

H1 灵境万象健康管理系统 (LJWX)

H2 ★ 系统简介

灵境万象健康管理系统是一个基于Docker的企业级健康数据管理平台，支持穿戴设备数据采集、实时监控大屏、智能健康分析等功能。

H3 🏠 系统架构

- **ljwx-admin**: Vue.js前端管理界面 (端口8080)
- **ljwx-boot**: Spring Boot后端API (端口9998)
- **ljwx-bigscreen**: Python Flask大屏应用 (端口8001)
- **ljwx-mysql**: 定制化MySQL数据库 (端口3306)
- **ljwx-redis**: 定制化Redis缓存 (端口6379)

H2 🚀 快速开始

H3 1. 多架构镜像拉取

我们的镜像支持 **AMD64** 和 **ARM64** 双架构，可以在任何服务器上运行：

```
1 # 自动拉取适合当前架构的镜像
2 docker pull crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-mysql:1.1.0
3 docker pull crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-redis:1.1.0
4 docker pull crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-boot:1.1.0
5 docker pull crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-bigscreen:1.1.0
6 docker pull crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-admin:1.1.0
7
8 # 强制拉取AMD64版本（在AMD64服务器上）
9 docker pull --platform linux/amd64 crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-boot:1.1.0
10
11 # 验证镜像架构支持
12 docker buildx imagetools inspect crpi-yilnm6upy4pmbp67.cn-
  shenzhen.personal.cr.aliyuncs.com/ljwx/ljwx-boot:1.1.0
```

H3 2. 创建目录结构

```
1 # 创建外挂日志和数据目录
2 ./setup-external-volumes.sh
```

这将创建以下目录结构：

```
1 ./logs/
2 |— mysql/          # MySQL 日志文件
3 |— redis/          # Redis 日志文件
4 |— ljwx-boot/      # Spring Boot 应用日志
5 |— ljwx-bigscreen/ # Python 大屏应用日志
6 |— ljwx-admin/     # Nginx 访问和错误日志
7
8 ./data/
9 |— mysql/          # MySQL 数据文件(可选)
10 |— redis/         # Redis 数据文件(可选)
11 |— ljwx-boot/     # Spring Boot 应用数据
12 |— ljwx-bigscreen/ # Python 大屏应用数据
```

H3 3. 快速部署

```
1 # 使用标准配置启动 (Docker内部卷)
2 docker-compose up -d
3
4 # 使用增强配置启动 (日志和数据外挂)
5 docker-compose -f docker-compose-enhanced.yml up -d
```

H2 配置说明

H3 日志和数据外挂配置

我们提供两种部署方式：

H4 方式1：标准部署 (docker-compose.yml)

- 使用Docker命名卷存储数据和日志
- 数据和日志由Docker管理
- 适合开发和测试环境

H4 方式2：增强部署 (docker-compose-enhanced.yml)

- 日志和数据直接外挂到宿主机
- 便于监控、备份和调试
- 适合生产环境

增强部署特性：

```

1  # MySQL 数据持久化和日志外挂
2  mysql:
3      volumes:
4          - mysql_data:/var/lib/mysql          # 数据持久化
5          - ./logs/mysql:/var/log/mysql        # 日志外挂
6
7  # Spring Boot 应用日志外挂
8  ljwx-boot:
9      volumes:
10         - ./logs/ljwx-boot:/app/ljwx-boot-logs    # 应用日志
11         - ./data/ljwx-boot:/app/data              # 应用数据

```

H3 微信配置

在 `custom-config.env` 文件中配置微信告警：

```

1  # 微信小程序配置
2  WECHAT_APP_ID=wx10dcc9f0235e1d77
3  WECHAT_APP_SECRET=your_wechat_app_secret
4
5  # 微信模板消息配置
6  WECHAT_TEMPLATE_ID=your_template_id
7  WECHAT_USER_OPENID=your_user_openid
8
9  # 启用微信告警
10 WECHAT_ALERT_ENABLED=true

```

微信告警功能：

- 健康异常数据实时推送
- 设备离线状态通知
- 系统运行状态监控
- 定制化告警规则

H3 定制化配置

H4 1. 自定义Logo

```

1  # 将您的logo文件替换为 custom-logo.svg
2  cp your-logo.svg custom-logo.svg

```

H4 2. 自定义配置文件

```
1 # 编辑大屏应用配置
2 vim custom-config.py
3
4 # 编辑前端配置
5 vim custom-admin-config.js
```

H4 3. 自定义资源文件

```
1 # 在 custom-assets 目录放置自定义图片资源
2 mkdir -p custom-assets
3 cp your-images/* custom-assets/
```

H4 4. 数据库初始化

```
1 # 自定义数据库初始化脚本
2 vim client-data.sql # 业务数据
3 vim client-admin.sql # 管理员数据
```

H2 运维管理

H3 查看服务状态

```
1 # 查看所有服务状态
2 docker-compose ps
3
4 # 查看服务日志
5 docker-compose logs -f ljwx-boot
6 docker-compose logs -f ljwx-bigscreen
7
8 # 查看外挂日志（增强模式）
9 tail -f logs/ljwx-boot/*.log
10 tail -f logs/mysql/error.log
11 tail -f logs/ljwx-admin/access.log
```

H3 数据备份

```
1 # MySQL数据备份
2 docker-compose exec mysql mysqldump -u root -p123456 lj-06 > backup_$(date
3 +'%Y%m%d_%H%M%S').sql
4
5 # Redis数据备份
6 docker-compose exec redis redis-cli BGSAVE
```

H3 服务重启

```
1 # 重启单个服务
2 docker-compose restart ljwx-boot
3
4 # 重启所有服务
5 docker-compose restart
6
7 # 更新服务
8 docker-compose pull
9 docker-compose up -d
```

H2 访问地址

启动成功后，可以通过以下地址访问各个服务：

- 管理后台: <http://localhost:8080>
 - 默认账号: admin/123456
- 健康大屏: <http://localhost:8001>
 - 实时健康数据展示
- API接口: <http://localhost:9998>
 - Swagger文档: <http://localhost:9998/doc.html>
- 数据库: localhost:3306
 - 用户名: root
 - 密码: 123456
 - 数据库: lj-06

H2 环境变量配置

在 `custom-config.env` 中可以配置以下参数：

```
1 # 数据库配置
2 MYSQL_PASSWORD=123456
3 MYSQL_DATABASE=lj-06
4 MYSQL_USER=root
5
6 # Redis配置
7 REDIS_PASSWORD=123456
8
9 # 应用配置
10 VITE_APP_TITLE=穿戴管理演示系统
11 VITE_BIGSCREEN_URL=http://localhost:8001
12
13 # 微信配置
```

```
14 WECHAT_APP_ID=wx10dcc9f0235e1d77
15 WECHAT_APP_SECRET=your_secret
16 WECHAT_TEMPLATE_ID=your_template_id
17 WECHAT_USER_OPENID=your_openid
18 WECHAT_ALERT_ENABLED=true
```

H2 监控和告警

H3 健康检查

```
1 # 检查服务健康状态
2 docker-compose exec ljwx-boot curl http://localhost:9998/health
3 docker-compose exec ljwx-bigscreen curl http://localhost:8001/health
```

H3 性能监控

- CPU和内存使用情况
- 数据库连接状态
- Redis缓存命中率
- 接口响应时间

H2 故障排除

H3 常见问题

1. 服务启动失败

```
1 # 查看详细错误日志
2 docker-compose logs service_name
3
4 # 检查端口占用
5 netstat -tulpn | grep :8080
```

2. 数据库连接失败

```
1 # 检查MySQL服务状态
2 docker-compose exec mysql mysql -u root -p123456 -e "SELECT 1"
3
4 # 重置数据库密码
5 docker-compose exec mysql mysql -u root -p -e "ALTER USER
  'root'@'%' IDENTIFIED BY '123456';"
```

3. 微信告警不工作

- 检查WECHAT_APP_SECRET是否正确
- 验证WECHAT_USER_OPENID是否有效
- 确认模板消息ID是否匹配

4. 日志文件权限问题

```
1 # 修复日志目录权限
2 sudo chmod -R 755 logs/
3 sudo chown -R $USER:$USER logs/
```

H2 版本更新

H3 更新到最新版本

```
1 # 1. 备份当前数据
2 docker-compose exec mysql mysqldump -u root -p123456 lj-06 >
  backup_before_update.sql
3
4 # 2. 拉取最新镜像
5 docker-compose pull
6
7 # 3. 重新启动服务
8 docker-compose down
9 docker-compose up -d
10
11 # 4. 验证服务状态
12 docker-compose ps
```

H2 技术支持

如果在部署过程中遇到问题，请：

1. 查看错误日志：`docker-compose logs -f service_name`
2. 检查配置文件：确认环境变量和挂载路径
3. 验证网络连接：确保服务间能正常通信
4. 检查资源使用：确保服务器有足够的CPU和内存

版本: v1.1.0

更新时间: 2025年6月

支持架构: linux/amd64, linux/arm64

技术栈: Spring Boot + Vue.js + Python Flask + MySQL + Redis