



# Low Back Pain Data Collection System

系统架构报告 | System Architecture Report

生成日期: 2025-11-26 (更新)

集成模式: 作为H5服务集成到主系统 | Integration Mode: H5 Service for Main System



## 集成模式说明

### 系统定位

本系统作为**H5服务**集成到主系统，不需要内部工作室管理功能。

### 参数传递

主系统通过URL参数传递以下信息：

- workspace\_id - 工作室ID
- workspace\_name - 工作室名称
- doctor\_id - 医生ID
- doctor\_name - 医生名称
- auth\_token - 认证令牌

### 核心服务

- 患者数据收集表单
- OCR文档识别 (Tesseract)
- 姿态分析 (MediaPipe)



## 系统架构图



### FRONTEND LAYER (前端层)



#### React Frontend

- URL:  
<https://low-back-pain-system.onrender.com>
- 部署平台: Render (Static Site)
- 构建工具: Vite
- 认证状态: 临时禁用 (VITE\_DISABLE\_AUTH=true)
- 集成方式: 通过URL参数接收主系统信息

React 18

Vite

Shadcn/ui

Tailwind CSS

✓ Live

## ⚙️ BACKEND LAYER (后端层)



### .NET Backend API

- URL:  
<https://low-back-pain-backend.onrender.com>
- Swagger API:  
</swagger>
- 部署平台: Render (Web Service - Docker)
- 框架版本: ASP.NET Core 7.0 + ABP vNext 7.3

ASP.NET Core 7.0

ABP vNext 7.3

Entity Framework

✓ Live

## 💾 DATABASE LAYER (数据库层)



### PostgreSQL Database

- 部署平台: Render (PostgreSQL Service)
- 数据库名: LowBackPainDB
- 用途: 患者数据、医疗记录
- 备份: 自动备份 (Render)

PostgreSQL 15

Relational DB

✓ Live

## PYTHON SERVICES LAYER (Python服务层)



### OCR Service

- URL:  
<https://lowbackpain-ocr-service.onrender.com>
- 健康检查:  
</health>
- OCR引擎: Tesseract 5.5.0
- 支持语言: 简体中文、繁体中文、英文

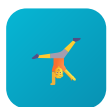
Python 3.10

Flask

Tesseract OCR

OpenCV

✓ Live



### MediaPipe Service

- URL:  
<https://lowbackpain-mediapipe-service.onrender.com>
- 健康检查:  
</health>
- 模型: MediaPipe Pose (Complexity 2)
- 关键点: 33个人体姿态关键点

Python 3.10

Flask

MediaPipe 0.10.18

OpenCV

✓ Live



# GitHub仓库结构



[github.com/HarwuSherlock/low-](https://github.com/HarwuSherlock/low-back-pain-system)

[back-pain-system](#)

## 主要目录说明

目录/文件	类型	说明
<b>backend-dotnet/</b>	后端	ASP.NET Core 7.0 + ABP vNext 7.3 后端服务，包含RESTful API和PostgreSQL数据访问
<b>python-services/</b>	Python 服务	OCR服务(Tesseract)和MediaPipe姿态分析服务
<b>src/</b>	前端	React + Vite前端应用源代码，包含页面、组件和上下文
<b>docs/</b>	文档	系统文档和生成的报告 (architecture_report.html等)
<b>scripts/</b>	脚本	数据生成、测试和PDF生成脚本
<b>_archive/</b>	归档	历史版本代码和测试数据归档
<b>.specify/</b>	文档	项目规格文档(19个MD文件)



[查看完整文件树结构](#)



## 组件汇总表

组件	类型	平台	状态	URL
React Frontend	前端应用	Render Static Site	运行中	<a href="#">链接</a>
.NET Backend	RESTful API	Render Web Service	运行中	<a href="#">链接</a>
PostgreSQL	关系型数据库	Render PostgreSQL	运行中	内部连接
OCR Service	Python服务	Render Web Service	运行中	<a href="#">链接</a>
MediaPipe Service	Python服务	Render Web Service	运行中	<a href="#">链接</a>

# 技术栈总览

---

## 前端技术

React 18

Vite

React Router

Tailwind CSS

Shadcn/ui

Lucide Icons

## 后端技术

ASP.NET Core 7.0

ABP vNext 7.3

Entity Framework

AutoMapper

Swagger/OpenAPI

## AI/ML服务

Python 3.10

Flask

Tesseract OCR

MediaPipe

OpenCV

NumPy

## 基础设施

Docker

PostgreSQL 15

Render Cloud

GitHub

Git



## 部署说明

---

### 当前部署状态

- **部署平台:** Render (免费层 Free Tier)
- **容器化:** 后端和Python服务使用Docker部署
- **自动部署:** 通过GitHub自动触发部署
- **限制:** 512MB RAM, 无GPU, 闲置15分钟后休眠
- **优化:** 使用Tesseract替代PaddleOCR (CPU兼容), MediaPipe使用轻量模型
- **认证:** 临时禁用 (VITE\_DISABLE\_AUTH=true) 用于测试
- **集成方式:** 接收主系统URL参数 (workspace\_id, doctor\_id等)





# 关于Render平台及后续迁移方案

---

## Render平台介绍

- **平台定位:** 现代化云部署平台，类似Heroku，提供简化的部署流程
- **选择原因:** 快速部署、免费层可用、自动CI/CD、原生支持Docker、GitHub集成
- **当前限制:** 免费层配置(512MB RAM)、闲置15分钟后休眠、无GPU支持
- **适用场景:** 开发测试环境、原型验证、小规模应用

## 迁移到生产环境的方案

### 方案1: 迁移到阿里云

- **部署方式:** 阿里云ECS + Docker容器部署
- **数据库:** PostgreSQL迁移到阿里云RDS (托管数据库服务)
- **优势:** 国内访问速度快、稳定性高、可配置更高规格、可选GPU实例(如需要)
- **成本估算:** 基础配置约¥200-500/月 (2核4G ECS + RDS基础版)

### 方案2: 集成到现有PMS系统

- **前端集成:** H5界面通过WebView嵌入到PMS系统
- **后端整合:** .NET后端API可合并到PMS后端服务
- **数据库:** PostgreSQL可合并到PMS数据库或独立部署
- **Python服务:** OCR和MediaPipe服务保持独立部署(推荐阿里云函数计算)
- **优势:** 统一管理、减少维护成本、共享用户认证体系

#### ✦ 推荐迁移路径

- **短期 (1-2周):** 保持Render部署，用于开发测试和功能验证
- **中期 (1-2月):** 迁移到阿里云进行生产环境部署和性能测试
- **长期 (3-6月):** 根据实际使用情况决定是否完全集成到PMS系统

## Low Back Pain Data Collection System

Department of Orthopaedics & Traumatology

The University of Hong Kong

Generated: 2025-11-26 | Architecture Report v2.0 (Updated)