

“Admin Web Panel ”

Table of Contents

1. [What is EZBus? \(#general\)](#)
2. [Prerequisites \(#prerequisites\)](#)
3. [Firebase \(#firebase\)](#)
4. [Payment Configuration \(#payment\)](#)
5. [Installation \(#installation\)](#)
6. [Activation & Configuration \(#configuration\)](#)
7. [FAQ \(#faq\)](#)

A) What is EZBus? - [top \(#toc\)](#)

EZBus is a transportation management system that is committed to enabling cities, individuals, and businesses to move anywhere. EzBus is tech-driven, fancy, and convenient.

EzBus consists of two Flutter apps for customers and drivers along with an Admin panel with a Laravel API system as a back-end and a Vue.js front-end system as a web interface.

Here we present the **Admin web panel sub system**

B) Prerequisites - [top \(#toc\)](#)

To ensure that the admin panel fully operational, some prerequisites need to be installed. In the following, I will list the required prerequisites and show how you can install them on a Ubuntu machine

1. Apache web server

```
sudo apt update  
sudo apt install apache2
```

2. Mysql server database engine

```
sudo apt install mysql-server
```

3. Set the password of the root for the mysql server database engine

```
#open a mysql terminal  
sudo mysql -u root #press enter when prompted for a password  
#write the following in mysql terminal  
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'YOUR PASSWORD HERE';  
#exit the mysql terminal  
exit;
```

4. Php 8.1

```
sudo apt install software-properties-common  
sudo add-apt-repository ppa:ondrej/php  
sudo apt update  
sudo apt install php8.1
```

5. Php 8.1 extensions

```
sudo apt install php8.1-common php8.1-mysql php8.1-xml php8.1-xmlrpc php8.1-curl php8.1-gd php8.1-imagick php8.1-cli php8.1-dev php8.  
sudo apt-get install php-curl php-xml php-mbstring -y
```

6. Enable mod_rewrite for Apache

```
sudo a2enmod rewrite  
sudo systemctl restart apache2
```

7. Configure apache for php8.1

```
sudo apt-get install libapache2-mod-php8.1  
sudo a2enmod php8.1  
sudo systemctl restart apache2
```

8. Composer

```

sudo apt-get update
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
sudo apt-get install curl php-cli php-mbstring git unzip
sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer

```

9. Please make sure that you have accounts for

- o [Google maps API \(<https://developers.google.com/maps/documentation/android-sdk/get-api-key>\)](https://developers.google.com/maps/documentation/android-sdk/get-api-key)

Please make sure that you enable the following APIs

- Geocoding API
- Maps JavaScript API
- Places API
- Directions API
- Maps SDK for Android
- Maps SDK for iOS

The screenshot shows the Google Cloud Platform API library interface. The 'Maps' category is selected under 'CATEGORY'. There are 15 results listed in a grid:

- Directions API**: Google. Directions between multiple locations.
- Distance Matrix API**: Google. Travel time and distance for multiple destinations.
- Geocoding API**: Google. Convert between addresses and geographic coordinates.
- Geolocation API**: Google. Location data from cell towers and WiFi nodes.
- Maps Elevation API**: Google. Elevation data for any point in the world.
- Maps Embed API**: Google. Make places easily discoverable with interactive Google Maps.
- Maps JavaScript API**: Google. Maps for your website.
- Maps SDK for Android**: Google. Maps for your native Android app.
- Maps SDK for iOS**: Google. Maps for your native iOS app.
- Maps Static API**: Google. Simple, embeddable map image with minimal code.
- Places API**: Google. Get detailed information about 100 million places.
- Roads API**: Google. Snap-to-road functionality to accurately trace GPS breadcrumbs.
- Street View Publish API**: Google. Publishes 360 photos to Google Maps, along with position, orientation, and connectivity.
- Street View Static API**: Google. Real-world imagery and panoramas.
- Time Zone API**: Google. Time zone data for anywhere in the world.

- o [Firebase account \(<https://firebase.google.com>\)](https://firebase.google.com)
- o **Braintree SDK**. You can test with a [sandbox \(<https://sandbox.braintreegateway.com>\)](https://sandbox.braintreegateway.com) account OR integrate a [live \(<https://www.braintreegateway.com>\)](https://www.braintreegateway.com) account.

C) Firebase - [top \(#toc\)](#)

1. Go to your account in <https://console.firebaseio.google.com> and create a new project. Name it "EZBus".

The screenshot shows the Firebase console homepage. The main heading is "Welcome to Firebase!" with the subtext "Tools from Google for building app infrastructure, improving app quality, and growing your business". Below this are two buttons: "Create a project" and "View docs".

A callout box at the bottom left provides information about Firebase projects:

- Firebase projects are containers for your apps
- Apps in a project share features like Real-time Database and Analytics
- [Learn more](#)

At the bottom, there are language selection ("Language English (United States)"), support links ("Support", "Terms", "Privacy Policy"), and a footer note: "© 2016 Google Inc. All rights reserved. Google and the Google logo are registered trademarks of Google Inc."

2. Create a new web app

Project Overview

Your project

Project name: ezbus-demo

Project ID: ezbus-demo

Project number: 544894325062

Default GCP resource location: Not yet selected

Web API Key: AlzaSyBzohnMmtSzxeVqzbhEcAvuXv9morS0

Environment

This setting customizes your project for different stages of the app lifecycle

Environment type: Unspecified

Your apps

There are no apps in your project

Select a platform to get started

Customize your nav!

You can now focus your console experience by customizing your navigation

[Learn more](#) [Got it](#)

3. Enter the new web app nickname, then click "Register app"

x Add Firebase to your web app

1 Register app

App nickname: ezbus-web

Also set up Firebase Hosting for this app. [Learn more](#)

Hosting can also be set up later. There is no cost to get started anytime.

[Register app](#)

2 Add Firebase SDK



4. Copy the firebase config object.

Please record them as we will use them later.

x Add Firebase to your web app

Register app

2 Add Firebase SDK

Use npm Use a <script> tag

If you're already using [npm](#) and a module bundler such as [webpack](#) or [Rollup](#), you can run the following command to install the latest SDK ([Learn more](#)):

`$ npm install firebase`

Then, initialize Firebase and begin using the SDKs for the products you'd like to use.

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
const firebaseConfig = {
  apiKey: "AIzaSyBzohnMmtSzxeVqzbhEcAvuXv9morS0",
  authDomain: "ezbus-[REDACTED].firebaseapp.com",
  projectId: "ezbus-[REDACTED]",
  storageBucket: "ezbus-[REDACTED].appspot.com",
  messagingSenderId: "54[REDACTED]5",
  appId: "1:54[REDACTED]:web:8[REDACTED]"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
```

Note: This option uses the [modular JavaScript SDK](#), which provides reduced SDK size.

Learn more about Firebase for web: [Get Started](#), [Web SDK API Reference](#), [Samples](#)

[Continue to console](#)



5. Go to settings of the web app, then open "Service accounts" tab. Click on "Generate new private key". Keep the downloaded file with the same name. The downloaded file name is in the form "ezbus-demo-firebase-adminsdk-jm2jc-c057abcdef.json".

Please record the file name as we will use it later.

The screenshot shows the 'Service accounts' tab in the Project settings. A modal window titled 'Generate new private key' is open, containing a warning message: '⚠ Your private key gives access to your project's Firebase services. Keep it confidential and never store it in a public repository.' Below the warning, there is a code snippet for generating a service account key in Node.js:

```
var admin = require("firebase-admin");
var serviceAccount = require("path/to/serviceAccountKey.json");

admin.initializeApp({
  credential: admin.credential.cert(serviceAccount)
});
```

[Generate key](#)

6. Go to "Authentication" section in the left pane, then click on "Get started"

The screenshot shows the 'Authentication' section in the Project Overview. On the left, there is a sidebar with 'Project shortcuts' and 'Authentication' selected. In the main area, there is a large 'Get started' button and a video thumbnail titled 'Introducing Firebase Authentication'. Below the video, there is a 'Learn more' section with three items:

- How do I get started? View the docs
- How does Authentication work? View the docs
- What can Authentication do for me? View the docs

7. Select Email provider

The screenshot shows the 'Sign-in method' tab selected in the Firebase Authentication console. Under the 'Native providers' section, the 'Email/Password' provider is highlighted with a red box. Other providers listed include Google, Facebook, Play Games, OpenID Connect, Phone, Game Center, Apple, GitHub, Microsoft, Twitter, and Yahoo. Below the provider list, there are sections for 'Anonymous' and 'Disabled' users.

8. Enable Email/Password

The screenshot shows the configuration for the 'Email/Password' sign-in provider. The provider is enabled, indicated by a checked checkbox. A note below explains that it allows users to sign up using their email address and password, with SDKs providing email address verification, password recovery, and email address change primitives. There is also an option for 'Email link (passwordless sign-in)' which is currently disabled. The 'Save' button is visible at the bottom right.

9. Add the super admin account. Go to "Users" section tab, then click on "Add user"

The screenshot shows the 'Users' tab selected in the Firebase Authentication console. The table header includes columns for Identifier, Providers, Created, Signed In, and User UID. A prominent blue 'Add user' button is located at the top right of the table area. Below the table, a message states 'No users for this project yet'.

10. Enter the email of the super admin and a password, then click on "Add user"

The screenshot shows the Firebase console's Authentication interface. On the left, there's a sidebar with project settings like 'Project Overview', 'Build', 'Release & Monitor', 'Analytics', 'Engage', and 'All products'. The 'Authentication' tab is selected. In the main area, under the 'Users' tab, a modal window is open for adding a new user. It has fields for 'Email' (set to 'admin@yourdomain.com') and 'Password' (set to 'admin123456'). Below the modal, a message says 'No users for this project yet'. At the top of the main area, there are tabs for 'Users', 'Sign-in method', 'Templates', 'Usage', 'Settings', and 'Extensions'.

11. Get the UID of the super admin. Go to "Users" section tab, then copy the UID.

Please record email, password, and UID as we will use them later.

This screenshot shows the same Firebase Authentication interface after the user has been added. The 'Users' table now contains one row: 'admin@yourdomain.com' with a User UID of 'I8nItaBjyQVLifV0uktyUuSiF53'. A 'Copy UID' button is visible next to the User UID column. The rest of the interface is identical to the previous screenshot.

D) Payment Configuration - [top \(#toc\)](#)

The system has four types of payments. They are as follows:

1. Braintree
2. RazorPay
3. FlutterWave
4. PayTabs
5. No payment (to allow free rides). **If you need this option, skip configuring any payment gateway in the following.**

In the following, we show how to configure each of them

1) Braintree

You need to get "Public Key", "Private Key", and "Merchant ID" from Braintree dashboard.

Please record them as we will use them later.

1. Go to your account in <https://www.braintreegateway.com> or <https://sandbox.braintreegateway.com>
2. Go to "Settings" from the upper right menu, then click on "API"

3. Click on "Generate New API Key"

4. Click on view for the generated key to get the "Public Key", "Private Key", and "Merchant ID".

Please record them as we will use them later.

2) RazorPay

You need to get both "Key Id" and "Key Secret" from RazorPay dashboard.

Please record them as we will use them later.

1. Open your "RazorPay" account dashboard

2. Choose "Settings" section in the left pane, then click on "API Keys"

3. Get both "Key Id" and "Key Secret" and save them.

3) FlutterWave

You need to get both "Public Key" and "Secret Key" from FlutterWave dashboard.

Please record them as we will use them later.

1. Open your "Flutterwave" account dashboard
2. Choose "Settings" section in the left pane, then click on "API"

The screenshot shows the FlutterWave API Keys page. On the left sidebar, under 'YOUR BUSINESS', there are links for Transactions, Customers, Balance, Transfers, Subaccounts, Cards, and Airtime & DSTV. Under 'COLLECT PAYMENTS', there are links for Store and Payment Links. Under 'BILLING TOOLS', there are links for Payment Plans, Invoices, and Events. At the bottom of the sidebar are buttons for 'Switch to Live Mode', 'Referrals', and 'Settings'. The main content area is titled 'Test API Keys' and contains three input fields: 'Public Key' (with a redacted value), 'Secret Key' (with a redacted value), and 'Encryption Key' (with a redacted value). Each field has an orange 'Copy key' button to its right.

3. Get both "Public Key" and "Secret Key" and save them.

4) PayTabs

You need to get both "Profile ID", "Server Key", "Currency", and "Region" from PayTabs dashboard.

Please record them as we will use them later.

1. Open your "Paytabs" account dashboard
2. Choose "Developers" section in the left pane, then click on "API"

The screenshot shows the PayTabs Developers API Keys page. The left sidebar includes links for Home, Profile, Team, Transactions, Developers (which is highlighted in blue), Invoices, PayLinks, Repeat Billing, and Reports. The main content area shows a table titled 'INTEGRATION KEYS (PROFILE ID 89661)'. The table has columns for Title, Server Key, and Client Key. It lists two entries: 'Test API Key' with Server Key SHJN266GRB-JDBKL2ZDHD-9GNL9LZM9R and Client Key CHKMQ7-92676D-2KBM9D-VDN77Q, and 'Test SDK Key' with Server Key S2JN266G92-JDBKL2ZDLL-LNGHJKBLTD and Client Key CRKMQ7-92R66D-2KBM9D-HB2MBM. There is a '+' button at the top right of the table.

Title	Server Key	Client Key
Test API Key	SHJN266GRB-JDBKL2ZDHD-9GNL9LZM9R	CHKMQ7-92676D-2KBM9D-VDN77Q
Test SDK Key	S2JN266G92-JDBKL2ZDLL-LNGHJKBLTD	CRKMQ7-92R66D-2KBM9D-HB2MBM

3. Get the "Server Key" and save it

4. Obtain your Profile Id and save it (the number before your name)

The screenshot shows the PayTabs admin dashboard. On the left, there's a sidebar with links: Home, Profile, Team, Transactions, Developers, Invoices, PayLinks, Repeat Billing, and Reports. The main area has a header with 'Test' and '89661: CREATIVEAPPS'. It displays a message: 'In order to complete your merchant application, please go to the [Onboarding](#) section and provide the required details.' Below this is a 'Verify your phone number' section with a 'Send verification code' button. Three summary boxes are shown: '18 TRANSACTIONS TODAY' (with a 'VIEW DETAILS' button), '960.00 EGP SALES TODAY' (with a 'VIEW DETAILS' button), and '960.00 EGP SALES THIS MONTH' (with a 'VIEW DETAILS' button). A 'LATEST TRANSACTIONS' table follows, with columns: Ref, Customer Name, Type, Payment Method, Currency, Amount, Date, Time, and Status. One row is visible: 'TST2204301054419 qqq q Sale Visa EGP 80.00 12/Feb/2022 17:44'. At the bottom right of the dashboard is a green circular icon.

5. Choose one of the following regions: 'ARE','EGY','SAU','OMN','JOR','GLOBAL'
6. Record the currency you registered in with PayTabs account. It is one from the list ['AED','EGP','SAR','OMR','JOD','US'].

E) Installation - [top \(#toc\)](#)

As we indicated, the admin panel composed of a Laravel API system as a back-end and a Vue.js front-end system as a web interface. We indicate the installation steps for each of them separately.

1) Laravel API system installation

I assume that the purchased admin web panel is located in the folder "ezbus" at "/home/ezbus". You can modify this path if you download the folder in a different path. Please follow the following steps for installation

1. Go inside the folder

```
cd /home/ezbus
```

2. Ensure that php 8.1 is the current version of your php by typing the command

```
sudo update-alternatives --config php
```

3. Install dependencies

```
composer install
```

4. Make a link to storage folder in the public folder

```
ln -s ../../storage/app/public public/backend/storage
```

5. Put the downloaded file in step (C)-5 above in the "ezbus" folder

6. Create the .env file from the provided example .env

```
cp .env.example .env
```

7. Edit the .env file and set the following

```
APP_URL = your web site url, for example www.mydomain.com
DB_USERNAME = root
DB_PASSWORD = password of root user of mysql database
FIREBASE_CREDENTIALS=Write the downloaded name in step (C)-5 and pre-append it with ./ (for example, ./ezbus-demo-firebase-ac
```

If you want to use Braintree, then set the following

```
BRAINTREE_ENV=sandbox or live
BRAINTREE_MERCHANT_ID= your Braintree merchant ID from step (D)-1
BRAINTREE_PUBLIC_KEY= your Braintree public key from step (D)-1
BRAINTREE_PRIVATE_KEY= your Braintree private key from step (D)-1
```

If you want to use RazorPay, then set the following

```
RAZORPAY_KEY= your RazorPay key id from step (D)-2
RAZORPAY_SECRET= your RazorPay key secret from step (D)-2
```

If you want to use FlutterWave, then set the following

```
FLW_PUBLIC_KEY= your FlutterWave public key from step (D)-3
FLW_SECRET_KEY= your FlutterWave secret key from step (D)-3
```

If you want to use PayTabs, then set the following

```
paytabs_profile_id= your PayTabs profile id from step (D)-4
paytabs_server_key= your PayTabs server key from step (D)-4
paytabs_currency= your PayTabs currency from step (D)-4
paytabs_region= your PayTabs region from step (D)-4
```

If you do not want to use any payment gateway, then do not set any of the above payment gateway keys.

8. Generate a new key

```

php artisan key:generate

9. Give full permission to storage and bootstrap/cache folders
    chmod -R 777 storage bootstrap/cache

10. Change owner of the public and avatar folders
    sudo chown -R $USER:$USER public
    sudo chown -R $USER:$USER storage/app/public/avatars/

11. Configure an Apache virtual host for EZBus
    o Create a configuration file for the virtual host for EZBus
        sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/EZBus.conf

    o Edit the configuration file "/etc/apache2/sites-available/EZBus.conf" and replace its contents with the following

        <VirtualHost *:80>
            ServerName EZBus
            DocumentRoot /home/ezbus/public
            <Directory /home/ezbus/public>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride All
                Require all granted
            </Directory>
            ErrorLog ${APACHE_LOG_DIR}/error.log
            CustomLog ${APACHE_LOG_DIR}/access.log combined
        </VirtualHost>

    o Enable the configuration file "EZBus.conf"
        sudo a2dissite 000-default.conf
        sudo a2ensite EZBus.conf
        sudo systemctl restart apache2

12. Open mysql terminal and create a database with name "ezbus_db"
    mysql -u root -p #enter the root password when prompted
    create database ezbust_db; #then exit the mysql terminal

13. Make sure that you give full permission to storage and bootstrap/cache folders.
    Database migration will fail if you did not do this
    chmod -R 777 storage bootstrap/cache

14. Make sure that you have changed the owner of the public and avatar folders.
    Database migration will fail if you did not do this
    sudo chown -R $USER:$USER public
    sudo chown -R $USER:$USER storage/app/public/avatars/

15. Update the super admin credentials in the users seeder file in "database/seeders/UsersTableSeeder.php". Write the email, password, and UID of the super admin, that was obtained in step (C)-11, in this seeder file.
    Database migration will fail if you did not do this

    $user = User::create([
        'name' => 'SuperAdmin',
        'email' => 'admin@yourdomain.com',
        'password' => Hash::make('admin123456'),
        'role' => 0,
        'status_id' => 1,
        'uid' => "l8nIta8bj0VLifjV0uktyUu5iF53"
    ]);

16. Migrate the database
    php artisan migrate:fresh --seed

17. Install redis server
    sudo apt-get install redis-server
    sudo systemctl enable redis-server.service
    sudo apt-get install php-redis

18. Install npm
    sudo apt-get install nodejs
    sudo apt-get install npm

```

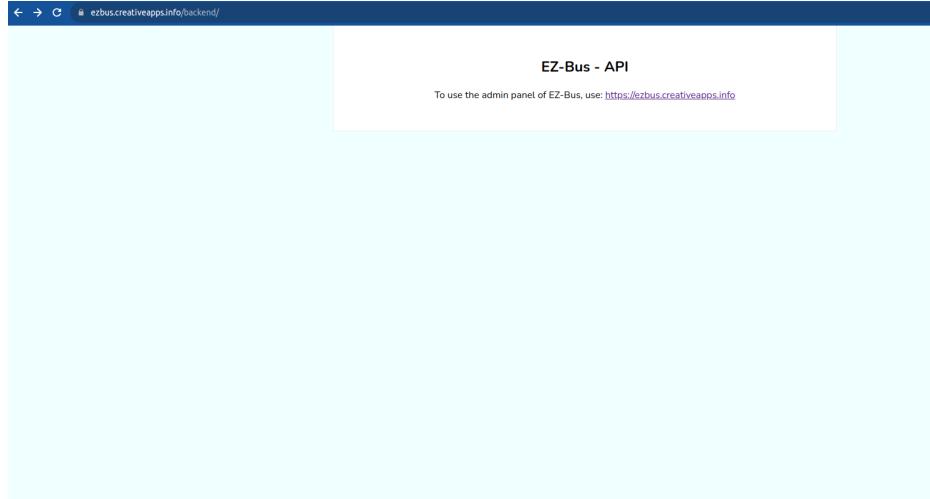
19. Install Laravel echo server

```
sudo npm install -g laravel-echo-server  
sudo laravel-echo-server init #I recommend to press enter to choose the default answer for every question (except if you have  
sudo nohup laravel-echo-server start & #start the laravel-echo-server in background and make it permanently alive
```

20. Update permissions of privacy and terms pages.

```
chmod 777 public/privacy.html  
chmod 777 public/privacy_local.html  
chmod 777 public/terms.html  
chmod 777 public/terms_local.html
```

Now, the Laravel API system is ready to be used. You can verify that by opening the url "your domain and append it with /backend (for example www.mydomain.com/backend) in a browser. You should see the following image



2) Vue.js Front End

In the following, we show how to install the Vue.js front end that will be used as a web interface for the admin panel

1. Go inside the folder

```
cd /home/ezbus
```

2. Go inside the folder "frontend"

```
cd frontend
```

3. Create the .env.production file from the provided example .env.production.example

```
cp .env.production.example .env.production
```

4. Edit the file ".env.production" and set the following

```
# API Config  
VUE_APP_API_URL=your web site url, for example www.mydomain.com and append it with /backend. For example, www.mydomain.com/ba  
  
# Firebase Config  
VUE_APP_API_KEY=AIzaSy..... (from step (C)-4)  
VUE_APP_AUTH_DOMAIN=ezbus.... (from step (C)-4)  
VUE_APP_PROJECT_ID=ezbus.... (from step (C)-4)  
VUE_APP_STORAGE_BUCKET=ezbus.... (from step (C)-4)  
VUE_APP_MESSAGING_SENDER_ID=544..... (from step (C)-4)  
VUE_APP_APP_ID=1:544.....:web:01b..... (from step (C)-4)  
  
# Google Maps Config  
VUE_APP_GOOGLE_MAPS_API_KEY=Your Google Maps API key.  
  
# System Name  
VUE_APP_SYSTEM_NAME= Your System Name (instead of EZ-Bus)
```

5. Edit the file "front-end/public/index.html" and set the title to your system name (instead of EZ-Bus)

```
<title>Your System Name</title>
```

6. Install dependencies

```
npm install
```

7. Build the project

```
npm run build
```

8. Copy the generated files to the public folder of the Laravel API system

```
cp -r dist/* ../public/
```

Now, the Vue.js front end is ready to be used. You can verify that by opening the url "your domain (for example www.mydomain.com) in a browser. You should see the login page of the admin panel

F) Activation & Configuration - [top \(#toc\)](#)

Once you installed EZBus as indicated above, login with the super admin account credentials to set the different keys used by EZBus

1. Go to this [link](https://auth.creativeapps.info) (<https://auth.creativeapps.info>) and enter your purchase key. You can get the purchase key from your CodeCanyon download section. More information is [here](#) (<https://help.market.envato.com/hc/en-us/articles/202822600-Where-Is-My-Purchase-Code->). domain (or localhost). You will get two secure keys. One for your web site url and the other for localhost. **Copy your secure keys and keep it in a safe place as you will not be able to generate them again.**

2. Open your web site and login with the super admin credentials.

3. You will see "Activate your account" button. Click on it and you will be redirected to the activation page

4. Enter your secure key and click "Activate"

5. **Add scheduling entry for cron. This step is very important to handle all automated tasks**

- First open your terminal and run

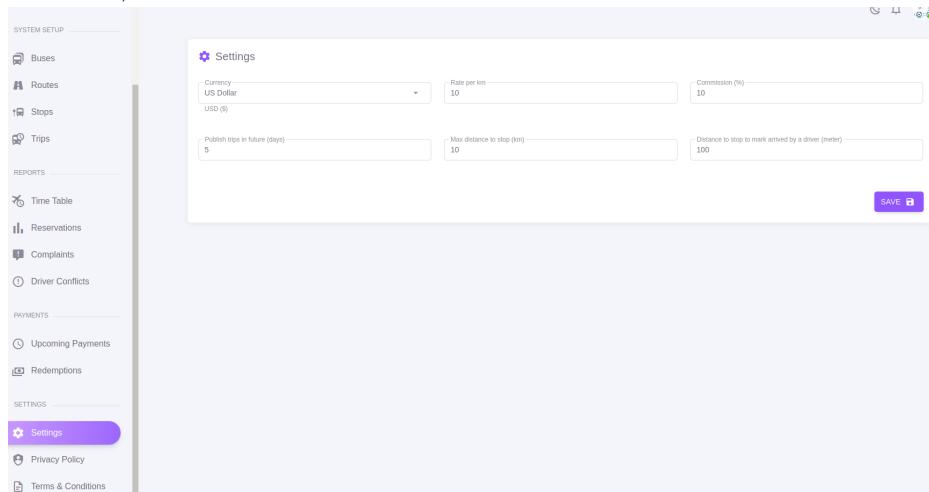
```
crontab -e
```

- Then add the following line

```
* * * * * cd /home/ezbus && php artisan schedule:run >> /dev/null 2>&1
```

6. Adjust the settings of the admin panel. Specifically, you need to set the following

- Currency**
- Rate per km:** this is the rate that will be used to calculate the trip price
- Commission percentage:** this is the percentage that will be deducted from the trip price and will be added to the admin account
- No. of days in future to publish trips:** this is the number of days that will be used to publish trips in the apps. No trips will be shown in the apps if the trip date is more than this number
- Max distance to stop:** this is the maximum distance that will be used to calculate the distance between the driver and the stop. If the distance is more than this number, the driver will not be able to start the trip
- Distance to stop to mark arrived by a driver:** this is the distance that will be used to calculate the distance between the driver and the stop. If the distance is less than this number, the driver will be able to mark arrived



G) FAQ - [top \(#toc\)](#)

1. I do not see any trips in the apps even I created some trips in the admin panel

Make sure that you write the crontab command correctly. Please see step (F)-5 above

2. When I change some details in a trip, it is not reflected in the apps

Note that the apps are working on a copied version from the trips table. This means that any change in the trips table will not be reflected immediately in the apps, but will be reflected in future trips.

3. Laravel-echo-server not working

Configure your firewall and make sure that you allow traffic through the port 6001

4. How do I know that the laravel-echo-server is working correctly?

Go to the url of the admin panel and append it with 6001, i.e., open `http://YOUR_WEB SERVER_URL:6001` and check if it gives OK

5. The home page of the admin panel gives a blank white page

- Make sure that you give full permission to storage and bootstrap/cache folders by

```
chmod -R 777 storage bootstrap/cache
```

- Make sure that you edit the `.env` file correctly
- Make sure that you edit the `.env.production` file correctly

Once again, thank you so much for using EZBus. As I said at the beginning, I'd be glad to help you if you have any questions. I'll do my best to assist. If you have a more general question relating to the items on CodeCanyon, you might consider visiting the forums and asking your question in the "Item Discussion" section.

CreativeAppsDev Team

[Go To Table of Contents \(#toc\)](#)