

# 部署文档

## 一、概述

本项目使用 Vue3 + Yii2的前后端分离模式。

- 前端：使用Vue.js框架，通过组件化的开发方式构建用户界面。Vue3提供了响应式数据绑定、虚拟DOM、路由管理（通过router/index.js）和状态管理等核心功能，使得开发过程更加高效且易于维护。
- 后端：采用Yii2框架，遵循MVC三层架构设计，包括Model、Controller和View。Yii2框架提供了丰富的功能和工具，用于处理请求、路由、数据库访问、身份验证等常见的后端任务。
- 交互：前后端之间的通信通过RESTful API完成，前端使用Axios等HTTP库发送请求到后端的API接口，后端处理请求并返回JSON格式的数据。
- 数据库：本项目使用MySQL进行数据存储和管理，利用Yii2框架的原生支持，实现高性能的数据处理和存储，保障系统的可靠性和可扩展性。

## 二、部署说明

### 前期准备

确保以下工具已安装并配置好相关环境变量：

- PHP
- Node.js
- NPM
- MySQL

### 获取项目代码

从本组 Github 仓库将代码克隆到本地：

```
git clone https://github.com/ljtZM/Internet_Database_Development-Team_Project.git
```

### 部署前端

1. 进入前端目录：

```
cd my-frontend
```

2. 请你依次运行以下命令，安装前端项目所需依赖项：

```
npm install
npm install @fortawesome/fontawesome-free
```

3. 如果你运行完上述指令并且没有出现问题，就可以执行下面的命令，启动前端开发服务器：

```
npm run serve
```

显示App运行信息后，即可访问网站 <http://localhost:8081/> 查看本组项目。

到达第3步，已经完成本次作业要求。如果后续需要将本项目部署到Web 服务器，可以执行以下步骤：

4. 在项目根目录下，运行以下命令

```
npm run build
```

这将生成一个 [dist](#) 文件夹，其中包含构建后的前端资源文件（HTML、CSS、JavaScript 文件等）。

5. 上传并配置服务器：

将生成的 dist 文件夹中的内容上传到 Web 服务器的目标目录，并根据服务器类型（如 Nginx 或 Apache）进行配置，确保服务器能够正确访问静态文件并设置相关的 URL 重写规则。

6. 访问部署后的前端：

通过浏览器访问网站 URL，即可查看和使用部署后的 Vue 前端项目。

## 部署后端

1. 进入后端目录：

```
cd backend
```

1. 安装项目所需的 PHP 依赖：

```
composer install
```

2. 数据库配置：

- 在 MySQL 中创建一个名为 [yii2basic](#) 的数据库
- 运行data目录下的[install.sql](#)文件
- 打开 [backend/config/db.php](#) 文件，修改数据库的用户名和密码

3. 启动后端服务器：

```
php yii serve
```

后端服务启动成功后，通过 `http://localhost:8080` 访问后端接口。

### 三、一键部署功能

为了方便部署，本组创建了一个批处理脚本，自动化执行部署步骤。具体内容如下：

前提条件：

- 在MySQL数据库中创建名为yii2basic的数据库。
- 运行data/install.sql文件，完成了数据库表的创建和初始化数据导入

执行流程：

- 修改数据库配置：输入 MySQL 密码，自动更新 backend/config/db.php 中的数据库连接密码。
- 后端部署：自动安装composer依赖项并启动服务。
- 前端部署：自动安装前端npm依赖，并启动服务监听在8081端口。

代码如下：

```
@echo off
chcp 65001 > nul
setlocal enabledelayedexpansion

echo =====
echo Script execution begins, please follow the prompts...
echo =====

:: Prompt user for MySQL password
set /p db_password=Please enter your MySQL password:

:: Create a temporary PowerShell script to update the database password
set tmp_ps1=%temp%\update-db-password.ps1
echo $mysql_password = '%db_password%' > %tmp_ps1%
echo $config_file_path = "backend/config/db.php" >> %tmp_ps1%
echo (Get-Content $config_file_path) ^| ForEach-Object {$ _ -replace "'password' =>
'', "'password' => '$mysql_password'"} ^| Set-Content $config_file_path >>
%tmp_ps1%
echo Write-Host "Database password updated successfully!" >> %tmp_ps1%

:: Execute the PowerShell script with a temporary change to the execution policy
PowerShell -NoProfile -ExecutionPolicy Bypass -File %tmp_ps1%

:: Clean up the temporary PowerShell script
del %tmp_ps1%

:: Move to backend directory for Composer installation
cd backend
echo Installing Composer dependencies...
start /wait cmd /c "composer install && echo Composer installation completed! &&
exit"

echo Starting backend server...
```

```

start cmd /k "echo Starting PHP backend server... && php yii serve && echo Backend
server started successfully!"

:: Return to the parent directory
cd /d "%~dp0"

:: Frontend deployment starts
echo =====
echo Frontend deployment begins...
echo =====

:: Enter frontend directory and install dependencies
cd my-frontend
echo Installing frontend dependencies...
start /wait cmd /c "npm install && npm install @fortawesome/fontawesome-free &&
echo Frontend dependencies installation completed! && exit"

:: Start frontend server
echo Starting frontend development server...
start cmd /k "set VUE_APP_PORT=8081 && npm run serve && echo Frontend server
started successfully!"

:: Completion message
echo =====
echo Script execution completed, frontend and backend services have been started.
echo =====
pause

```

在命令行执行如下命令：

```
deploy.bat
```

成功开启前后端服务：

The image shows two terminal windows side-by-side. The left window, titled 'C:\WINDOWS\system32\cmd.', shows the output of the PHP backend server starting, displaying the document root and the server URL (http://localhost:8080/). The right window, also titled 'C:\WINDOWS\system32\cmd.', shows the output of the Vue.js frontend development server. It displays a warning about a module not found, the app running at http://localhost:8081/ and http://10.130.111.208:8081/, and a note that the development build is not optimized.

```

C:\WINDOWS\system32\cmd. x  +  v  -  □  ×
Starting PHP backend server...
Server started on http://localhost:8080/
Document root is "D:\junior\WebDatabase\web\final\Internet_Data
base_Development-Team_Project-main\backend\web"
Quit the server with CTRL-C or COMMAND-C.
[Thu Dec 19 00:08:14 2024] PHP 8.3.7 Development Server (http:/
/localhost:8080) started

C:\WINDOWS\system32\cmd. x  +  v  -  □  ×
WARNING Compiled with 1 warning                                00:08:35
warning in ./src/components/MContainer.vue?vue&type=script&la
ng=js
Module not found: Error: Can't resolve '@assets/pic' in 'D:\ju
nior\WebDatabase\web\final\Internet_Database_Development-Team_P
roject-main\my-frontend\src\components'

App running at:
- Local:   http://localhost:8081/
- Network: http://10.130.111.208:8081/

Note that the development build is not optimized.
To create a production build, run npm run build.

```

日志输出:

```
D:\junior\WebDatabase\web\final\Internet Database Development-Team Project-main>deploy.bat
=====
Script execution begins, please follow the prompts...
=====
Please enter your MySQL password:123
Database password updated successfully
Installing Composer dependencies...
Starting backend server...
=====
Frontend deployment begins...
=====
Installing frontend dependencies...
Starting frontend development server...
=====
Script execution completed, frontend and backend services have been started.
=====
Press any key to continue . . .
```