

- 博客管理系统 - 部署文档
  - 1. 系统环境要求
  - 2. 数据库设置
    - 步骤 1: 安装并启动MySQL服务
    - 步骤 2: 创建数据库和表
      - 方式一：手动设置
  - 3. 后端项目配置
    - 1.  
blog/Manager/blog\_manager/src/main/resources/application.properties
    - 2. config.json
  - 4. 项目构建与启动
    - 步骤 1: 构建项目
    - 步骤 2: 启动应用
      - 方式一：使用Maven直接运行
      - 方式二：运行JAR包（推荐用于生产环境）
  - 5. 访问应用
- 前端项目部署
  - 环境要求
  - 步骤 1: 安装依赖
  - 步骤 2: 开发环境运行
    - 启动开发服务器
  - 完整应用访问
  - 常见问题解决
    - 前端访问后端接口失败
    - 数据库连接失败
    - 构建失败

# 博客管理系统 - 部署文档

---

本文档详细介绍了如何部署博客管理系统的后端服务。

## 1. 系统环境要求

---

在部署之前，请确保您的服务器或本地环境满足以下要求：

- **Java Development Kit (JDK):** Java 17。项目基于Java 17，请确保安装了正确版本的JDK。
- **Maven:** 3.6+。用于项目构建和依赖管理。
- **MySQL数据库:** 5.7+ 或 8.0+。建议使用8.0以上版本。
- **操作系统:** Windows, Linux, 或 macOS。

## 2. 数据库设置

---

数据库是本系统的核心，存储了所有用户和博客数据。

### 步骤 1: 安装并启动MySQL服务

如果您尚未安装MySQL，请先下载并安装。确保MySQL服务正在运行。

### 步骤 2: 创建数据库和表

我们提供了两种方式来初始化数据库：

#### 方式一：手动设置

1. 使用MySQL客户端（如MySQL Command Line Client, Navicat, DataGrip等）连接到您的数据库。
2. 执行根目录下的 `database_setup.sql` 脚本。这会创建名为 `blog_manager` 的数据库，并初始化所需的表（`users`, `blogs`）和数据（默认管理员账户）。

```
# 使用root用户登录MySQL
mysql -u root -p

# 执行SQL脚本
source database_setup.sql;
```

## 3. 后端项目配置

---

项目的主要配置文件有两个：

# 1.

## blog/Manager/blog\_manager/src/main/resources/application.properties

这个文件包含了Spring Boot应用的核心配置。

- **数据库连接:** 如果您没有使用 `setup_database.bat` 脚本，请务必手动检查并修改以下数据库连接信息，确保与您的环境匹配：

```
spring.datasource.url=jdbc:mysql://localhost:3306/blog_manager?
useSSL=false&serverTimezone=UTC&allowPublicKeyRetrieval=true
spring.datasource.username=your_mysql_username
spring.datasource.password=your_mysql_password
```

- **服务器端口:** 默认端口为 `5200`，您可以根据需要修改：

```
server.port=5200
```

## 2. config.json

此文件位于项目后端根目录，用于配置外部服务。

### blog/Manager/blog\_manager/config.json

```
{
  "xmodelAPIKey" : "your_xmodel_api_key", //星火大模型API调用KEY
  "bigmodelAPIKey" : "your_bigmodel_api_key" //智谱大模型API调用KEY
}
```

- `xmodelAPIKey`, `bigmodelAPIKey`: 系统集成的大模型服务API密钥。

**注意:** 博客文件自动存储在项目根目录的 `blogs` 文件夹中，图片文件自动存储在项目根目录的 `images` 文件夹中，无需额外配置。

## 4. 项目构建与启动

---

# 步骤 1: 构建项目

在项目根目录下打开终端或命令行，执行以下Maven命令来编译和打包项目：

```
mvn clean install
```

构建成功后，会在 `target/` 目录下生成一个 `blog_manager-0.0.1-SNAPSHOT.jar` 文件。

## 步骤 2: 启动应用

您可以通过以下两种方式启动后端服务：

### 方式一：使用Maven直接运行

在开发环境中，可以方便地使用以下命令启动：

```
mvn spring-boot:run
```

### 方式二：运行JAR包（推荐用于生产环境）

使用 `java -jar` 命令来运行构建好的JAR文件：

```
java -jar target/blog_manager-0.0.1-SNAPSHOT.jar
```

应用启动后，您将在控制台看到Spring Boot的启动日志。

## 5. 访问应用

当后端服务成功启动后：

- **API服务地址:** <http://localhost:5200>

您可以使用API测试工具（如 Postman, Insomnia）来访问后端接口，或与前端应用进行集成。

# 前端项目部署

---

前端项目位于 `Vue/blog_manager` 目录，基于 Vue 3 + TypeScript + Vite 构建。

## 环境要求

- **Node.js:** 16.0+ 或 18.0+（推荐使用最新 LTS 版本）
- **npm:** 8.0+ 或 **yarn:** 1.22+

## 步骤 1: 安装依赖

打开项目目录 `Vue/blog_manager`，在终端或命令行中执行：

```
cd Vue/blog_manager
npm install
```

## 步骤 2: 开发环境运行

启动开发服务器

```
npm run dev
```

前端开发服务器将在 `http://localhost:5174` 启动，并自动配置了代理到后端服务 `http://localhost:5200`。

**注意:** 请确保后端服务已经在 `localhost:5200` 启动，前端项目已配置好 API 代理。

## 完整应用访问

---

当前端和后端都成功部署后：

- **前端应用地址:** `http://localhost:5174`（开发环境）或您配置的域名
- **后端API地址:** `http://localhost:5200`

# 常见问题解决

---

## 前端访问后端接口失败

1. 确认后端服务已启动且运行在 `localhost:5200`
2. 检查前端 `vite.config.ts` 中的代理配置是否正确
3. 确认防火墙没有阻止相关端口

## 数据库连接失败

1. 确认MySQL服务已启动
2. 检查 `application.properties` 中的数据库连接信息
3. 确认数据库用户具有足够的权限

## 构建失败

1. 确认Node.js版本符合要求（16.0+）
2. 删除 `node_modules` 文件夹并重新安装依赖
3. 检查网络连接，确保能正常下载依赖包