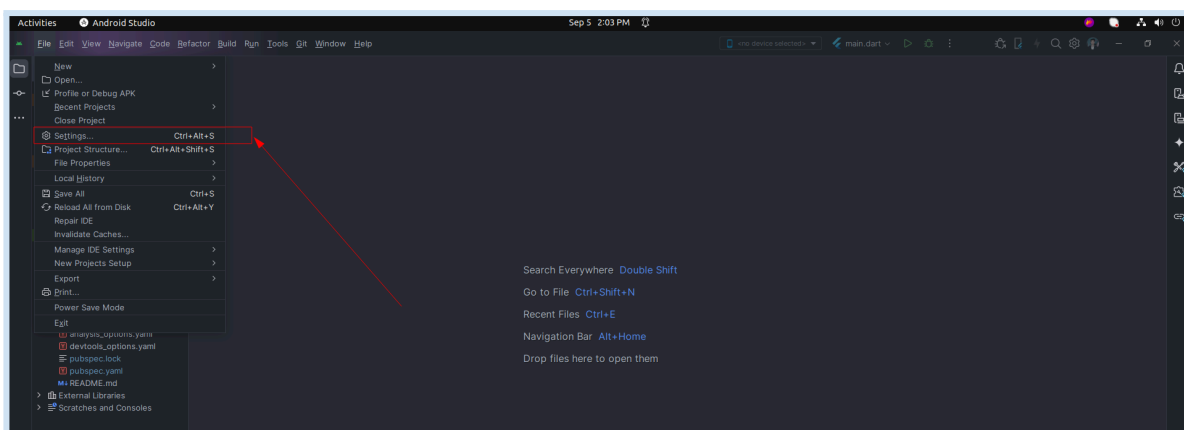
1. Launch Android Studio and select the 'Open' option from the welcome screen. Navigate to and select the '**POS SAAS WA Marketing**' folder that you obtained from Codecanyon. This will open the project in Android Studio, allowing you to start working on it.



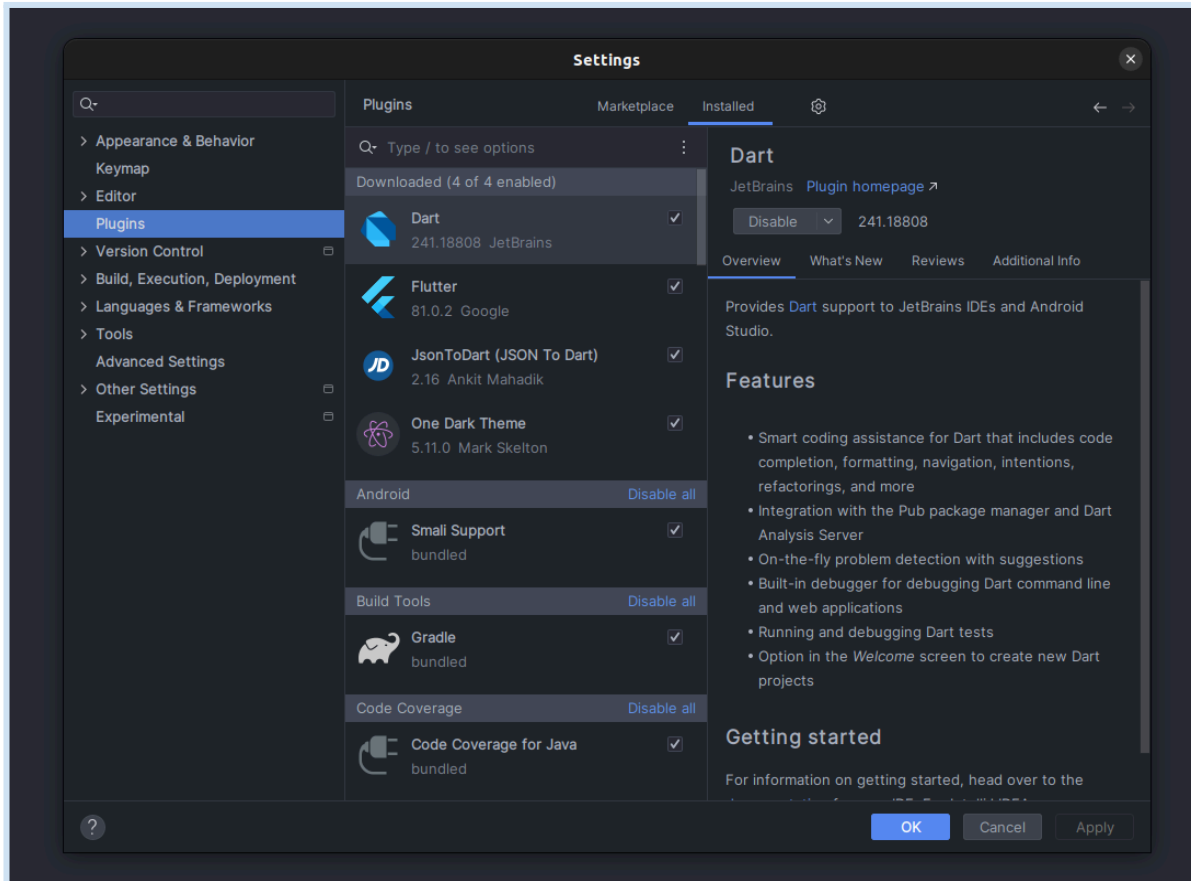


2. If you are creating or opening a Flutter project in Android Studio for the first time, you'll need to install the Flutter plugin and configure the Flutter and Dart SDK paths. To do this, open Android Studio and navigate to "Plugins" from the settings menu. Search for and install the Flutter plugin, which will automatically include the Dart plugin as well. After installing, set the path to your Flutter SDK under the Flutter plugin settings, ensuring that the Dart SDK path is also correctly configured. This setup is essential for developing Flutter projects within Android Studio.

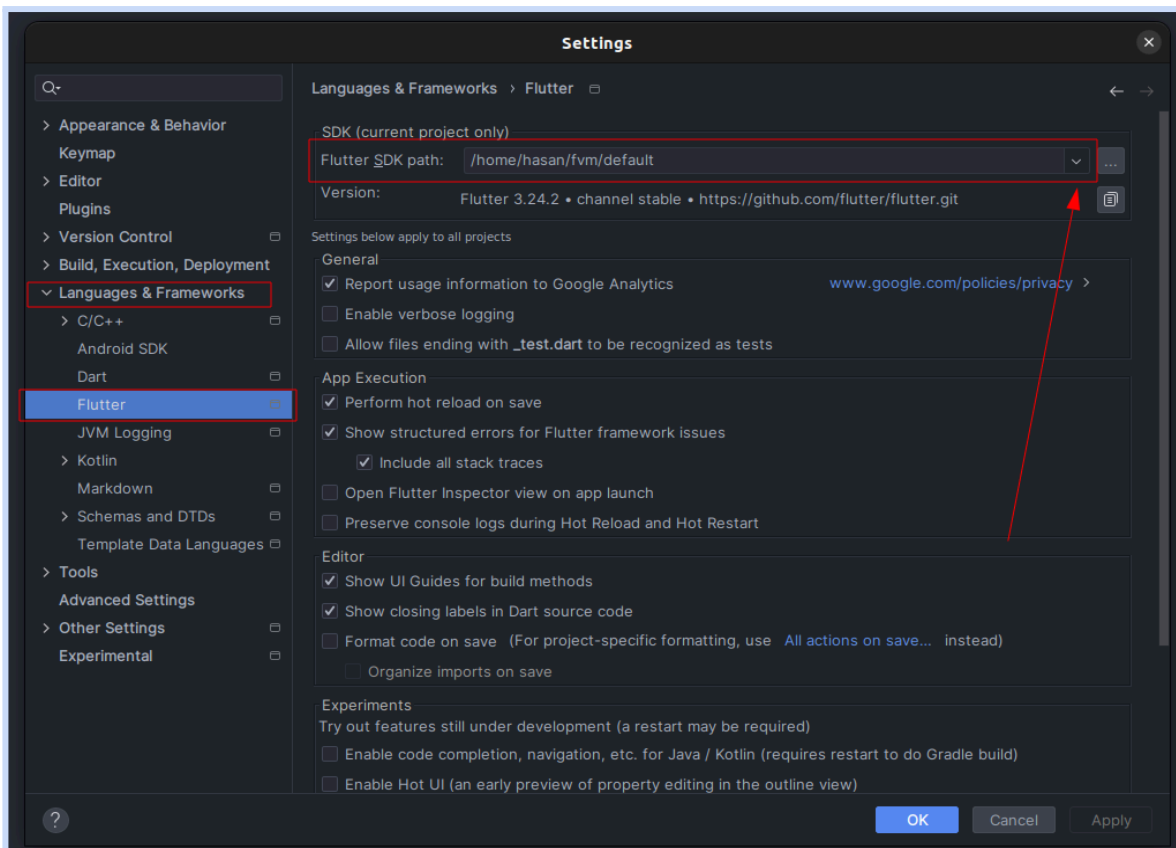
➤ Open File->Settings, then go to **Plugins**



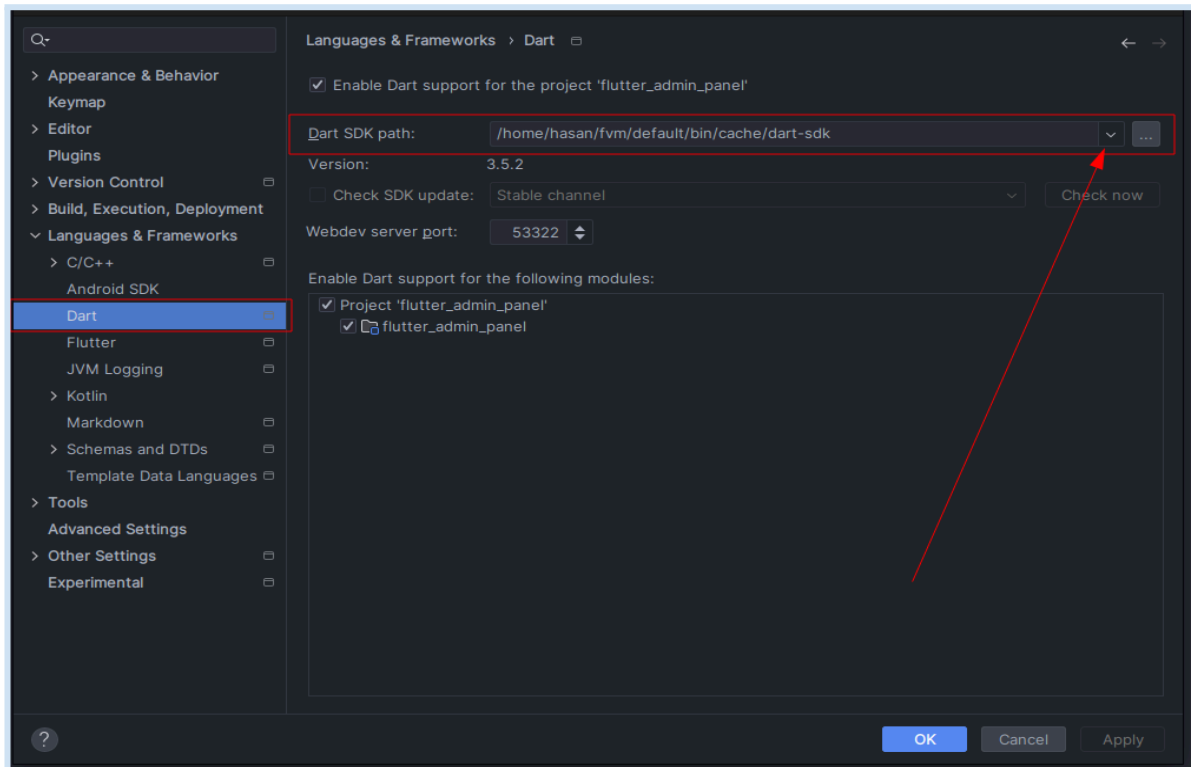
- And Install flutter, dart, and kotlin plugins.



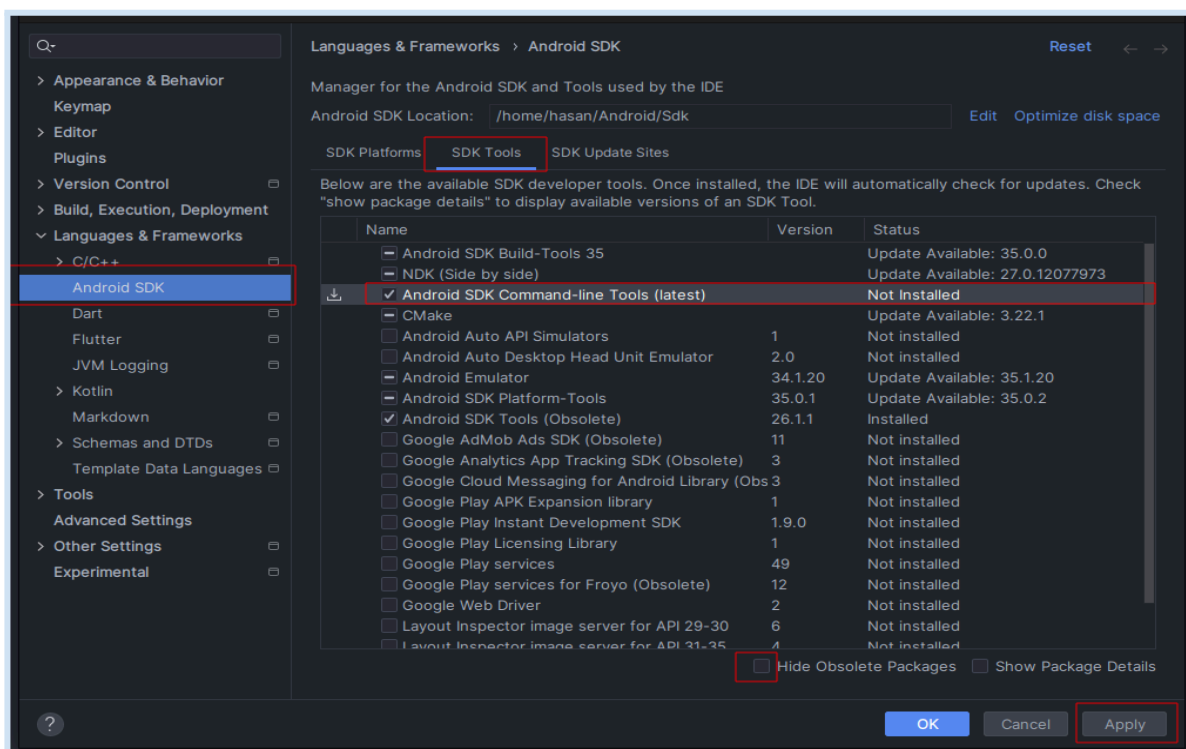
- In Android Studio, open the settings or preferences menu again. In the left pane, select 'Language and Frameworks,' then click on 'Flutter.' In the right panel, set the path to your Flutter SDK by selecting the directory where Flutter is installed (e.g., `C:\flutter`).



- Now dart sdk should automatically be selected, if not select the flutter path and add `\bin\cache\dart-sdk` as shown in image.

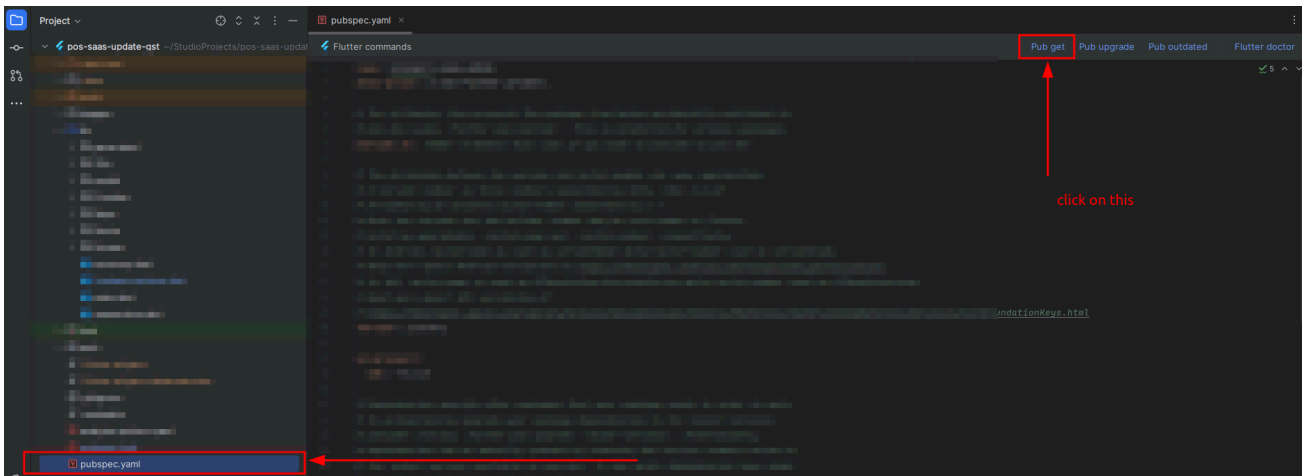


- To configure the Android SDK, navigate to the SDK Manager located in the upper right corner of your screen. Next, go to the "Android SDK" section and uncheck the 'Hide Obsolete Packages' option. From the available packages, select 'Android SDK Command Line Tool Latest.' Click 'Apply' to confirm your selections, and then click 'OK' to start downloading the latest version. Please refer to the image below for guidance.

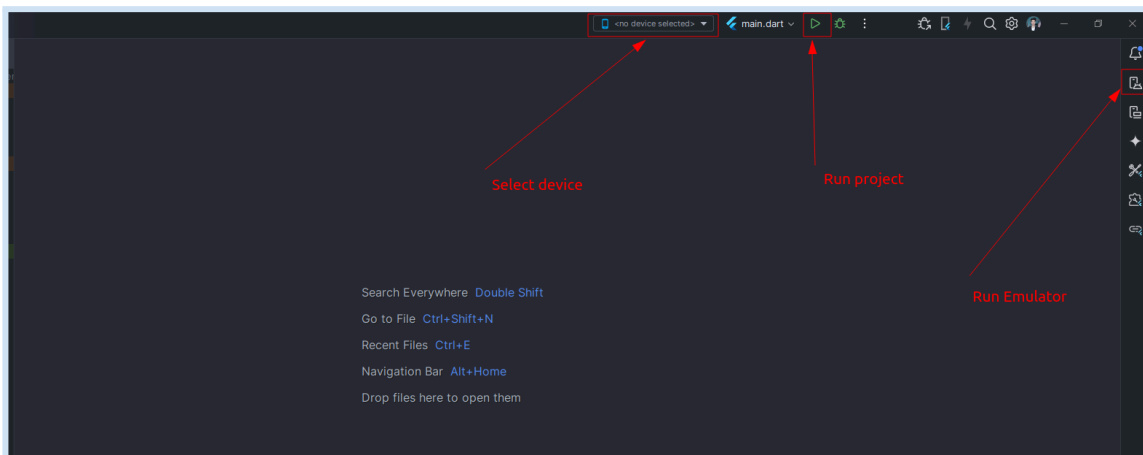


## Run Flutter Project

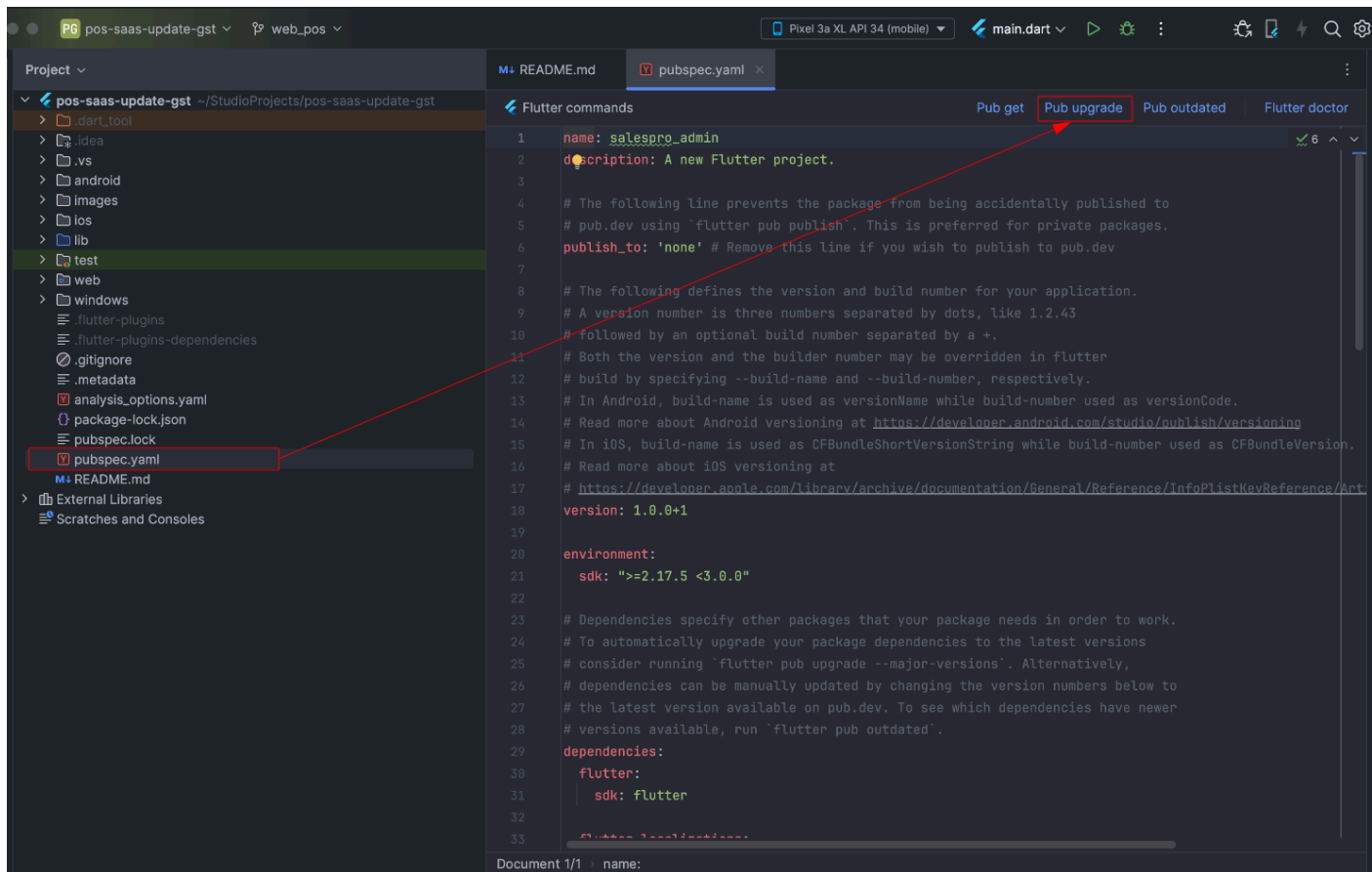
- For first-time users, enable Dart support for the project. Afterward, navigate to the **pubspec.yaml** file and retrieve the packages by clicking on "Pub Get" located at the top right corner, as illustrated in the image.



- After enabling Dart support and fetching packages, select a device or run an emulator. Click the 'Play' icon in the upper-right corner to launch the app. For physical devices, enable 'Developer Options' and 'USB Debugging' on your phone.



# How to update this Project Version



To update your project version, go to the **pubspec.yaml** file. And click on the **Pub upgrade** button at the top. It will automatically update all the packages version. Make sure your pc is connected with the internet.

To update your project version and dependencies in a Flutter project, follow these steps:

## 1. Open the **pubspec.yaml** File:

- Locate the **pubspec.yaml** file in the root directory of your Flutter project.
- Open it in your preferred text editor or IDE.

## 2. Update the Project Version:

- Find the **version** field at the top of the **pubspec.yaml** file.
- Change the version number to your desired version. For example: `version: 1.1.0+2`
- The version format is **major.minor.patch+build**.

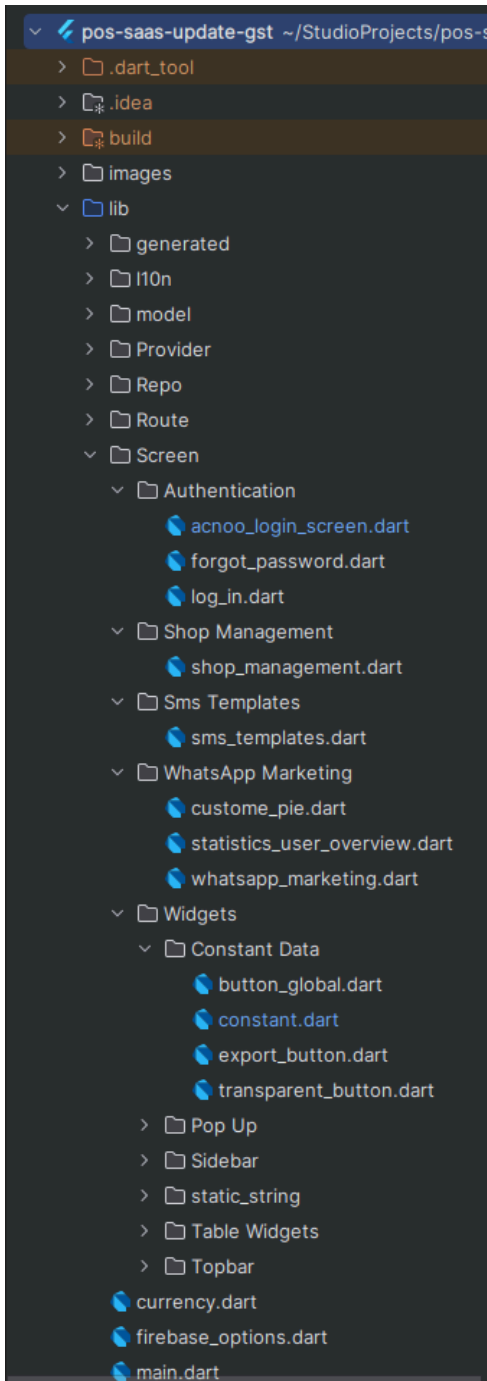
## Run **pub upgrade**:

- Ensure your computer is connected to the internet.
- In your IDE, locate and click the **Pub upgrade** button. This button is typically found at the top of the editor or in the **pubspec.yaml** tab.
- Alternatively, you can run the following command in your terminal from the project directory:

```
flutter pub upgrade
```

This command updates all packages to the latest versions allowed by the constraints in **pubspec.yaml**.

Project Structure:



- **Images:** Apps/widgets/snippets images as well as applcon are stored here.
- **Lib:** Applications main files and directories are located here.
  - **Screen:** Full app's UI codes are stored here.
  - **Widgets:** All the reusable widgets are stored here, like(Top bar, Side bar).Also the constant file located in the folder. Where the all the colors are stored which is used to design the app.
  - In **firebase\_option.dart** all the code stored which is related with the firebase.From here you can easily change the firebase credentials.
  - **route:** All the routes have been declared here.

## Firestore Setup

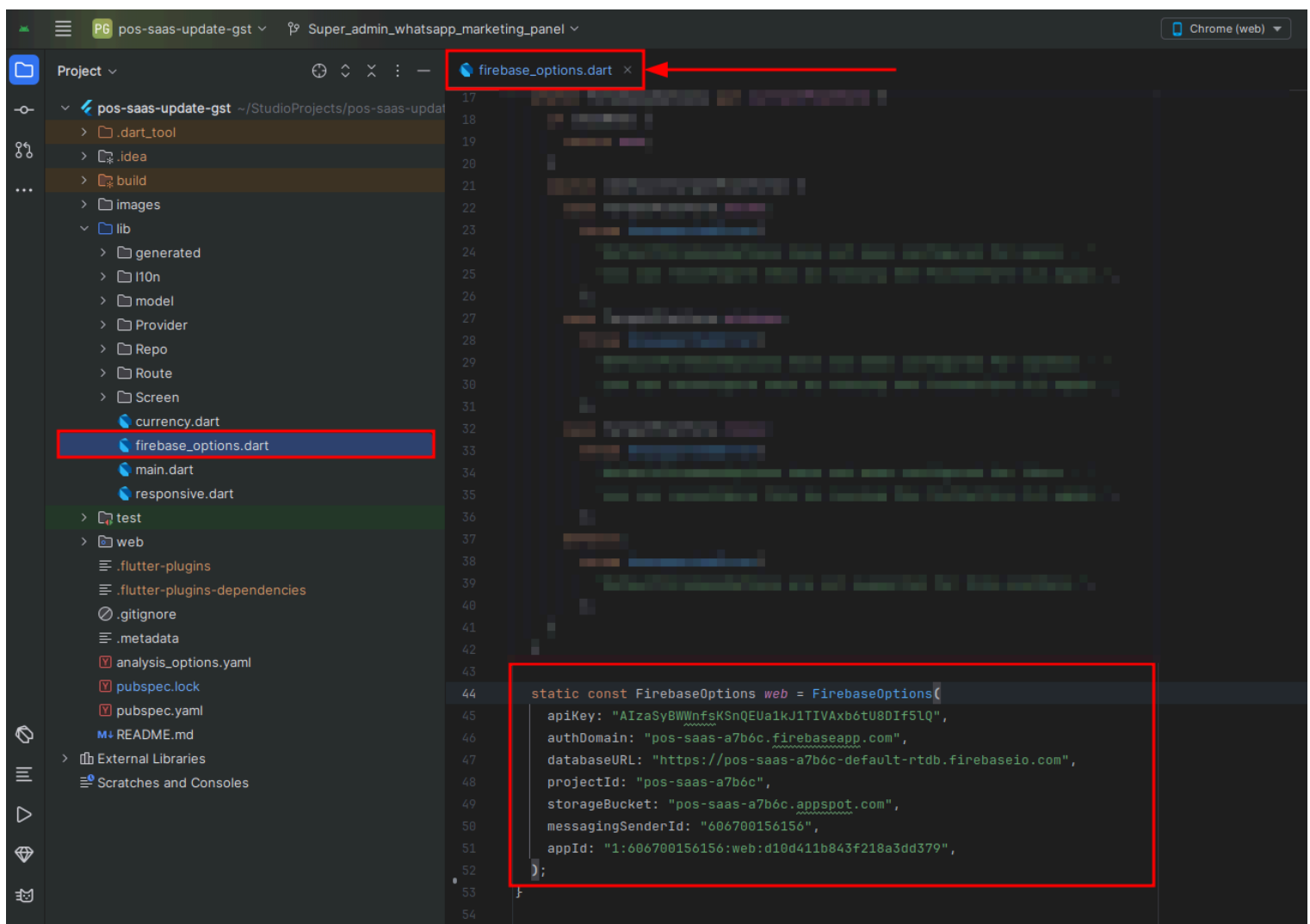
To set up Firestore, follow these steps:

1. Open the `firebase_options.dart` file in your project.
2. Locate the section where you need to add your Firestore credentials.

path: `/pos-saas-update-gst/lib/firebase_options.dart`

3. Paste the credentials you used for the **POS SAAS Super Admin** into the appropriate location in the file.

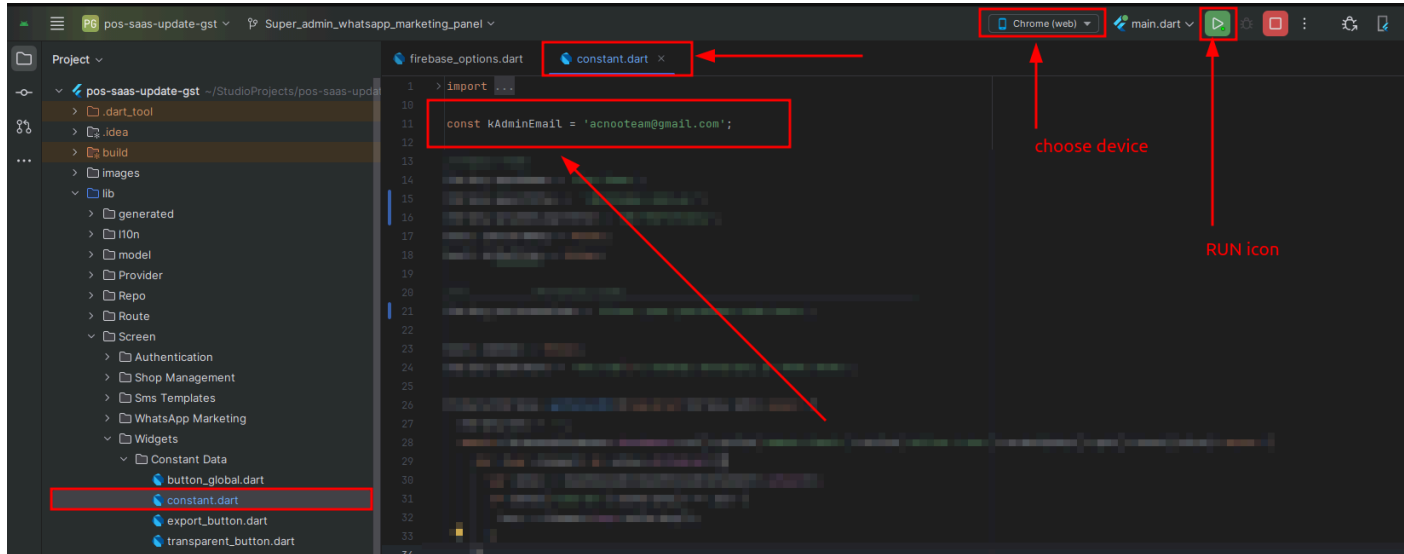
Make sure to save your changes after pasting the credentials. This will ensure that your application is properly configured to use Firestore services.



```
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44 static const FirebaseOptions web = FirebaseOptions(
45   apiKey: "AIzaSyBWWnfsKSnQEua1kJ1TIVAx6tU8DI5LQ",
46   authDomain: "pos-saas-a7b6c.firebaseio.com",
47   databaseURL: "https://pos-saas-a7b6c-default-rtdb.firebaseio.com",
48   projectId: "pos-saas-a7b6c",
49   storageBucket: "pos-saas-a7b6c.appspot.com",
50   messagingSenderId: "606700156156",
51   appId: "1:606700156156:web:d10d411b843f218a3dd379",
52 );
53
54
```

Once you've made these changes, save the file. Your Firestore setup will be complete!

## Add Login Credential



To update the email address in your project, follow these steps:

1. Navigate to the `constant.dart` file at the following location:  
`Project/lib/screen/widgets/Constant Data/constant.dart`.
2. Open the `constant.dart` file.
3. Locate the line containing the email address `acnooteam@gmail.com`.
4. Replace it with the email address you used to create the POS SAAS Super Admin.

Make sure the email address you entered is valid and exactly matches the one you used. After making the changes, save the file.

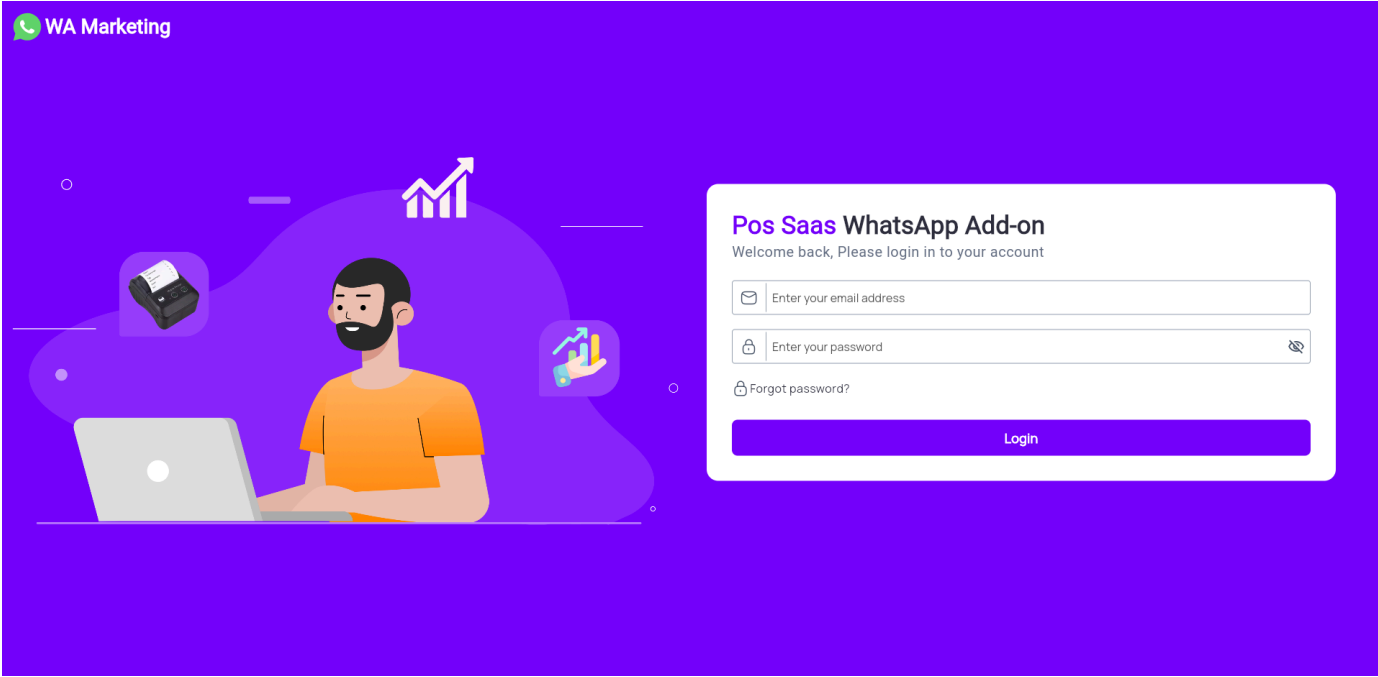
Now run the app.

To run the app, follow these steps:

1. In your development environment, locate the top bar.
2. Select the platform you want to run the application on (Web).
3. Click the "Run" button.

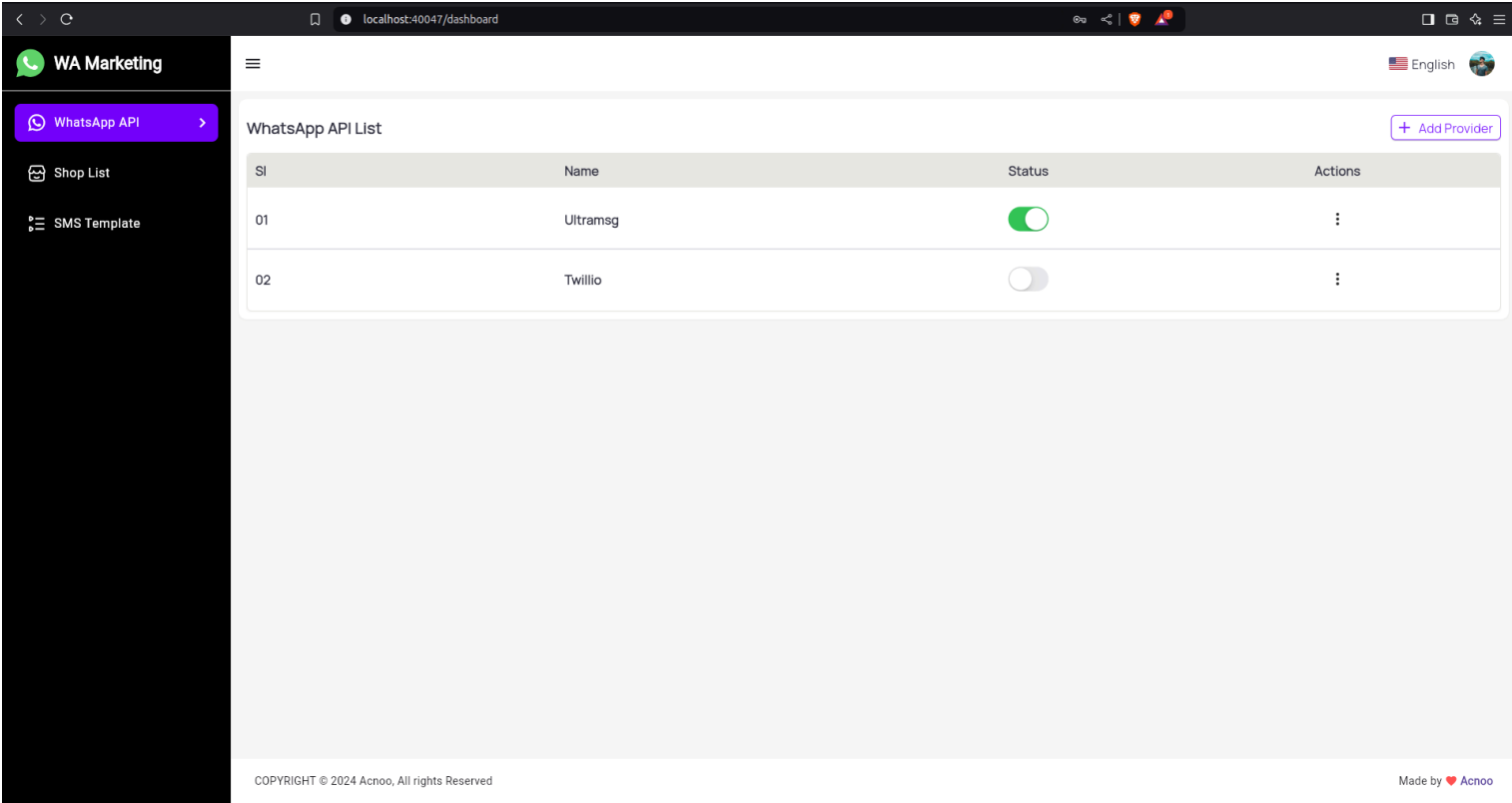
This will compile and launch your application on the selected platform. Make sure to check for any errors in the console during the process!

On the login screen, enter the credentials you pasted or used for the Super Admin account. Click the "Login" button.

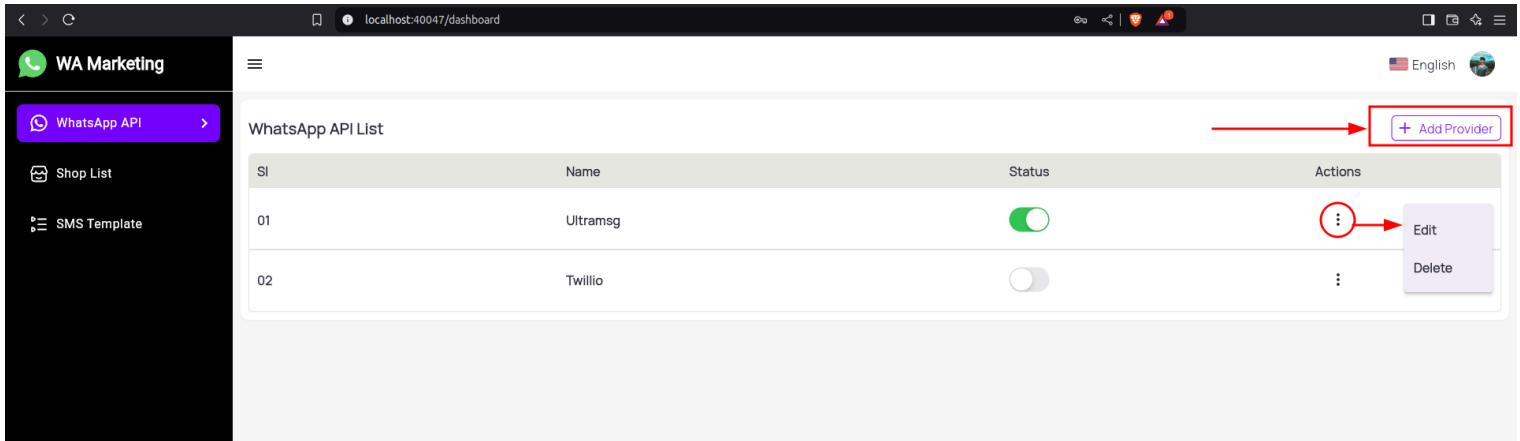


Once you've entered the correct credentials, you should be successfully logged into your account.

An environment will open like this,



## Add SMS gateway



Initially, there were two providers added: Ultramsmsg and Twilio. You can manage these providers as follows:

- Edit or Delete Providers:**
  - Click on the popup menu icon next to the provider you wish to modify.
  - From the menu, you can choose to edit the provider's details or delete it.
- Add a New Provider:**
  - Tap the "Add Provider" icon to create a new provider.

**We recommend that you edit the existing providers to better customize your setup.**

The screenshot shows the 'Edit UltraMsg credentials' form. The form has the following fields and values:

- API URL: `https://api.ultramsmsg.com/instance6169/`
- Phone Number: `+8801712022529`
- Instance ID: `instance9169`
- Token: `hwkfqekzwwfprwlp1`

At the bottom of the form, there are two buttons: 'Cancel' and 'Update'. The 'Update' button is highlighted with a red box.

To edit the credentials for Ultramsmsg, follow these steps:

- Locate the edit icon next to the Ultramsmsg provider.
- Click on the edit icon to open the credential fields.
- Replace the existing credentials with your own.
- Ensure that you enter valid values for each field.

After making your changes, be sure to save them to update the provider's credentials successfully.

For Twilio

**Edit Twilio credentials** ×

API URL

Phone Number  
456453

Account SID

Auth Token  
465

Cancel Update

To edit the credentials for Twilio, follow these steps:

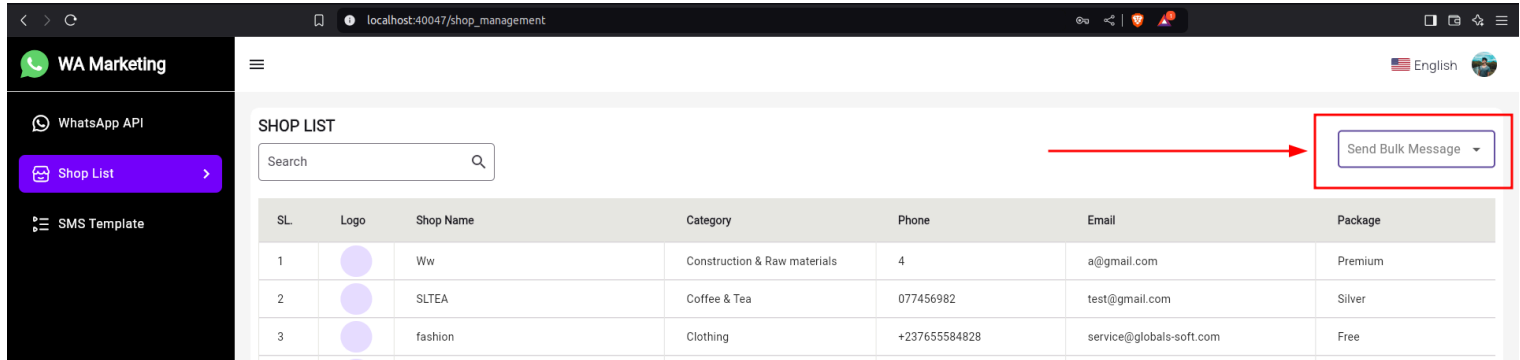
5. Locate the edit icon next to the Twilio provider.
6. Click on the edit icon to open the credential fields.
7. Replace the existing credentials with your own.
8. Ensure that you enter valid values for each field.

After making your changes, be sure to save them to update the provider's credentials successfully.

## Shop list screen

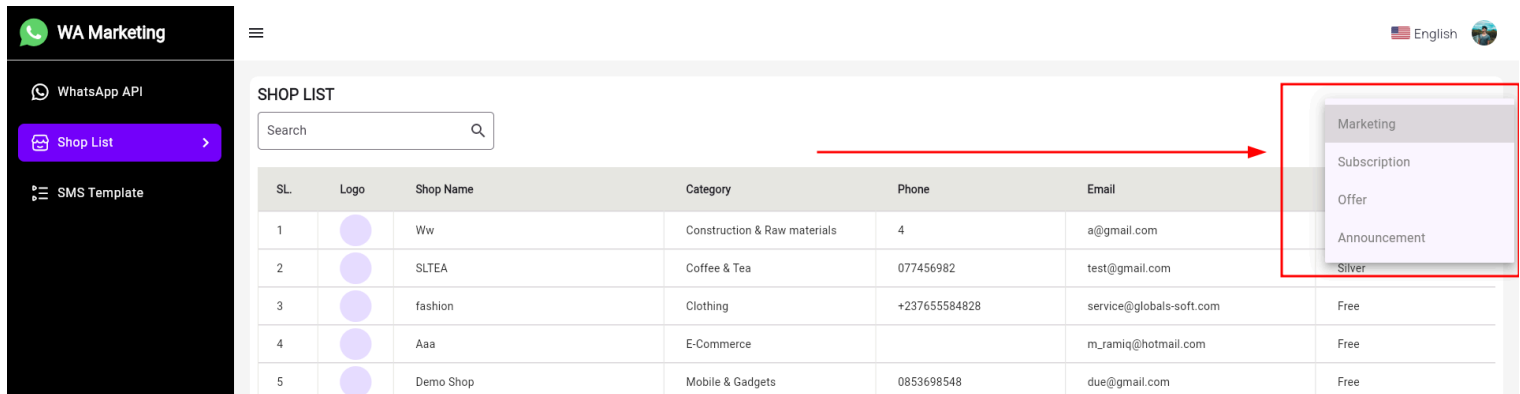
In this section, you can view the list of shops associated with your account.

- **Dropdown Menu:** In the top right corner of the screen, there is a dropdown button.
- You can select from the following options: **Marketing**, **Subscription**, **Announcement**, and **Offer**.
- Once you select an option, it will automatically send your message to the owners of the shops in the list.
- The owners will receive the message tailored to the selected option and the content you set for that section.



The screenshot shows the 'SHOP LIST' interface. At the top right, there is a 'Send Bulk Message' dropdown menu, highlighted with a red box and an arrow. Below the search bar is a table with the following data:

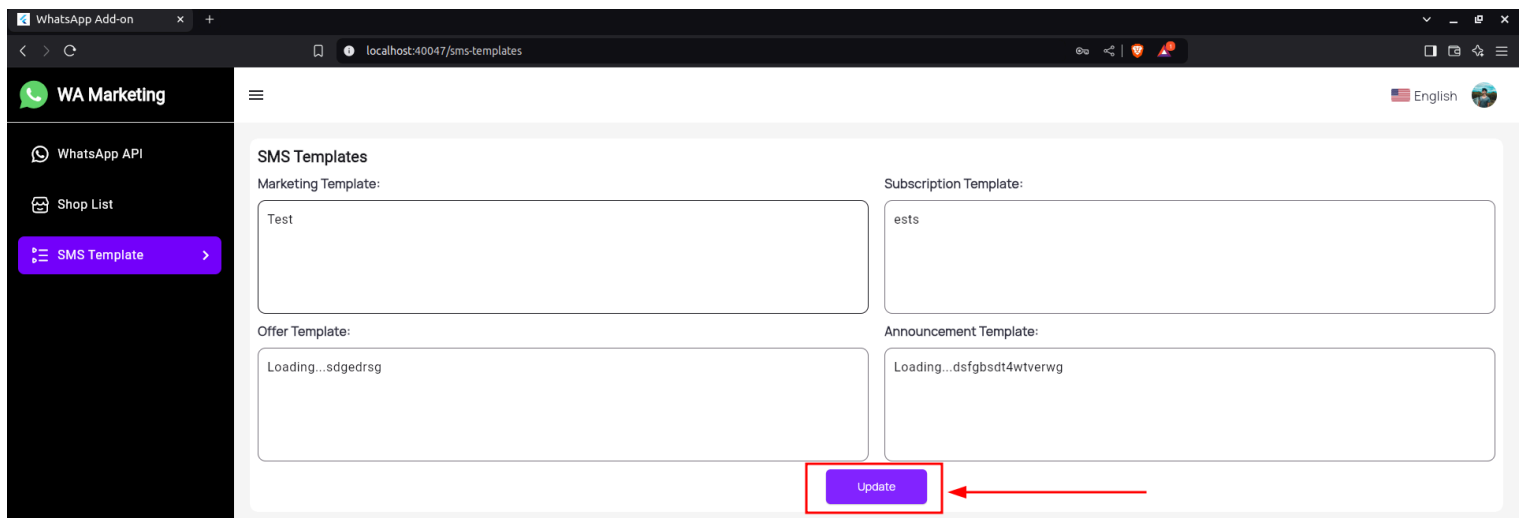
SL.	Logo	Shop Name	Category	Phone	Email	Package
1		Ww	Construction & Raw materials	4	a@gmail.com	Premium
2		SLTEA	Coffee & Tea	077456982	test@gmail.com	Silver
3		fashion	Clothing	+237655584828	service@globals-soft.com	Free



The screenshot shows the 'SHOP LIST' interface with the dropdown menu open. The menu options are: Marketing, Subscription, Offer, Announcement, and Silver. The table below contains 5 rows of shop data:

SL.	Logo	Shop Name	Category	Phone	Email	Package
1		Ww	Construction & Raw materials	4	a@gmail.com	
2		SLTEA	Coffee & Tea	077456982	test@gmail.com	
3		fashion	Clothing	+237655584828	service@globals-soft.com	Free
4		Aaa	E-Commerce		m_ramiq@hotmail.com	Free
5		Demo Shop	Mobile & Gadgets	0853698548	due@gmail.com	Free

You can customize the content for your selected item from the SMS Template screen.



The screenshot shows the 'SMS Templates' interface with four input fields for different template types and an 'Update' button at the bottom, highlighted with a red box and an arrow. The templates are:

- Marketing Template: Test
- Subscription Template: ests
- Offer Template: Loading...sdgedrsg
- Announcement Template: Loading...dsfgbsdt4wtvverwg

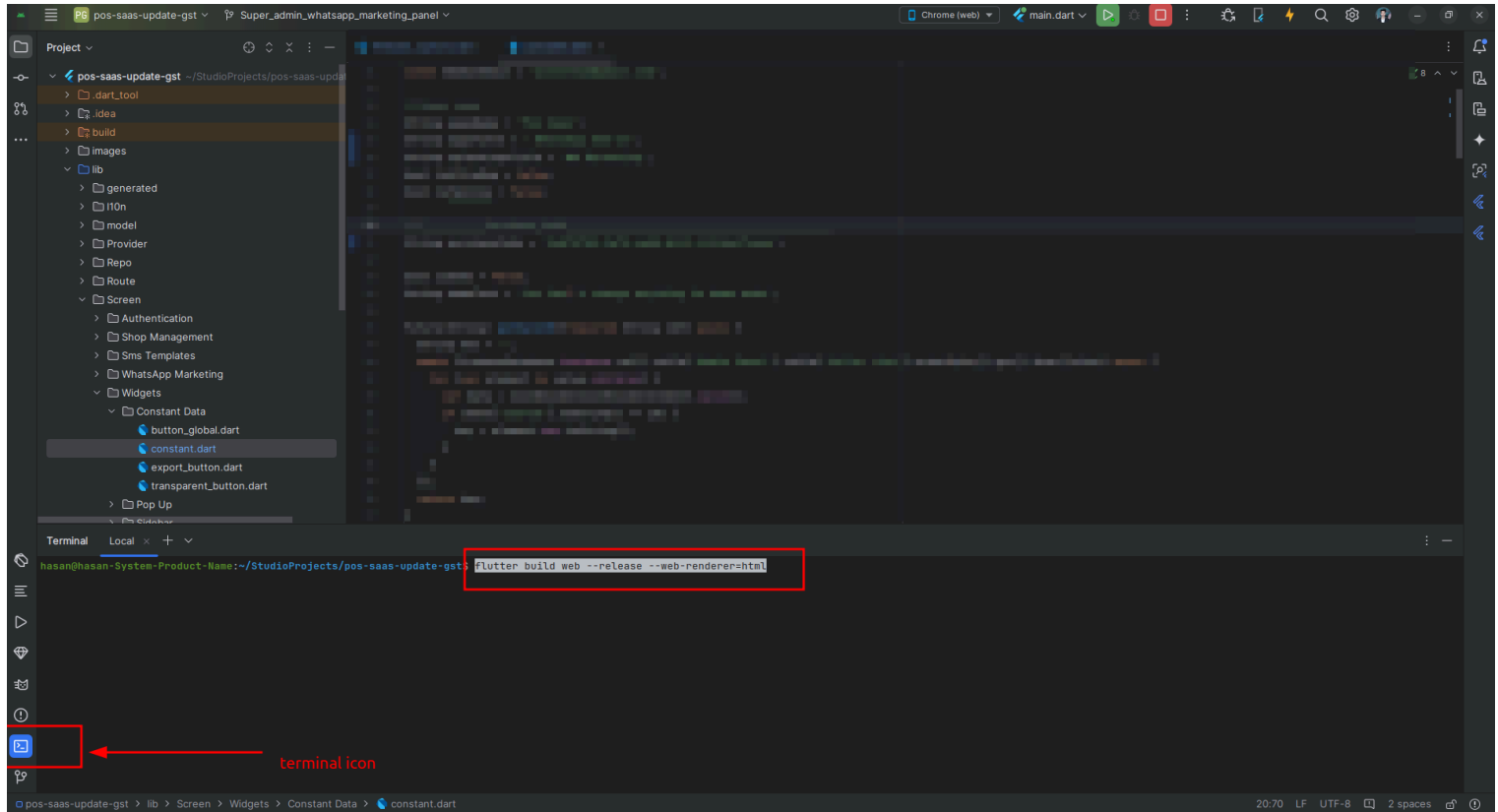
After writing your content, be sure to press the "Update" button to save your changes.

## Build File For Server Upload

To build the web application to upload the server, you have to open the terminal section. With android studio you can open it from the bottom bar of the environment. In terminal write the command

```
flutter build web --release --web-renderer=html
```

and press enter.

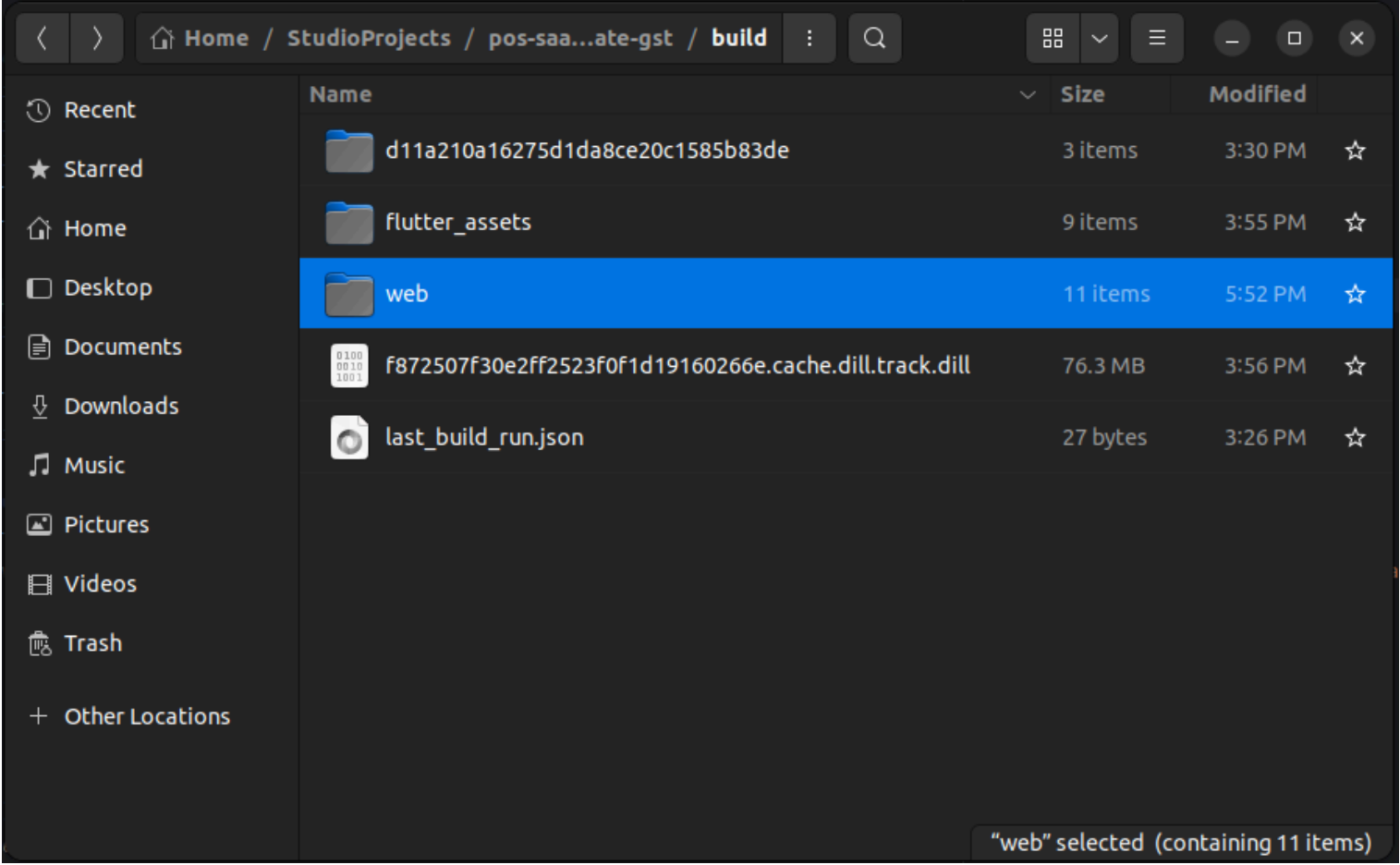


It will take some time to build the file. After build the app, you have to go to the directory where you store the file. And open the web folder which is inside the build folder.

```
Path: /pos-saas-update-gst/web
```

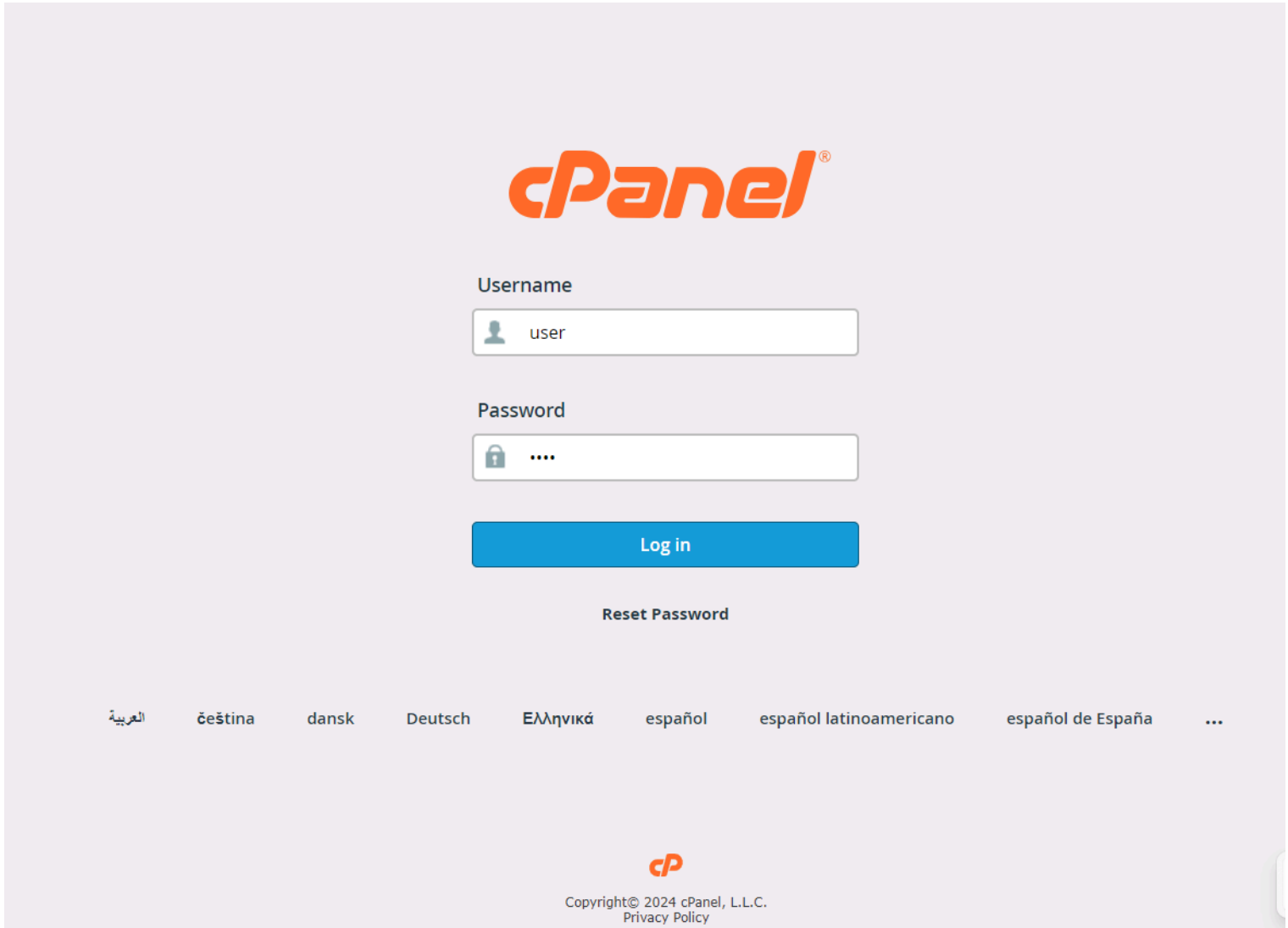
Now compress the web folder and extract it to the server

Make sure you select the web folder which is inside the build folder. Not outside the build folder.



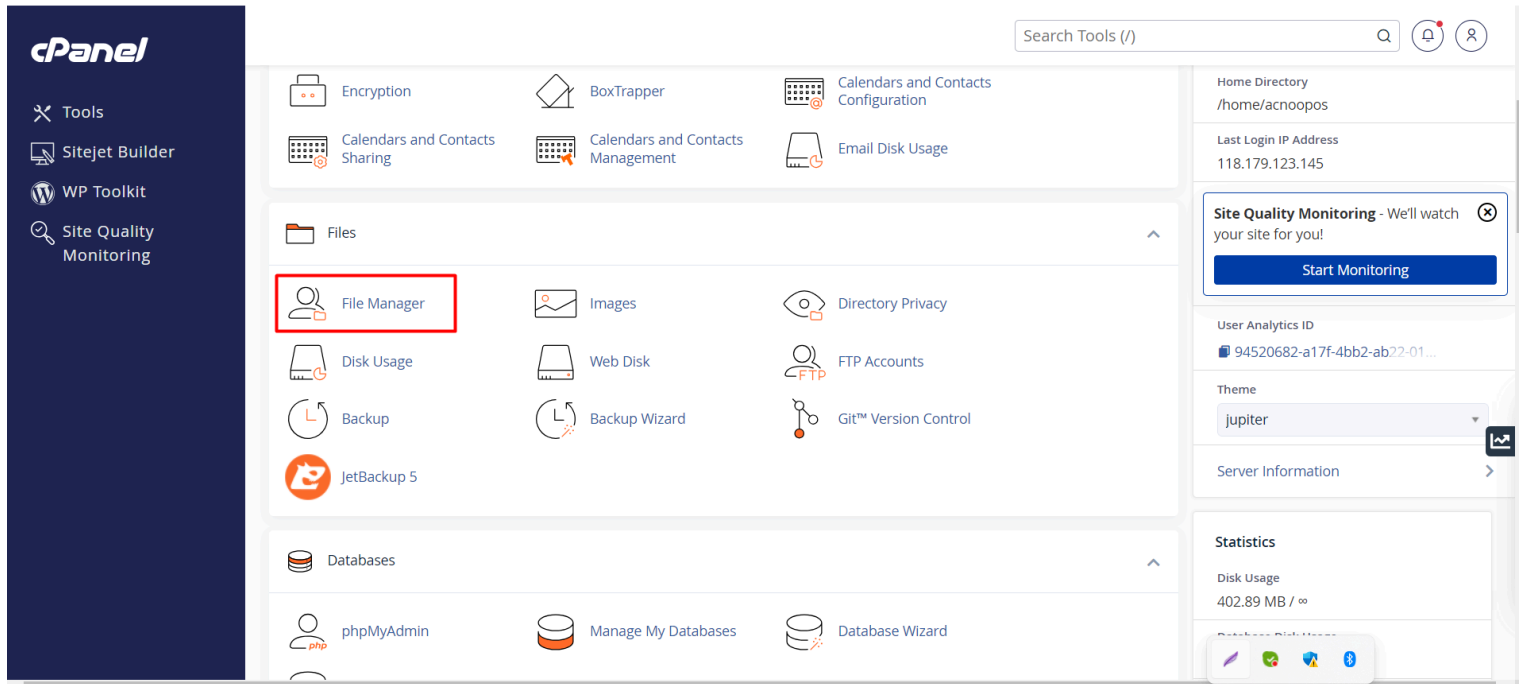
## Upload web file to server

Open the cPanel from your laptop/pc and login with your credentials.

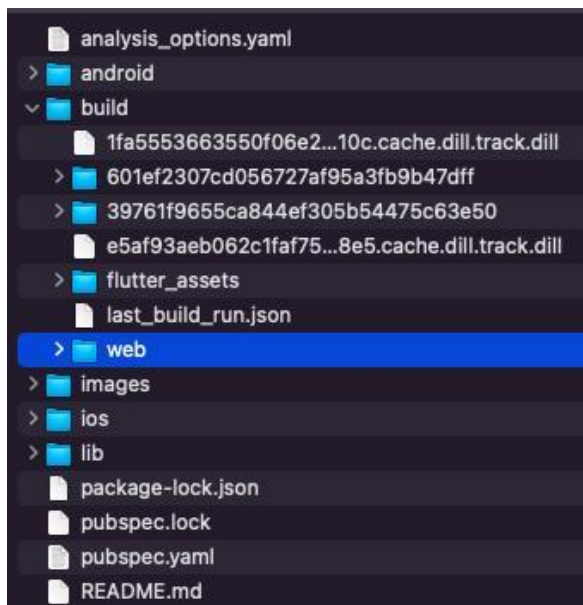


The image shows the cPanel login interface. At the top center is the cPanel logo in orange. Below it are two input fields: 'Username' with a user icon and the text 'user', and 'Password' with a lock icon and three dots. A blue 'Log in' button is positioned below the password field. Underneath the button is a 'Reset Password' link. At the bottom, there is a horizontal menu of language options: العربية, čeština, dansk, Deutsch, Ελληνικά, español, español latinoamericano, español de España, and an ellipsis. The footer contains the cPanel logo, the text 'Copyright© 2024 cPanel, L.L.C.', and a link to 'Privacy Policy'.

A new screen will appear with various menu options. From there, select and open "**File Manager.**"

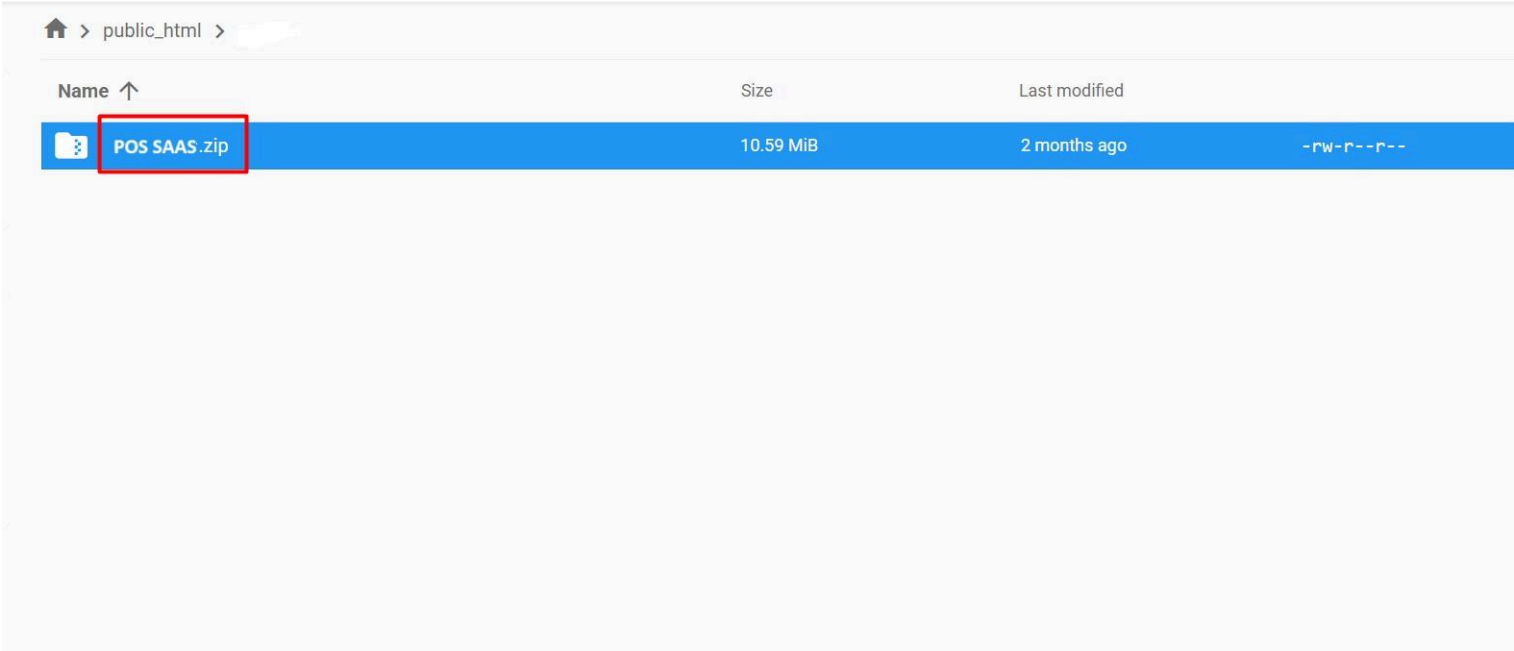


Now upload the server file that you built for your POS SaaS Whatsapp marketing app. You can find it in the build folder.

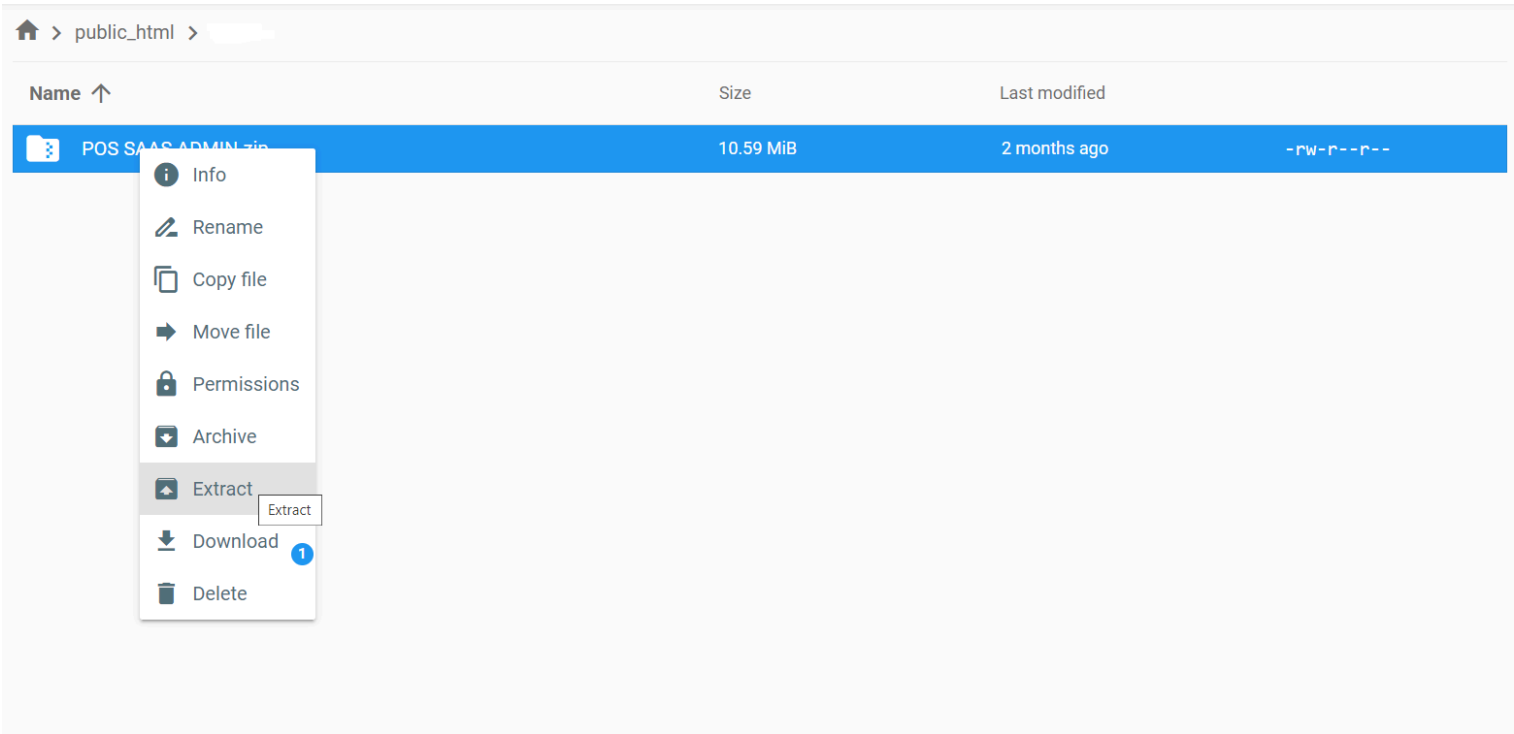


First, compress the web file you built, then proceed to upload it. You can move the file from the build folder to another location to make it easier to find. You can name the folder anything you like, but ensure it's easily identifiable for future reference.














After uploading the file, you will see it in your File Manager. Then, extract the file in this folder to make all the files visible.



To extract the file, place your **mouse cursor** over the **file you uploaded**, right-click on it, and select **"Extract"** from the menu to extract the contents.



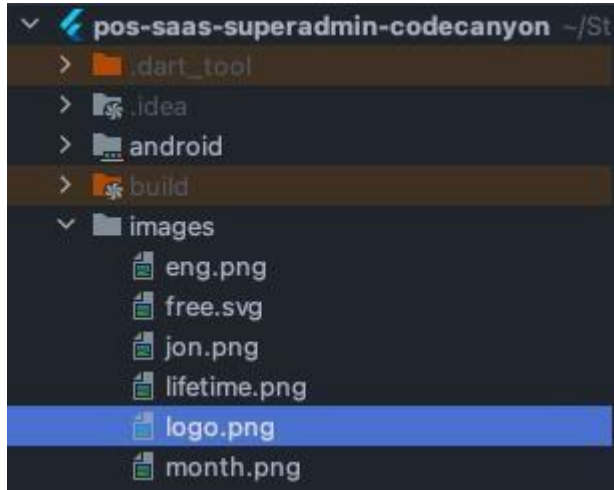
Now, you will be able to see the full contents of the files like this.

Name ↑	Size	Last modified	
 assets	—	a month ago	drwxr-xr-x
 canvaskit	—	a month ago	drwxr-xr-x
 icons	—	a month ago	drwxr-xr-x
 .htaccess	232 B	2 months ago	-rw-r--r--
 .last_build_id	32 B	a month ago	-rw-r--r--
 favicon.png	99.43 KiB	a month ago	-rw-r--r--
 flutter.js	7.6 KiB	a month ago	-rw-r--r--
 flutter_bootstrap.js	7.91 KiB	a month ago	-rw-r--r--
 flutter_service_worker.js	10.24 KiB	a month ago	-rw-r--r--
 index.html	1.95 KiB	a month ago	-rw-r--r--
 main.dart.js	3.51 MiB	a month ago	-rw-r--r--
 manifest.json	934 B	a month ago	-rw-r--r--
 version.json	108 B	a month ago	-rw-r--r--

## Change the App logo

To change the logo in the easy way, you have to make a logo for your app, and rename it to `logo.png`

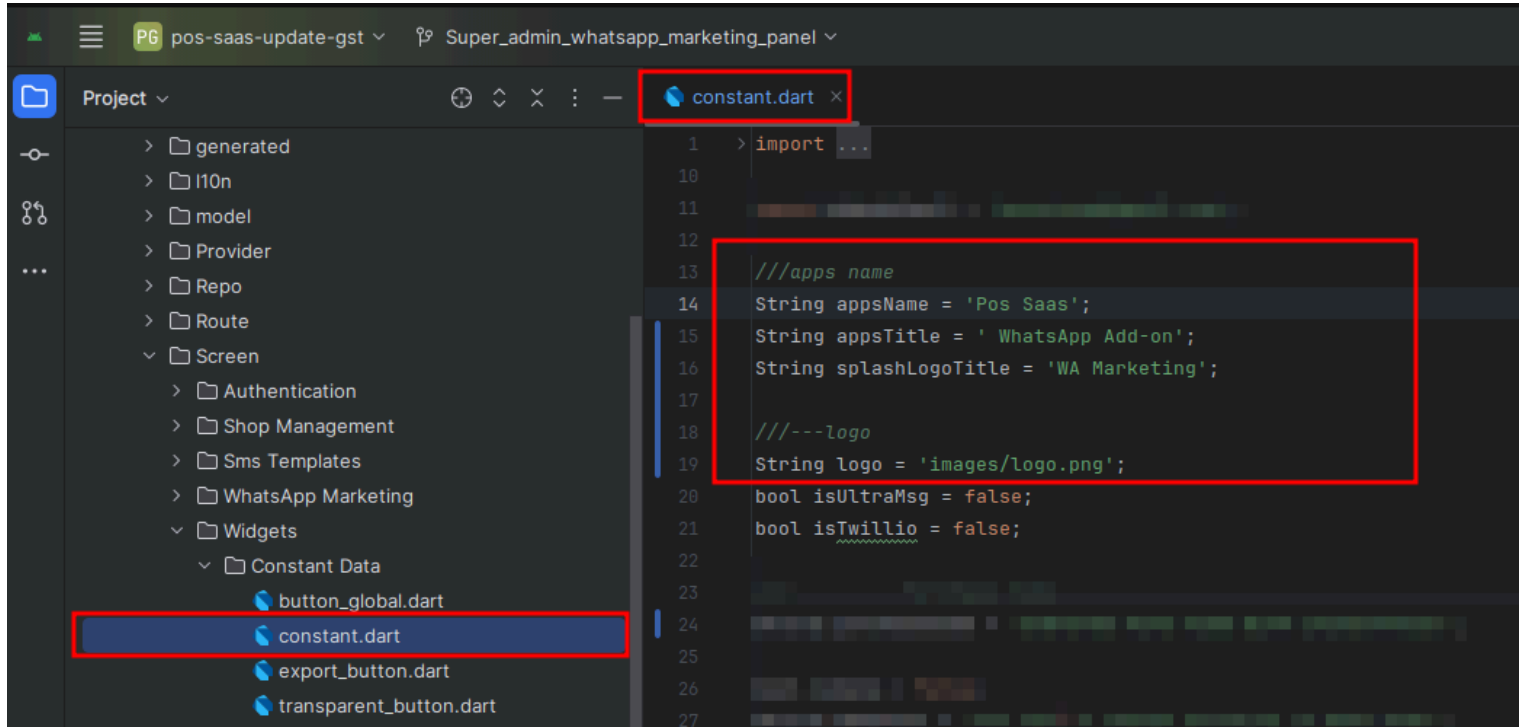
Now open the project, copy your logo from the local folder and paste it to the images directory in the project.



It will give you a warning to replace or override the image. You have to agree with that. And the image will replace the previous image.

## App name change

To change the app name which uses inside the app, you have to go to the constant.dart file and change the name as you want.  
appName/appsTitle/splashLogoTitle



```
1 > import ...
10
11
12
13 ///apps name
14 String appName = 'Pos Saas';
15 String appsTitle = ' WhatsApp Add-on';
16 String splashLogoTitle = 'WA Marketing';
17
18 ///---logo
19 String logo = 'images/logo.png';
20 bool isUltraMsg = false;
21 bool isTwillio = false;
22
23
24
25
26
27
```

After doing all the tasks successfully, without any error then build the file for server upload.

## Contact Information

For any inquiries, please reach out to us using the following contact details:

- **Email:** [acnooteam@gmail.com](mailto:acnooteam@gmail.com)
- **Skype:** [Team Live support](#)

We look forward to hearing from you!