

- [WANFi软件编译手册](#)
  - [项目介绍](#)
  - [软件工程项目目录树](#)
  - [后端模块编译说明](#)
  - [客户平台前端编译说明](#)
  - [用户平台前端编译说明](#)
  - [安装包构建说明](#)

# WANFi软件编译手册

---

## 项目介绍

---

WANFi远距离WiFi产品设计软件服务和计费系统，主要用于WiFi AP设备管理、用户管理及费用管理，系统包含客户平台和用户平台，客户平台包含设备管理和用户套餐计费管理及系统管理等功能；用户平台主要面向终端用户，终端用户可通过充值和购买流量套餐实现互联网接入服务。WANFi控制平台目前基于TPLINK Omada平台API实现。

本项目基于Spring Cloud、Ant Design Vue的前后端分离的开发平台，采用微服务解决方案，包括核心模块：角色用户、菜单权限、网关路由、数据权限等，使用中间件：日志脱敏、任务调度、短信邮箱等。

## 软件工程项目目录树

---

```
tree -L 1 wfc.git
wfc.git
├── be.wfc           # 后端模块目录
├── build.wfc       # 系统构建目录
├── fe.wfc          # 客户平台前端目录
└── fe.wfc.user    # 用户平台前端目录
```

## 后端模块编译说明

---

```

tree -L 2 be.wfc/
be.wfc/
├── LICENSE
├── README.md
├── bin
│   ├── clean.bat
│   ├── package.bat
│   ├── run-auth.bat
│   ├── run-gateway.bat
│   ├── run-modules-file.bat
│   ├── run-modules-gen.bat
│   ├── run-modules-job.bat
│   ├── run-modules-system.bat
│   └── run-monitor.bat
├── docker
│   ├── copy.bat
│   ├── copy.sh
│   ├── deploy.bat
│   ├── deploy.sh
│   ├── docker-compose.yml
│   ├── mysql
│   ├── nacos
│   ├── nginx
│   ├── redis
│   ├── start-backend.bat
│   ├── start-backend.sh
│   ├── start-frontend.bat
│   ├── start-frontend.sh
│   └── wfc
├── pom.xml
├── sql # 数据库脚本
│   ├── upgrade
│   ├── wfc_config_db # nacos配置数据库
│   ├── wfc_system_db # 客户平台数据库
│   └── wfc_user_db # 用户平台数据库
├── wfc-api # API接口模块
│   ├── pom.xml
│   ├── wfc-api-omada # Omada API接口
│   ├── wfc-api-system # 客户平台 API接口
│   └── wfc-api-user # 用户平台 API接口
├── wfc-auth # 认证中心 [8081]
│   ├── pom.xml
│   ├── src
│   └── target
├── wfc-common # 通用模块
│   ├── pom.xml
│   ├── wfc-common-core # 核心模块
│   ├── wfc-common-datascope # 权限范围
│   ├── wfc-common-datasource # 多数据源
│   ├── wfc-common-log # 日志记录
│   ├── wfc-common-mail # 邮件服务
│   ├── wfc-common-mybatis # 数据操作
│   ├── wfc-common-redis # 缓存服务
│   ├── wfc-common-seata # 分布式事务
│   ├── wfc-common-security # 安全模块
│   └── wfc-common-sensitive # 数据脱敏

```

```
├── wfc-common-swagger # 系统接口
├── wfc-gateway # 网关模块 [8080]
│   ├── pom.xml
│   ├── src
│   └── target
├── wfc-modules # 业务模块
│   ├── pom.xml
│   ├── wfc-file # 文件服务 [9201]
│   ├── wfc-gen # 代码生成 (未启用)
│   ├── wfc-job # 定时任务 [9203]
│   ├── wfc-payment # 支付服务 [9204]
│   ├── wfc-system # 客户平台模块 [9205]
│   ├── wfc-user # 用户平台模块 [9206]
├── wfc-visual # 系统监控 (未启用)
│   ├── pom.xml
│   └── wfc-visual-monitor
```

## 环境部署

```
JDK >= 1.8 (推荐1.8版本)
Maven >= 3.6.3
Mysql >= 5.7.0 (推荐5.7版本)
Redis >= 7.4.2
nacos >= v2.5.0
sentinel >= 1.6.0 (optional)
```

## 编译系统

```
cd be.wfc
mvn clean package -P prod -DskipTests
```

# 客户平台前端编译说明

## 工程目录树

```
tree -L 1 fe.wfc
fe.wfc
├── README.md
├── build
├── dist
├── env.config.ts
├── eslint.config.js
├── index.html
├── node_modules
└── package.json
```

```
├─ packages
├─ pnpm-lock.yaml
├─ pnpm-workspace.yaml
├─ public
├─ src
├─ tsconfig.json
├─ uno.config.ts
├─ vite.config.ts
└─ vitest.config.ts
```

## 环境准备

确保你的环境满足以下要求：

- **NodeJS:**  $\geq 18.12.0$ ，推荐 18.19.0 或更高。
- **pnpm:**  $\geq 8.7.0$ ，推荐 8.14.0 或更高。

## 安装依赖

```
npm install -g pnpm
```

```
npm install -g pnpm
pnpm i
```

由于本项目采用了 pnpm monorepo 的管理方式，因此请不要使用 npm 或 yarn 来安装依赖。

## 启动项目

```
pnpm dev
```

## 构建项目

```
pnpm build
```

# 用户平台前端编译说明

## 工程目录树

```
tree -L 1 fe.wfc.user
fe.wfc.user
├── README.md
├── build
├── dist
├── env.config.ts
├── eslint.config.js
├── index.html
├── node_modules
├── package.json
├── packages
├── pnpm-lock.yaml
├── pnpm-workspace.yaml
├── public
├── src
├── tsconfig.json
├── uno.config.ts
├── vite.config.ts
└── vitest.config.ts
```

## 环境准备

确保你的环境满足以下要求：

- **NodeJS:**  $\geq 18.12.0$ ，推荐 18.19.0 或更高。
- **pnpm:**  $\geq 8.7.0$ ，推荐 8.14.0 或更高。

## 安装依赖

```
npm install -g pnpm
```

```
pnpm i
```

由于本项目采用了 pnpm monorepo 的管理方式，因此请不要使用 npm 或 yarn 来安装依赖。

## 启动项目

```
pnpm dev
```

## 构建项目

# 安装包构建说明

## 工程目录树

```
build.wfc/
├── bin                                # build binary file
├── build                              # build directory
│   ├── bin
│   ├── docker                        # docker compose root directory
│   │   ├── env
│   │   ├── mysql
│   │   │   ├── conf
│   │   │   ├── data
│   │   │   ├── db
│   │   │   └── logs
│   │   ├── nacos
│   │   │   ├── conf
│   │   │   └── logs
│   │   ├── nginx
│   │   │   ├── conf
│   │   │   ├── conf.d
│   │   │   ├── html
│   │   │   └── logs
│   │   ├── redis
│   │   │   ├── conf
│   │   │   └── data
│   │   └── wfc
│   │       ├── auth
│   │       │   └── jar
│   │       ├── gateway
│   │       │   └── jar
│   │       ├── modules
│   │       │   ├── file
│   │       │   │   └── jar
│   │       │   ├── gen
│   │       │   │   └── jar
│   │       │   ├── job
│   │       │   │   └── jar
│   │       │   ├── payment
│   │       │   │   └── jar
│   │       │   ├── system
│   │       │   │   └── jar
│   │       │   ├── user
│   │       │   │   └── jar
│   │       ├── upload
│   │       └── visual
│   │           └── monitor
│   │               └── jar
```

```
|   └─ systemd                               # system service daemon file saved here
|       └─ system
├─ debbuild                                  # deb build directory
|   └─ DEBIAN
|   └─ etc
|   └─ opt
|       └─ wfc
├─ docs
|   └─ 01-requirement
|   └─ 02-design
|   └─ 03-configuration
└─ release
    └─ debs                                 # deb package release
        └─ tars                             # tar package release
```

## 准备工作

```
JDK >= 1.8 (推荐1.8版本)
Maven >= 3.0
Mysql >= 5.7.0 (推荐5.7版本) # docker内部构建
Redis >= 7.4.2                # docker内部构建
nacos >= v2.5.0                # docker内部构建
sentinel >= 1.6.0 (optional)
```

## 构建发布包

如果不同步拉取git仓库，带-g

- TAR发布包

```
cd wfc.git/build.wfc/bin
./build.sh -g tar
```

- DEB发布包

```
cd wfc.git/build.wfc/bin
./build.sh -g deb
```