

# Installation

## Server Requirements

For running app make sure you have:

- PHP >= 8.0.2 version (8.1+ recommended)
- MySQL 5.6+
- Nginx or Apache (Nginx recommended)

These PHP Extensions are require:

- Intl
- GD
- BCMath
- PDO
- SQLite3
- Ctype
- Fileinfo
- JSON
- Mbstring
- OpenSSL
- Tokenizer
- XML
- Exif

## Installation

- [How to install VPS with Debian 10](#)
- [How to Set Up AWS S3](#)
- [How to Set Up Digital Ocean Spaces](#)

## 1. Upload files on your server

Upload project files to the web root folder of your domain. It's mostly located in `html`, `www` or `public_html` folder name.

## 2. Configure your domain host root folder

Configure your web server's document root to point to the public directory of the files you previously uploaded. For example, if you've uploaded the files in `html` folder, your domain root directory should be changed to `html/project_files/public` folder or anything else where domain root is in `project /public` directory.

## 3. Set write permissions

Set `755` permission (CHMOD) to these files and folders directory within all children subdirectories:

- `/bootstrap`
- `/storage`
- `/.env`

## 4. Open your application in your web browser

Then open your application in web browser. If everything works fine, you will be redirected to the setup wizard installation process.

## 5. Server Check

On the first page you will see server check. Make sure all items are green. If not, then correct your server setup by recommended values and refresh your setup wizard page.

## 6. Follow setup wizard steps

That was the hardest part of installation process. Please follow instructions in every step of Setup Wizard to successfully install VueFileManager.

## 7. Set up Cron

### If you are running VueFileManager on shared web hosting (CPanel, Plesk etc.)

1. Create new cron job

2. Set execution cycle every minute
3. Search the absolute directory path where you uploaded VueFileManager files (like `/www/project_files`). The path must start with `/`.
4. Copy the command below, paste it to the command text area and replace in command string `replace_by_your_path` exactly with your path you found in step 3.
5. It should look like this with your pasted project path.

```
php replace_by_your_path/artisan schedule:run >> /dev/null 2>&1
```

## If you are running VueFileManager on linux server

1. Search the absolute directory path where you uploaded VueFileManager files (like `/www/project_files`). The path must start with `/`.
2. Copy the command below, paste it to your cron list and replace in command string `/www/project_files` exactly with your path you found in step 1.

```
* * * * * cd /www/project_files && php artisan schedule:run >> /dev/null 2>&1
```

## 8. CORS Configuration (If you Set External Storage s3 Service)

In your s3 bucket settings you should have option to set up your CORS (Cross-Origin Resource Sharing). It's basically adding your app url to the list of allowed CORS. This step is required for reading pdf documents from s3 in your VueFileManager app without loading issues.

## 9. Broadcasting

Coming soon...

## Nginx Configuration

If you running VueFileManager under Nginx, don't forget set this value in your `nginx.conf` file:

```
http {
    client_max_body_size 1024M;
}
```

And example Nginx config for your domain:

```
server {
    listen 80;
    listen [::]:80;

    # Log files for Debugging
    access_log /var/log/nginx/laravel-access.log;
    error_log /var/log/nginx/laravel-error.log;

    # Webroot Directory for Laravel project
    root /var/www/vuefilemanager/public;
    index index.php index.html index.htm;

    # Your Domain Name
    server_name example.com;
    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    # PHP-FPM Configuration Nginx
    location ~ \.php$ {
        try_files $uri =404;
        fastcgi_split_path_info ^(.+\.(php|\.php))(/.+)$;
        fastcgi_pass unix:/run/php/php7.3-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        include fastcgi_params;
    }
}
```

## Apache Configuration

Make sure you have enabled `mod_rewrite`. There is an example config for running VueFileManager under apache:

```
<VirtualHost example.com:80>
    DocumentRoot /var/www/vuefilemanager/public
    ServerName example.com

    <Directory "/var/www/vuefilemanager/public">
        AllowOverride All
        allow from all
        Options +Indexes
        Require all granted
    </Directory>

    RewriteEngine on
    RewriteCond %{SERVER_NAME} =example.com
    RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
</VirtualHost>
```

# Developers

## Running development environment on your localhost

For running development environment make sure you have:

- Node >= 14
- NPM >= 6

If you would like to express set up, please update your database credentials in .env file

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=laravel
DB_USERNAME=root
DB_PASSWORD=
```

If you would like to generate demo content, run this command below. Demo account will be created with credentials `howdy@hi5ve.digital` and password `vuefilemanager`.

```
php artisan setup:dev
```

If you would like express installation without demo data, run this command below. Demo account will be created with credentials `howdy@hi5ve.digital` and password `vuefilemanager`.

```
php artisan setup:prod
```

After that, please make sure your current host/domain where you are running app is included in your .env SANCTUM\_STATEFUL\_DOMAINS variable.

To start server on your localhost, run command below.

```
php artisan serve
```

For developing Vue front-end, you have to install npm modules by this command:

```
npm install
```

To compile and hot-reloads for front-end development. Then run this command:

```
npm run hot
```

To compile for production build, run this command

```
npm run prod
```

## Support

The following support channels are available at your fingertips:

- [CodeCanyon support message](#)

## Supporting VueFileManager

We are trying to make the best for VueFileManager. There are a lot of things to do, and a lot of features we can make. But, it can't be done without you, development is more and more complicated and we have to hire new colleagues to help us. There is couple way you can support us, and then, we support you with all great new features we can make. Thank you for participating on this awesome application!

- [Buy me a Coffee](#)
- [Become a Patreon](#)
- [One-time donation via PayPal](#)

## Security Vulnerabilities

If you discover a security vulnerability within this project, please send an e-mail to [peterpapp@makingcg.com](mailto:peterpapp@makingcg.com). All security vulnerabilities will be promptly addressed.