

# Ezda local configuration

## Requirements:

- PHP 8.0.14 (if not accessible then use 8.1.33(tested) or install 8.3 (not tested))
- Composer (<https://getcomposer.org/doc/00-intro.md#installation-windows>)
- env file (<https://drive.google.com/drive/folders/1l4d3HQ01QNQk4GydQpBzbP5kPtQ85id3?usp=sharing>)
- Sql file (<https://drive.google.com/drive/folders/1l4d3HQ01QNQk4GydQpBzbP5kPtQ85id3?usp=sharing>)
- storage file (<https://drive.google.com/drive/folders/1l4d3HQ01QNQk4GydQpBzbP5kPtQ85id3?usp=sharing>)
- MySQL and MySQL Workbench (<https://dev.mysql.com/downloads/installer/>)
- Github repo of project (<https://github.com/Lshiroc/ezda>)

## Installation:

- Download php and unzip php to C:/php and configure path from windows environment variables
- open CMD and write php -v for check php version is installed or not if installed go next steps
- Go C:/php or where you installed php find php.ini file and remove ; from listed entries

extension=fileinfo

extension=ftp

extension=mbstring

extension=exif (exif must placed under of mbstring )

extension=pdo\_mysql

- Clone Github repo to your local (<https://github.com/Lshiroc/ezda>) you need to ask access because it is private repo

- After successful cloning project from github you need to get env file and storage folder from google drive , unzip and copy it to projects root folder
- Download **MySQL** and **MySQL Workbench**. After installing them, update the database-related values in your `.env` file according to your local setup, such as `DB_PORT` , `DB_HOST` , `DB_USERNAME` , `DB_PASSWORD` , etc.
- Connect to your local server through MySQL Workbench and create a new database. To run the website locally, you need the SQL file of the database. you can install it from shared link
- Next, go to the project and run `composer install` in the terminal.
- If `composer install` gives an error related to `php.ini` , you need to locate the `php.ini` file in your PHP installation directory and edit it. For example, if you receive an error such as **“curl is not enabled”** or **“not found”**, search for the corresponding extension in the `php.ini` file. You will see a semicolon (;) in front of it — remove that semicolon. It is used to deactivate the extension. Then run `composer install` again. It will probably show more errors; repeat the same process for each missing extension.
- After that, run: `php artisan storage:link`
- Then run: `php artisan serve`
- The project should open at `localhost:8000` . Images, CSS, and everything else should work fine. Some images for newer news posts may not appear because the SQL file you imported is newer than the files inside your local storage folder. In other words, some of the latest news images may not exist in the local storage. But everything else should work without issues.

Proyekt haqqında ümumi:

Style:

- Proyektə bootstrap istifadə olunur
- Sass üçün fayl main.scss adlanır. output main.css faylıdır. orada başqa main5, main6.css faylları da var amma onlar heç yerdə istifadə olunmur.
- Bəzən stillər hər səhifənin içərisində də yazılır `<style>` ilə

Pages:

- Əsas veb saytın səhifələri `resources/views`-da yerləşir. 3cür layout var `page`, `post` və `search` pagelər demək olar bütün ümumi səhifələr üçün istifadə olunur, `post` xəbər və s tipli şeylər üçün istifadə olunur

search də ancaq search səhifəsi üçün istifadə olunur

- Qalan səhifələr əsas olaraq həmin resources/views/templates-də yerləşir hansı ki layout-ların içərisində olur. Məsələn:

```
<page-layout>
```

```
<page-about>
```

```
</page-layout>
```

Amma kodda biraz fərqli import edirlər, bu sadəcə nümunədi.

- Views-in içərisindəki digər folderlərdə də səhifələr var amma əsas bu qeyd elədiklərim istifadə olunur.
- Footer /resources/views/includes-da yerləşir ümumidir.
- Amma header hər /resources/views/layouts-da hər üç page, post və search üçün təkrar yazılır, yəni bir dəfə import edilməyib.

Admin:

- Admin səhifəsi üçün kodlar /resources/js də yerləşir. Frontend vue-da yazılıb
- Build etmək üçün npm run mix istifadə edirsən amma başqa tipdə kommandlarda var, package.json-a baxa bilərsən

Config:

- config folderində ümumi configlər yazılıb, ayrı-ayrı hərəsi nə mənaya gəlir əmin deyiləm. Çox vaxt edit etmək lazım olmur.

Models:

- app/Models folderində backend üçün hər item üçün(məsələn post, page, user və s) modellər yazılıb, içərisində onların fieldləri, methodları var

API:

- app/HTTP-də API-lər yerləşir.

Admin panel:

- <http://localhost:8000/adm1nscippanel> Admin panel bu linkdədi  
email: [wkohler@example.net](mailto:wkohler@example.net) password: password
- Admin panel-də ilk açılan Dashboard səhifəsinin boş olması normaldır.

