

Single DB Doc

Software compiled using the below technologies along with versions:

1. PHP 8.1.24
2. MySQL : 11.0.2-MariaDB
3. Nginx: nginx/1.25.2 Or Apache: http 2.4.58
4. Composer: 2.5.8
5. Laravel: 9.52.16

Server Requirements

The Laravel framework consists of a few system requirements. You must ensure that your Web Server meets the minimum PHP version and extensions as follows:

```
PHP >= 8.0
Ctype PHP Extension
cURL PHP Extension
DOM PHP Extension
Fileinfo PHP Extension
Filter PHP Extension
Hash PHP Extension
Mbstring PHP Extension
OpenSSL PHP Extension
PCRE PHP Extension
PDO PHP Extension
Session PHP Extension
Tokenizer PHP Extension
XML PHP Extension
```

Server Configuration

```
server {
    listen 80;
    listen [::]:80;
    server_name example.com;
```

```
root /srv/example.com/public;

add_header X-Frame-Options "SAMEORIGIN";
add_header X-Content-Type-Options "nosniff";

index index.php;

charset utf-8;

location / {
    try_files $uri $uri/ /index.php?$query_string;
}

location = /favicon.ico { access_log off; log_not_found off; }
location = /robots.txt { access_log off; log_not_found off; }

error_page 404 /index.php;

location ~ \.php$ {
    fastcgi_pass unix:/var/run/php/php8.0-fpm.sock;
    fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;
    include fastcgi_params;
}

location ~ /\.well-known.* {
    deny all;
}
}
```

Installation

1. Go to terminal and run the command below.

```
# For Linux
sudo cp env.saas.hrm .env

# For Windows
cp env.saas.hrm .env
```

If you have no terminal, copy `env.saas.hrm` and paste OR rename as `.env`

2. Go to the `phpmyadmin` & create database `hrm`

In `*.env` file, update the database credentials like such -

```
DB_HOST=localhost  
DB_PORT=3306  
DB_DATABASE="hrm"  
DB_USERNAME="root"  
DB_PASSWORD="password"
```

3. If you have an ssl certificate, then update the `*.env` file `APP_HTTP` & `APP_URL` values.

```
APP_HTTPS=true  
APP_URL="https://${APP_DOMAIN}"
```

Or leave it as such -

```
APP_HTTPS=false  
APP_URL="http://${APP_DOMAIN}"
```

Update APP_DOMAIN as your domain

```
APP_DOMAIN="onesttech.com"
```

If you are using sub domain

```
APP_DOMAIN="hrm.onesttech.com"
```

Set

```
APP_BRANCH="MultiBranch"
```

If you don't have any vendor directory in the root folder. Then run this command if you have available Terminal.

```
composer update
```

Run

```
php artisan key:generate
```

4. Go to the terminal and run the command below.

```
php artisan module:enable Saas
```

If you have no terminal, open `modules_statuses.json` and update `{"Saas": false}` to `{"Saas": true}`. If you see already done, just skip this step

5. Go to the terminal and run the command below.

```
php artisan saas:single-migrate --seed
```

if you have no terminal, open `database/sqls` folder, go to the `phpmyadmin` or select the database and import `saas_hrm.sql`

6. Login Panel

```
your_domain.com/sign-in
```

Default Email: admin@onesttech.com

Password: 12345678

Note: Change your temporary password

For Live Tracking:

To create a new server API key, use the Cloud Messaging tab of the Settings panel of your firebase project:

1. Click on Settings > Cloud Messaging. The section Cloud Messaging API (Legacy) will show as disabled

The screenshot shows the 'Project settings' page with the 'Cloud Messaging' tab selected. There are two main sections: 'Firebase Cloud Messaging API (V1)' which is enabled, and 'Cloud Messaging API (Legacy)' which is disabled. A call-to-action button 'Manage API in Google Cloud Console' is visible in the bottom right corner of the disabled section.

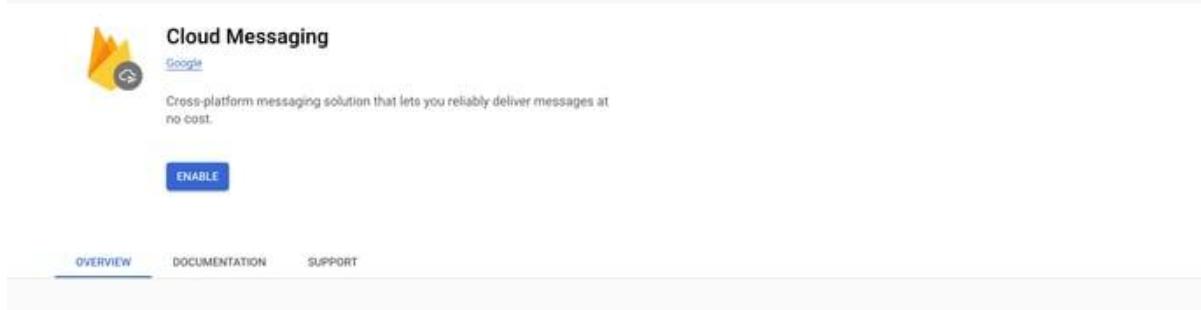
Section	Status
Firebase Cloud Messaging API (V1)	Enabled
Cloud Messaging API (Legacy)	Disabled

2. Click the vertical ellipsis (three dots) and click Manage API in Google Cloud Console

The screenshot shows the 'Project settings' page with the 'Cloud Messaging' tab selected. The interface is identical to the previous screenshot, displaying the status of both the V1 and Legacy APIs. A prominent call-to-action button 'Manage API in Google Cloud Console' is located in the bottom right corner of the 'Cloud Messaging API (Legacy)' section.

Section	Status
Firebase Cloud Messaging API (V1)	Enabled
Cloud Messaging API (Legacy)	Disabled

3. This will open the Google Cloud Console. Click on the ENABLE button



4. Back in the project settings, the Cloud Messaging API (Legacy) will now show Enabled and have a server key. Copy the new server API key from the Firebase settings panel to your clipboard.

A screenshot of the "Project settings" page in the Firebase console. The "Cloud Messaging" tab is active. In the first section, "Firebase Cloud Messaging API (V1)", it says "Enabled" with a green checkmark. Below it, a note says "Recommended for most use cases. [Learn more](#)". In the second section, "Cloud Messaging API (Legacy)", it also says "Enabled" with a green checkmark. A note below it says "If you are newly integrating messaging into your app, use the latest Firebase Cloud Messaging API (V1). If you are an existing user of Cloud Messaging API (Legacy), consider migrating to the latest Firebase Cloud Messaging API (V1). [Learn more](#)". Both sections have a "Server key" field, which contains a long, blurred string of characters. At the bottom right of the "Cloud Messaging API (Legacy)" section is a blue "Add server key" button.

5. Add the API key to your Ably Notification app dashboard

Then paste in the newly generated server API key into the text box beneath "Setting up Google/Firebase Cloud Messaging" and click "Save GCM/FCM settings". Once saved, the page should look something like:

Push Notifications

Ably can deliver native push notifications to devices using, amongst others, Apple Push Notifications and Firebase Cloud Messaging.

Find out more about [how to use Ably's push notification service](#)

Push Notifications Setup

[Configure Push](#)

Before you can use Ably's push notification services, you must provide credentials for the third party push notification services you wish to integrate with below.

Firebase Cloud Messaging	Apple Push Notification Service
Key AAAAf... AAA...tS...	Not configured Click "Configure push" to set up.

For Live Tracking you have to update the following files -

Go to firebase console panel, generate firebase credentials as [firebase.json](#) and keep root directory and update in setting-

The screenshot shows the HRM OneTech administrative interface. On the left, there is a sidebar with various menu items like Conference, Payroll, Accounts, Awards, Travels, Performance, Appointment, Visit, Support, Announcement, Contacts, Report, Configurations, and Settings. The 'Settings' item is selected, and its sub-menu includes General Settings, App Setting, Currency, and Language. The main content area is titled 'Settings' and has a sub-section titled 'Firebase Setup'. This sub-section contains fields for FIREBASE API KEY, FIREBASE AUTH DOMAIN, FIREBASE AUTH DATABASE URL, FIREBASE AUTH PROJECT ID, FIREBASE AUTH STORAGE BUCKET, FIREBASE AUTH SENDER ID, FIREBASE AUTH APP ID, and FIREBASE AUTH MEASUREMENT ID. A red box highlights the 'Firebase Setup' tab and the entire 'Firebase Setup' form area. At the top of the main content area, there are navigation links for Garcia and Robertson Plc, English, 03:18:00 PM, and user status indicators for Omar Cantu.

Google Maps API key:

Google Maps requires a special Google Maps API key. Without this key, you cannot display Google Maps on your website. There used to be two separate keys, the Google Maps API browser key and the Google Maps API GeoCoding key.

Generate a new API key

1. Go to the [Google Maps Platform](#)
2. Click the Get Started button in the middle of the screen.
3. Click on the Google Cloud Platform home in the upper left corner.
4. Click on Billing to make sure your billing details are up-to-date. If they are not, your Google Maps will not work properly.
5. Once you've confirmed your billing is up-to-date, click on the Google Cloud Platform home in the upper left corner again.
6. Hover to APIs & Services and go to Credentials.
7. If you want to use an existing project, please select it from the list. Otherwise, select 'Create a new project' and enter a project name.
8. Click Create credentials and select API key. You will see a new dialog that displays the newly created API key.

+ CREATE CREDENTIALS

 **DELETE**

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

9. Click the Close button in the API key dialogue. Your new API key will be listed on the Credentials page under API keys.

API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

Your API key



Restrict your key to prevent unauthorized use in production.

CLOSE

RESTRICT KEY

View your existing API keys

1. Go to the [Google Maps Platform](#)
2. If the side menu is not visible, click the three-line (hamburger) menu icon
3. Click 'APIs & Services' (API icon)
4. Click 'Credentials' (key icon)

Update **GEOCODING_API_KEY** from google map api key.

The screenshot shows the HRM OnsiteTech application interface. The left sidebar has a 'Settings' section with a 'General Settings' option selected. The main content area is titled 'Settings' and contains a 'Geocoding setup' tab. This tab includes fields for 'GEOCODING API KEY' and 'GEOCODING BASE URL', both of which are highlighted with a red box. A blue 'Save' button is located at the bottom right of the form. The top navigation bar shows the company name 'Garcia and Robertson Plc', the language 'English', the time '03:17:00 PM', and the user 'Omar Cantu'. The bottom of the page has a copyright notice '©2024 Hrm All rights reserved'.

Payment Methods:

The screenshot shows the HRM OnsiteTech application interface. The left sidebar has a 'Settings' section with a 'General Settings' option selected. The main content area is titled 'Settings' and contains a 'Payment Gateway' tab. This tab includes fields for 'Stripe Key' and 'Stripe Secret', both of which are highlighted with a red box. Below this, there are sections for 'Demo Checkout' (with a toggle switch turned on) and 'Offline Payment Type' (with options for 'Cash', 'Cheque', and 'Bank Transfer', all with toggle switches turned on). A blue 'Save' button is located at the bottom right of the form. The top navigation bar shows the company name 'Garcia and Robertson Plc', the language 'English', the time '03:15:12 PM', and the user 'Admin'. The bottom of the page has a copyright notice '©2024 Hrm All rights reserved'.

Email Setup:

The screenshot shows the 'Settings' page of the OpenHRM Onestech application. The left sidebar has a 'Settings' dropdown expanded, with 'General Settings' selected. The main content area is titled 'Email Setup [SMTP]' and contains the following fields:

Setting	Value
MAIL HOST *	smtp.gmail.com
MAIL PORT *	587
MAIL USERNAME *	MAIL USERNAME
MAIL FROM ADDRESS *	MAIL FROM ADDRESS
MAIL PASSWORD *	MAIL PASSWORD
MAIL ENCRYPTION *	tls
MAIL FROM NAME *	Onestech

A red box highlights the 'Email setup' tab in the top navigation bar and the entire 'Email Setup [SMTP]' form area.