

Multi Restaurants Flutter + PHP Laravel Admin Panel

Multi Restaurants App it's complete project using Flutter as mobile app for clients and PHP Laravel admin panel for restaurants manager and admin. you can reuse more than 60 widgets to customize your application.

NB: The same code used for both iOS and Android

PHP Laravel Admin Panel

Laravel is a web application framework with expressive, elegant syntax. We've already laid the foundation freeing you to create without sweating the small things.

The Laravel framework has a few system requirements. Of course, all of these requirements are satisfied by the Laravel Homestead virtual machine, so it's highly recommended that you use Homestead as your local Laravel development environment.


However, if you are not using Homestead, you will need to make sure your server meets the following requirements:

- PHP >= 7.2.0
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension
- Ctype PHP Extension
- JSON PHP Extension
- proc_open Extension

Installation

After you download (*and rate our app on Codecanyon* [Using this link](#)).

1 - You should find the folder named `/Multi Restaurants Laravel App` zip it and upload it inside your `/public_html` in your hosting (we use Bluehost hosting [Create Your Own](#)).

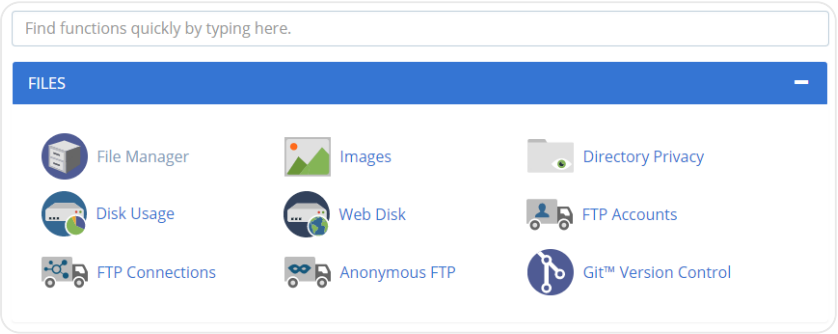


The Best Web Hosting
only \$3.95 per month

[Get Started](#)

Use our referral link
and get discount of

\$10

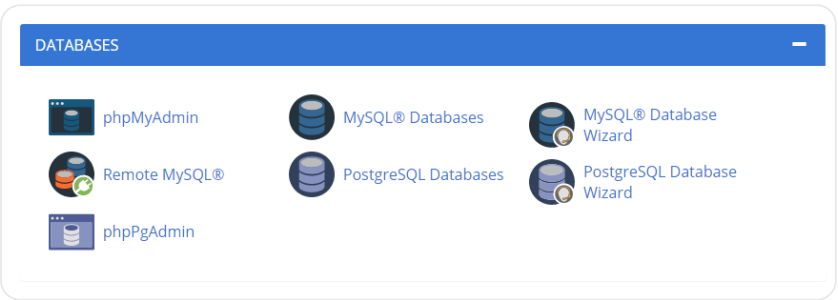


2 - Extract it you should get the following files in your file manager

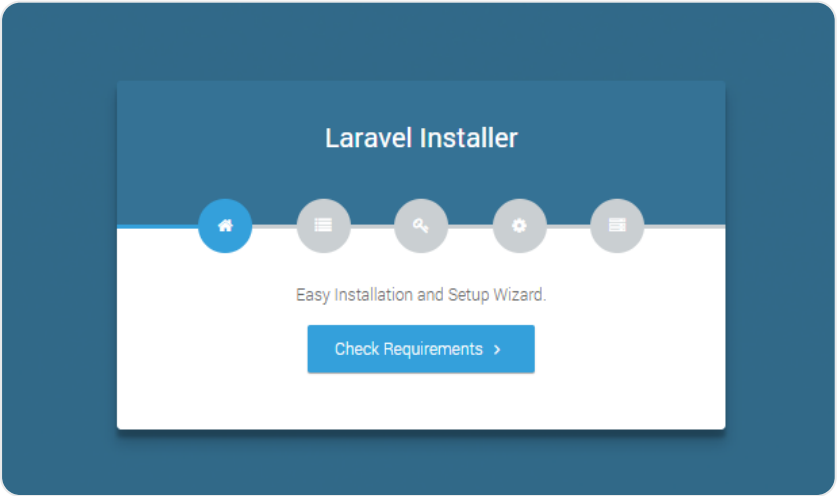
A screenshot of a file manager interface. At the top is a navigation bar with links: 'Home', 'Up One Level', 'Back', 'Forward', 'Reload', 'Select All', and 'Unselect All'. Below this is a table with two columns: 'Name' and 'Size'. The table lists various files and folders, including '.well-known', 'app', 'bootstrap', 'cgi-bin', 'config', 'database', 'public', 'resources', 'routes', 'storage', 'tests', 'vendor', '.env', '.env.example', '.gitattributes', and '.gitignore'. Each entry has a corresponding icon (folder or file) and a size value.

Name	Size
.well-known	4 KB
app	4 KB
bootstrap	4 KB
cgi-bin	4 KB
config	4 KB
database	4 KB
public	4 KB
resources	4 KB
routes	4 KB
storage	4 KB
tests	4 KB
vendor	4 KB
.env	739 bytes
.env.example	690 bytes
.gitattributes	116 bytes
.gitignore	185 bytes

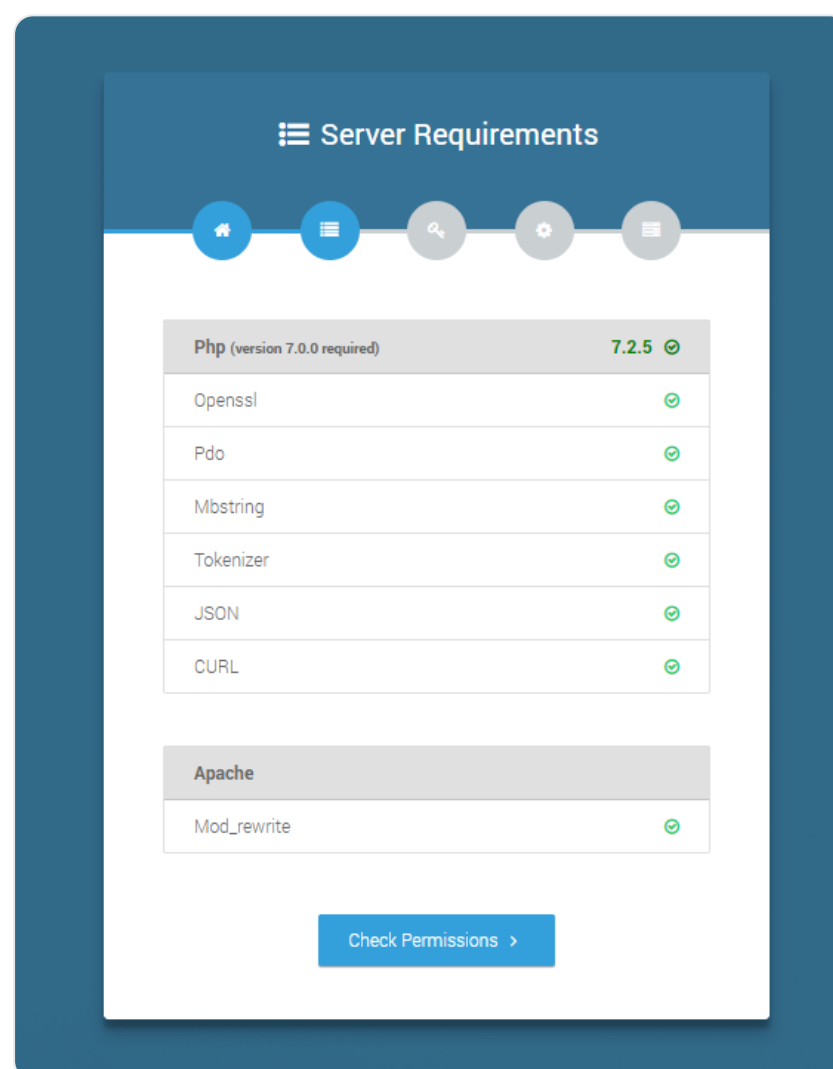
3 - Go to **MySQL Database** to Create new database for example **multi_restaurants** and new user and assign this database to the user created (remember the password used in this step we will use it in installation step).



4 - Open <http://yourdomain.com/public/install> .

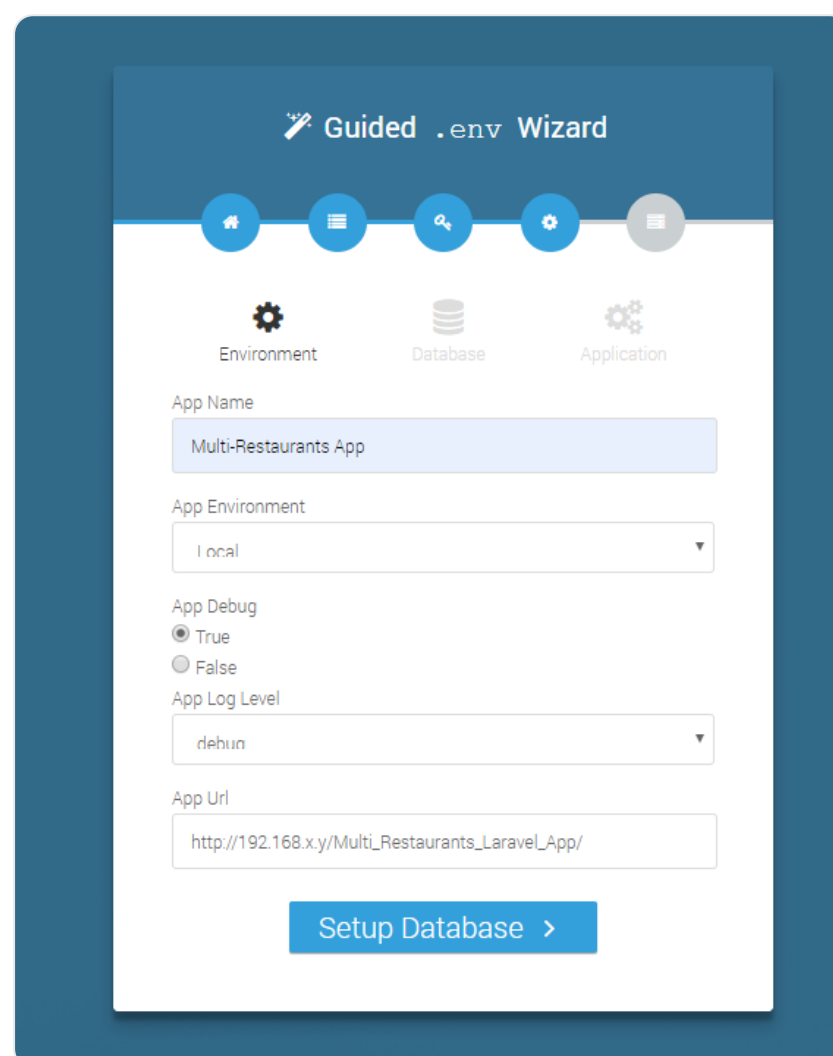


5 - Click check requirements after that:

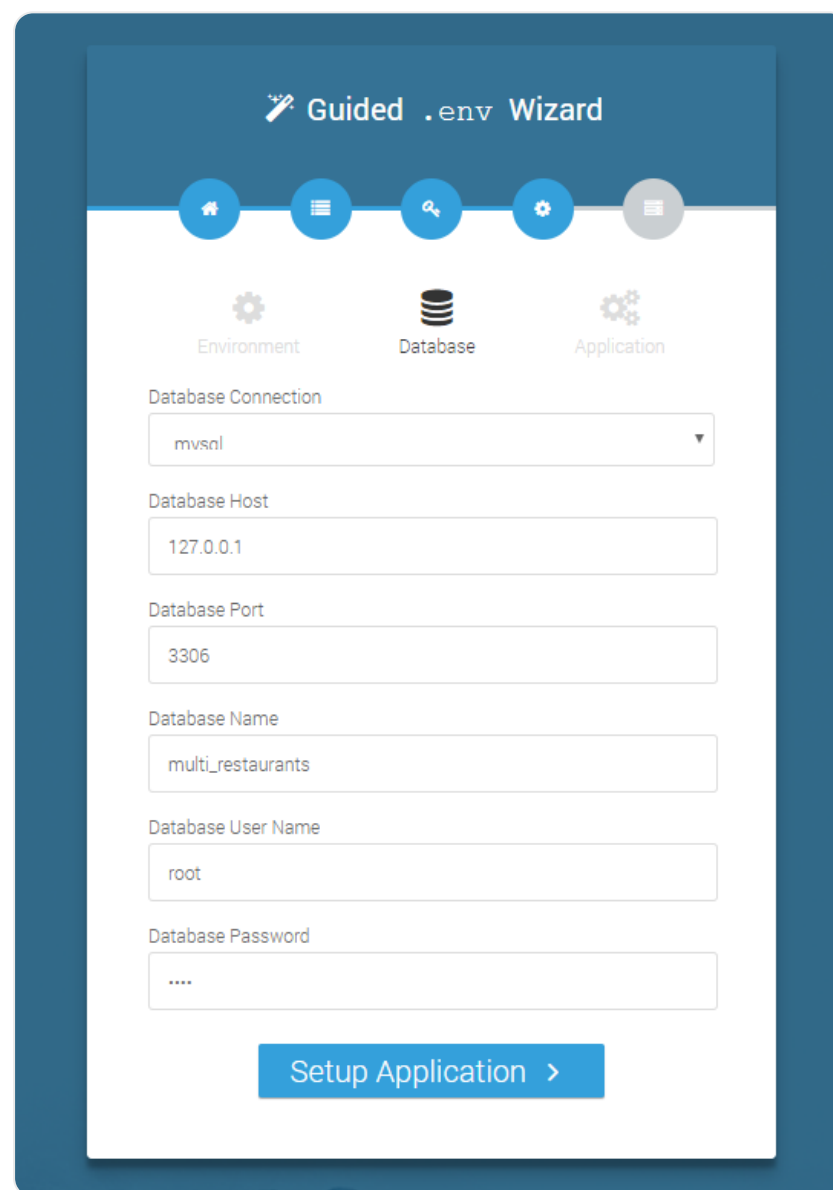


6 - Fill the form with your application details:

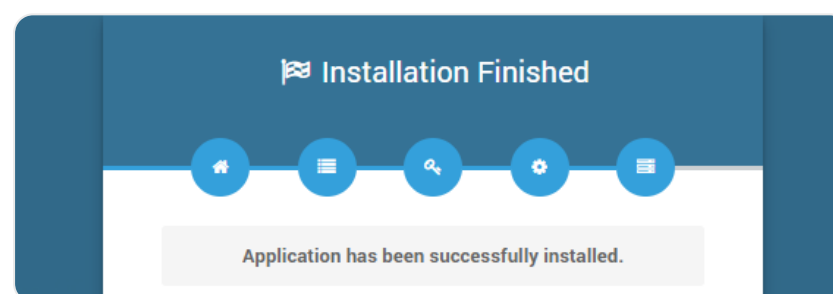
You can find your purchase code here [Using this link](#)



7 - Use same credentials created in [bluehost](#):



8 - You should get it if you have any other error please go to `public_html/storage` and delete the `installed` file and remove your database created in step 3 after that repeat the instructions be carefully.



Client & Driver Mobile App

Mobile app developed using flutter framework created by Google is an open-source mobile application development. It is used to develop applications for Android and iOS, as well as being the primary method of creating applications

Installation

- To edit this project you must have Flutter and Dart installed and configured successfully on your device [Install flutter](#)
- Set up your editor [Install the Flutter and Dart plugins](#)

Common

1- Download and unzip the `Multi Restaurants Flutter App` , use your preferred IDE (Android Studio / Visual Code / IntelliJ) to open the project.

2- Go to `/assets/img/logo.png` and replace it with your logo or app icon (its used in the splash screen not in the app icon)

Get Google Maps API key at

<https://cloud.google.com/maps-platform/>.

Android

1- Create your app icon `ic_launcher` and notification icon `ic_notification` folders [How to generate app icon?](#) after you generate icons folder replace the following folders:

- `/mipmap-hdpi` in `/android/app/src/main/res/` folder
- `/mipmap-mdpi` in `/android/app/src/main/res/` folder
- `/mipmap-xhdpi` in `/android/app/src/main/res/` folder
- `/mipmap-xxhdpi` in `/android/app/src/main/res/` folder
- `/mipmap-xxxhdpi` in `/android/app/src/main/res/` folder

NB: must rename app icon `ic_launcher` and notification icon `ic_notification`

2- And just run the following command

```
flutter pub get
```

If you want to change the package name following files and folders

3- Rename `food_delivery_app.iml` and `android/food_delivery_app_android.iml` (just replace `food_delivery_app` by you app name)

4- Open `/android/app/build.gradle` and change the package name

```
json

defaultConfig {
    applicationId "<REPLACE WITH YOUR
PACKAGE NAME>" // this is the package name
    minSdkVersion 19
    targetSdkVersion 28
    versionCode
flutterVersionCode.toInteger()
    versionName flutterVersionName
    testInstrumentationRunner
    "android.support.test.runner.AndroidJUnitRunner"

}
```

5- Open `/android/app/src/main/AndroidManifest.xml` ,
`/android/app/src/profil/AndroidManifest.xml` ,
`/android/app/src/debug/AndroidManifest.xml` and

specify your *YOUR PACKAGE NAME* , *YOUR APPLICATION NAME*, *YOUR GOOGLE MAPS KEY*

NB: You can get Google Maps API key at <https://cloud.google.com/maps-platform/>.

```
xml
<manifest
xmlns:android="http://schemas.android.com/apk/r

    package="<YOUR PACKAGE NAME>"

    <uses-permission
android:name="android.permission.INTERNET"/>
    <uses-permission
android:name="android.permission.ACCESS_FINE_LO

    <uses-permission
android:name="android.permission.ACCESS_COARSE_

    <!-- io.flutter.app.FlutterApplication is
an android.app.Application that
        calls
FlutterMain.startInitialization(this); in its
onCreate method.
        In most cases you can leave this as-
is, but you if you want to provide
        additional functionality it is fine
to subclass or reimplement
        FlutterApplication and put your
custom class here. -->
    <application
        android:name=".Application"
        android:label="<YOUR APPLICATION
NAME>"
        android:icon="@mipmap/ic_launcher">

        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="<YOUR GOOGLE
MAPS KEY>" />
```

6- Open `/android/app/src/main/kotlin/<Your Package name folders>/Application.kt` and change the package name same thing with

`/android/app/src/main/kotlin/<Your Package name folders>/MainActivity.kt`

7- In your `pubspec.yaml` in the top of this file, change the name with your app name and the project description.

8- In your project opened in your editor select root folder then right click and choose (Replace in Path) to replace all `package:food_delivery_app/` by your package name

9- Copy the generated `google-service.json` file in `/android/app/` folder, you can see this video tutorial

[How to configure push notifications](#)

iOS

Specify your API key in the application delegate

`ios/Runner/AppDelegate.m` :

```
objectivec

#include "AppDelegate.h"
#include "GeneratedPluginRegistrant.h"
#import "GoogleMaps/GoogleMaps.h"

@implementation AppDelegate

- (BOOL)application:(UIApplication
*)application
    didFinishLaunchingWithOptions:
(NSDictionary *)launchOptions {
    [GMServices provideAPIKey:@"YOUR KEY
HERE"];
    [GeneratedPluginRegistrant
registerWithRegistry:self];
    return [super application:application
didFinishLaunchingWithOptions:launchOptions];
}

@end
```

Or in your swift code, specify your API key in the application delegate `ios/Runner/AppDelegate.swift` :

```
swift

import UIKit
import Flutter
import GoogleMaps

@UIApplicationMain
@objc class AppDelegate: FlutterAppDelegate {
    override func application(
        _ application: UIApplication,
        didFinishLaunchingWithOptions
launchOptions: [UIApplicationLaunchOptionsKey:
Any]?
    ) -> Bool {
        GMSServices.provideAPIKey("YOUR KEY HERE")
        GeneratedPluginRegistrant.register(with:
self)
        return super.application(application,
didFinishLaunchingWithOptions: launchOptions)
    }
}
```

Opt-in to the embedded views preview by adding a boolean property to the app's `Info.plist` file with the key `io.flutter.embedded_views_preview` and the value `YES` .

- Click the Get dependencies or Packages get to install the libraries from `pubspecs.yaml` file.
- Open the simulator to run iOS or Android (as the step above)

- Then press the run button to start project (you can still open multi simulator at the same time)

Configuration

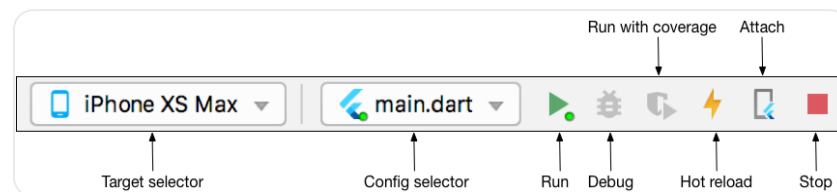
- Open `/assets/cfg/configurations.json` and edit the remote link example if you use local host as described in the admin panel

```
json
{
  "api_base_url":
  "http://yourdomain.com/public/api/",
  "base_url": "http://yourdomain.com/public/"
}
```

Run the app

In the target selector, select an Android device for running the app. If none are listed as available, select Tools> Android > AVD Manager and create one there. For details, see [Managing AVDs](#).

Click the run icon in the toolbar, or invoke the menu item Run > Run. Locate the main Android Studio toolbar:



Build and Install App

1- Generate Key by running the following command:

```
keytool -genkey -v -keystore key.jks -keyalg
RSA -keysize 2048 -validity 10000 -alias key
```

2- Open `/android/key.properties` and edit the following attributes:

```
storePassword=<Your Key Password>
keyPassword=<Your Key Password>
keyAlias=key
storeFile=<Your Absolute Path to key.jks
generated in step 1>
```

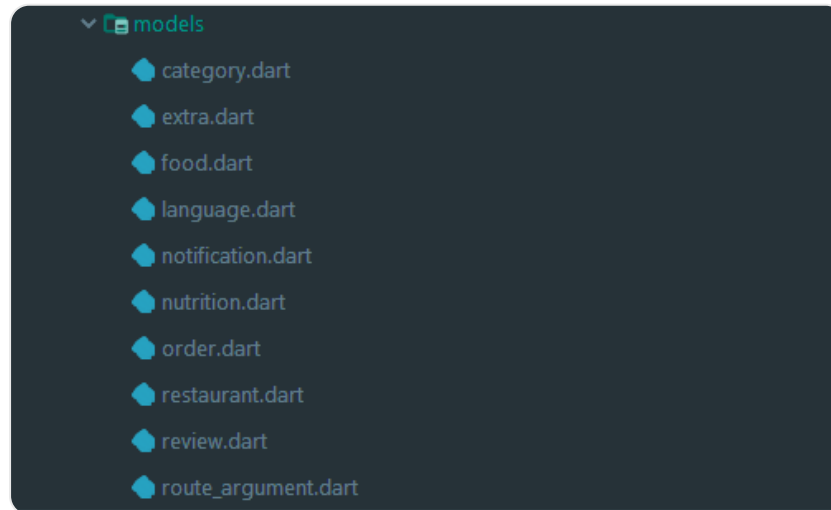
3- After you making all you changes and customizations save the project, Open the console, navigate to you project folder

- Run `flutter build apk --release` you should get the apk file in the `/output` folder
- Run `flutter install` to install you application on your connected devises

Files Structure

The flutter App `Multi_Restaurants_Flutter_App` use MVC Pattern.

- Models In this folder
`Multi_Restaurants_Flutter_App/lib/src/models` you should find all models and Entities used in the App just you can link them with your backend (**Firestore, PHP, JAVA, or other Api**) and you can you collection and list in your **App**

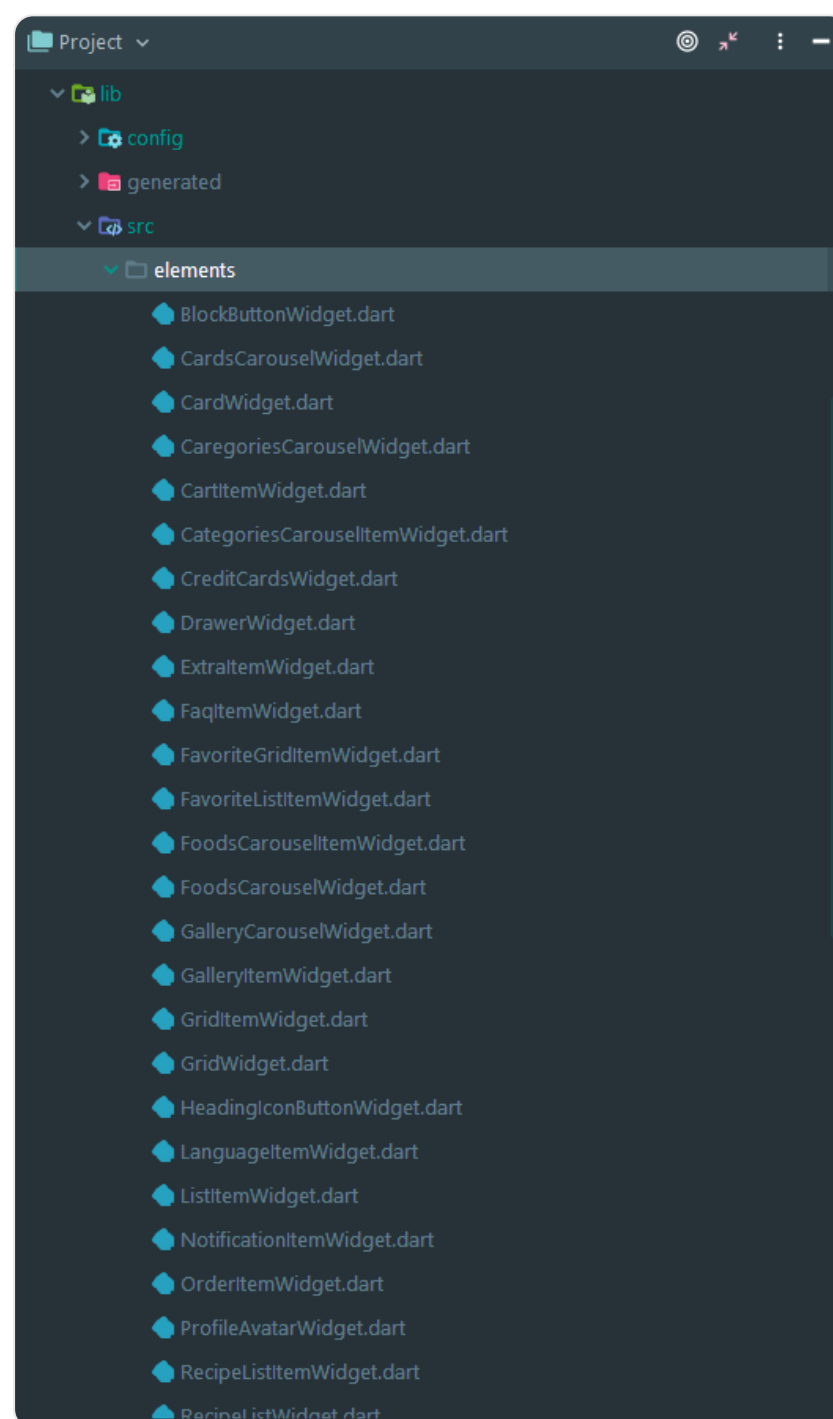


Example:

```
dart

//...
class Food {
  String id;
  String name;
  String restaurantName;
  double price;
  String image;
  String description;
  String ingredients;
  String weight;
  //...
}
```

- Controllers `Multi_Restaurants_Flutter_App/lib/src` inside this folder you should find controllers of the app
- Views In this folder
`Multi_Restaurants_Flutter_App/lib/src/elements` there are **+60 Widgets** ready to use in your **App**:



Just you can call the widget class and customize it with you parameters and you can get the stylish widget integrated in your app

Example:

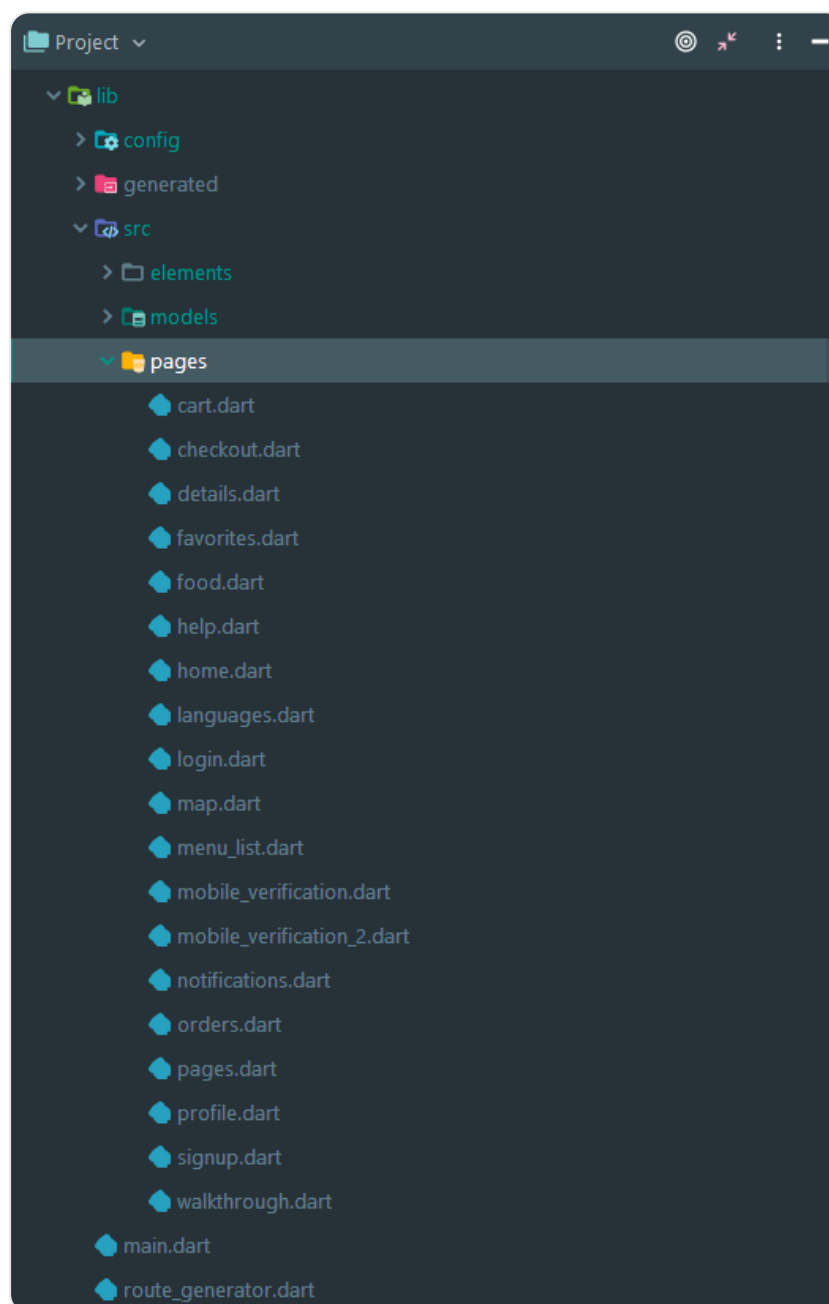
```
dart

//...
    new BlockButtonWidget(
      onPressed: () {

Navigator.of(context).pushNamed('/Pages',
arguments: 2);
      },
      color: Theme.of(context).accentColor,
      text: Text('Verify'.toUpperCase(),
        style:
Theme.of(context).textTheme.title.merge(TextSty
Theme.of(context).primaryColor))),
    ),
//...
```

In this folder

`Multi_Restaurants_Flutter_App/lib/src/pages` there are
+20 Pages/Screens pre-made:



Just you can edit or link your backend with this page to get awesome customized page

Update

NB: If you will install the project for the first time skip this step

Download new version from Codecanyon [Here](#) and extract the files in the folder `/Update`.

PHP Laravel Admin Panel

- Zip the content of this folder `Update/Multi Restaurants Laravel App`, upload it to your main folder in your hosting `/public_html` extract this zip file and overrate all files.

NB: You must find `<VERSION_CODE>` in [change log](#) in our item description in Codecanyon

- Open `http://yourdomain.com/public/update/<VERSION_CODE>`, follow the steps

Flutter Client Mobile Application

- Extract the files in the folder `/Update` and replace the folders and files.
- To update the dependencies open your console, go to your root folder of your project and run:

```
flutter pub get
```

NB: We aren't responsible if you loose your customization or your data, please do those steps be carefully

Getting Started with Admin Panel

Accounts information

After the installation you can login with 3 default roles

ADMIN

User or email: admin@demo.com | Password: 123456

RESTAURANT MANAGER

User or email: manager@demo.com | Password: 123456

CLIENT

User or email: client@demo.com | Password: 123456

Settings

Global settings

You can edit the global settings you should find the link in the side bar named [Settings > Global Settings](#)

The screenshot shows the 'Global Settings' page. On the left, a sidebar lists settings categories: Roles & Permissions, Global Settings (selected), Localisation, Social Authentication, Payment Settings, Mail, Translation, and Custom Fields. The main area is titled 'Global Settings' and contains the following fields:

- Application Name:** Multi-Restaurants (Text input)
- Short description:** Manage Mobile Application (Text input)
- Theme contract:** Light Theme (Dropdown menu)
- Theme color:** Red (Dropdown menu)
- Navbar Color:** Blue (Dropdown menu)
- Background Main Logo:** Blue (Dropdown menu)
- Fixed Header:** ☐ Fix Header on top
- Fixed Footer:** ☒ Fix Footer on bottom
- Custom Fields Models:** Users (Text input with a red 'x' icon)
- Blocked IPs:** (Empty text input)

Localisation Settings

Manage localisation setting date, timezone, default app language

The screenshot shows the 'Localisation' page. On the left, a sidebar lists settings categories: Localisation (selected), Global Settings, Roles & Permissions, Social Authentication, Payment Settings, Mail, Translation, and Custom Fields. The main area is titled 'Localisation' and contains the following fields:

- Date format:** IJSFY (H:is) (Text input)
- Timezone:** Africa/Algiers (Dropdown menu)
- Enable human date format:** ☒
- Application language:** English (Dropdown menu)
- Buttons:** Save Localisation (red), Cancel (grey)

Date Format: Enter the Date format that correspond to your Country

We formatted the date with 'l jS F Y (H*is*)s' so it gives

- j – Day of the month without leading zeros (1 to 31)
- M – A short textual representation of a month, three letters (Jan through Dec)
- Y – A full numeric representation of a year, 4 digits (Examples: 1999 or 2003)
- H*is*s – Hours, minutes and seconds
- A – AM / PM
- e – Time zone (default UTC)

Social Authentication

After you create an app for login with social media you can put all the information of the facebook app, google plus app or twitter app in this section [Settings > Social Authentication](#)

Social Authentication

f Authentication using facebook

Enable facebook

☒ Check it to use facebook as login method

Application ID

2088483311467392

Copy/Past your facebook application id

Application Secret

4fb7bff52d8eb13041f6fc59030e4b62

Copy/Past your facebook application secret key

▼ Authentication using twitter

Enable twitter

☐ Check it to use twitter as login method

Application ID

twitter

Copy/Past your twitter application id

Application Secret

twitter 1

Copy/Past your twitter application secret key

G Authentication using google

Enable google

☒ Check it to use google as login method

Application ID

527129559488-roolg8aq110p8rlq952lqa9tr

Copy/Past your google application id

Application Secret

Fpl8SLgc69ZWodk-xHaOrxn

Copy/Past your google application secret key

Save Social Authentication

Cancel

Payment Settings

You must create account on stripe payments gateway after that you can add the (Stripe Key and Stripe Secret)

Also you can edit the default tax percentage used in the app [Settings > Payments Setting](#)

Payment Settings

Stripe

Enable Stripe

☒ Check it to enable stripe payment method

Stripe Key

pk_test_pltZldgfh554TTVUL4O4I

Insert Stripe Key

Stripe Secret

sk_test_o9BVZlfgdx3RKDUytaokX4My3a20

Insert Stripe Secret

Default Tax

Default Tax

10

Insert Default Tax Ex: 20 for (20%)

Save Payment Settings

Cancel

Email Settings

[Settings > Emails](#)

SMTP Driver Active

Mailgun Driver

Sparkpost Driver

Mail Host

smtp.mailtrap.io

Insert the mail host address

Mail Port

2525

Insert the mail port

Mail encryption

TLS

Select the mail encryption TLS / SSL

Username

cb1bb56ad7b2e

Insert the mail username most of services use email as username

Mail Password

b005c361fdcf5b

Insert the mail password

Sender Email

3c3ed03f69-5d3964@inbox.mailtrap.io

Insert the sender email address

Sender Name

Name Sender

Insert the sender Name

Save SMTP Driver

Cancel

Translation

[Settings > Translation](#)

- To add new language to the admin panel, just duplicate [public_html/resources/lang/en](#) and

rename it with your local language code for example `ar` for arabic then go to `https://yourdomain.com/public/settings/translation/en` and choose you language and translate it

Deutsch

English

Español

French

Português

Sync Translations

Auth

Error

Installer_Messages

Lang

Pagination

Passwords

Validation

agree

I agree to the terms

Delete

already_member

I already have a membership

Delete

email

Email Address

Delete

failed

These credentials do not match our records.

Delete

forgot_password

I forgot my password

Delete

login

Login

Delete

login_facebook

Sign in using Facebook

Delete

login_google

Sign in using Google+

Delete

Custom Fields

You can add custom field and theirs types after that enable the custom field form `Settings > Global Settings`

Settings > Custom Fields

Custom Fields List

Create Custom Field

Export

Refresh

Print

Reset

Columns

Show 10 entries

Search

Name	Type	In Table	Order	Entity	Updated At	Action
phone	Text	No	2	Users	1 month ago	View Edit Delete
bio	Textarea	No	1	Users	1 month ago	View Edit Delete
address	Text	No	3	Users	1 month ago	View Edit Delete

Showing 1 to 3 of 3 entries

Previous

1

Next