Install Ticaga

Let's log into the new user we created earlier:

```
su ticaga
```

Let's give our **www** folder the correct permissions:

```
sudo chown -R $USER /var/www/
```

Let's go to the www folder:

```
cd /var/www/
```

Let's create the Laravel project which we will use for Ticaga:

```
composer create-project --prefer-dist laravel/laravel ticaga
```

Provide permissions to the ticaga folder:

```
sudo chown -R $USER /var/www/ticaga
```

Now let's go into the ticaga folder:

```
cd ticaga
```

Let's remove all the default files from Laravel:

```
rm -rf ./* ./.env ./.env.example ./composer.lock
```

THIS IS AN IMPORTANT STEP AND DANGEROUS, ENSURE THE ./ IS BEFORE THE *, ELSE YOU WILL DELETE THE WHOLE SERVER!

If you are worried about breaking your server, please contact our support team or open a sales ticket if you have a trial license.

Let's get the latest version of Ticaga:

```
wget -N https://ticaga.com/latest.zip
```

Unzip the files:

```
unzip latest.zip
```

Let's install NodeJS for Ticaga:

```
sudo apt install nodejs -y
sudo apt install npm -y
```

Now we've done everything we need, it's time to edit the .env file for Ticaga.

```
sudo mv .env.example .env
```

and let's edit the newly created .env:

```
sudo nano .env
```

or you can use vi to edit:

```
sudo vi .env
```

You can now edit the main settings:

```
APP_NAME="Ticaga Demo"
```

This is your company name.

```
APP_URL=https://demo.ticaga.com
```

This is your URL to your Ticaga, sub-domain or main domain if you prefer.

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=ticaga
DB_USERNAME=ticaga_user
DB_PASSWORD=p@s3w0r$2024!
```

These are the database name, username and password you created when doing the Database.

Let's provide the permissions for the bootstrap and storage folders:

```
sudo chmod -R ugo+rw bootstrap storage
```

Now we need to run the composer command to create the vendors folder for Ticaga, without this Ticaga won't work. We don't ship the vendors folder as it is huge:

```
composer install
```

Now we need to generate a secure App Key:

```
sudo php artisan key:generate
```

Let's sort the storage link out so you can upload avatars:

```
sudo php artisan storage:link
```

Let's create the database tables for Ticaga and seed the database with content pre-filled for Ticaga:

```
sudo php artisan migrate --seed
```

Now let's install Vite (this ensures nothing goes wrong during set-up):

```
sudo npm install
sudo npm install -g vite
```

If you see this:

```
added 153 packages, and audited 154 packages in 8s

37 packages are looking for funding
  run `npm fund` for details

6 vulnerabilities (3 moderate, 3 high)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
```

Run the following:

```
sudo npm audit fix
```

Next upgrade NPM run:

```
sudo npm update
```

Publish Livewire Assets:

```
php artisan livewire:publish --assets
```

Need to replace the BladeCompiler for Laravel for Ioncube to work:

```
sudo mv ./hot_fix/vendor/laravel/framework/src/Illuminate/View/Compilers/BladeCompiler.php
./vendor/laravel/framework/src/Illuminate/View/Compilers/BladeCompiler.php
```

Lets now build Ticaga:

```
sudo npm run build
```

Clear Ticaga's caches:

```
php artisan optimize:clear
```

Reboot the server to kick Ioncube into action:

```
sudo reboot
```

Log back into the server and go to Ticaga:

```
cd /var/www/ticaga
```

Log into the username:

```
su ticaga
```

Load up Ticaga:

```
php artisan up
```

That's it now you have a fully working Ticaga.