

# 配方排班管理系統 - 管理員指南

## 目錄

- [1. 系統架構](#)
- [2. 安裝和部署](#)
- [3. 用戶管理](#)
- [4. 配方管理](#)
- [5. 排班管理](#)
- [6. 系統維護](#)
- [7. 安全設置](#)
- [8. 故障排除](#)

## 系統架構

### 技術棧

後端 - PHP 8.1+ - MySQL 8.0+ - JWT 認證 - RESTful API

前端 - React Native - Expo Go - Redux 狀態管理 - React Navigation

部署 - Nginx 反向代理 - PHP-FPM - Ubuntu 22.04 LTS

### 系統組件



# 安裝和部署

---

## 系統要求

**服務器要求** - Ubuntu 22.04 LTS 或更新版本 - 2GB+ RAM - 20GB+ 存儲空間 - 網絡連接

**軟件要求** - PHP 8.1+ - MySQL 8.0+ - Nginx - Node.js 18+ (開發環境)

## 自動部署

使用提供的部署腳本進行一鍵部署：

```
sudo ./deploy.sh
```

部署腳本將自動： 1. 安裝必要的軟件包 2. 配置 MySQL 數據庫 3. 設置 Nginx 反向代理 4. 初始化數據庫結構 5. 配置防火牆規則

## 手動部署

如需手動部署，請參考以下步驟：

### 1. 安裝軟件包

```
sudo apt update
sudo apt install -y nginx php-fpm php-mysql mysql-server
```

### 2. 配置 MySQL

```
sudo mysql_secure_installation
sudo mysql -e "CREATE DATABASE recipe_schedule_db;"
sudo mysql -e "CREATE USER 'recipe_app'@'localhost' IDENTIFIED BY
'your_password';"
sudo mysql -e "GRANT ALL PRIVILEGES ON recipe_schedule_db.* TO
'recipe_app'@'localhost';"
```

### 3. 導入數據庫結構

```
mysql -u recipe_app -p recipe_schedule_db < backend/config/init_database.sql
```

## 4. 配置 Nginx

創建站點配置文件 `/etc/nginx/sites-available/recipe_schedule_app`：

```
server {
    listen 80;
    server_name your_domain.com;
    root /var/www/recipe_schedule_app/backend/api;
    index index.php;

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location ~ \.php$ {
        fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        include fastcgi_params;
    }

    # CORS 設置
    add_header Access-Control-Allow-Origin *;
    add_header Access-Control-Allow-Methods "GET, POST, PUT, DELETE, OPTIONS";
    add_header Access-Control-Allow-Headers "Content-Type, Authorization";
}
```

## 5. 啟用站點

```
sudo ln -s /etc/nginx/sites-available/recipe_schedule_app /etc/nginx/sites-enabled/
sudo systemctl restart nginx
sudo systemctl restart php8.1-fpm
```

# 用戶管理

---

## 默認用戶

系統初始化後包含以下默認用戶：

用戶名	密碼	角色	說明
admin	password123	super_admin	超級管理員
chef_wang	password123	chef	主廚
manager_li	password123	admin	管理員
staff_chen	password123	staff	員工

**重要:** 部署後請立即更改所有默認密碼！

### 用戶角色權限

功能	員工	主廚	管理員	超級管理員
查看基礎配方	✓	✓	✓	✓
查看高級配方	✗	✓	✓	✓
創建配方	✗	✓	✓	✓
查看個人排班	✓	✓	✓	✓
查看所有排班	✗	✓	✓	✓
創建排班	✗	✗	✓	✓
用戶管理	✗	✗	✗	✓

### 創建新用戶

通過 API 創建新用戶：

```
curl -X POST http://your-server/api/users \
-H "Content-Type: application/json" \
-H "Authorization: Bearer YOUR_ADMIN_TOKEN" \
-d '{
  "username": "new_user",
  "password": "secure_password",
  "email": "user@example.com",
  "full_name": "新用戶",
  "role": "staff"
}'
```

# 配方管理

---

## 配方分類

系統預設以下配方分類： - 前菜 - 主菜 - 甜點 - 湯品 - 飲品 - 醬料

## 創建配方

通過 API 創建新配方：

```
curl -X POST http://your-server/api/recipes \
-H "Content-Type: application/json" \
-H "Authorization: Bearer YOUR_TOKEN" \
-d '{
  "name": "新配方",
  "description": "配方描述",
  "category_id": 2,
  "difficulty_level": "medium",
  "prep_time": 30,
  "cook_time": 60,
  "servings": 4,
  "min_role": "staff",
  "ingredients": [
    {
      "name": "食材1",
      "quantity": "100",
      "unit": "克"
    }
  ],
  "steps": [
    {
      "instruction": "製作步驟1"
    }
  ]
}'
```

## 配方權限設置

配方的 `min_role` 字段控制最低訪問權限： - `staff`：所有用戶可見 - `chef`：主廚及以上可見 - `admin`：管理員及以上可見 - `super_admin`：僅超級管理員可見

# 排班管理

---

## 班次類型

- `morning` : 早班 (通常 06:00-14:00)
- `afternoon` : 午班 (通常 14:00-22:00)
- `evening` : 晚班 (通常 18:00-02:00)

## 排班狀態

- `scheduled` : 已安排
- `confirmed` : 已確認
- `completed` : 已完成
- `absent` : 缺勤

## 創建排班

```
curl -X POST http://your-server/api/schedules \
-H "Content-Type: application/json" \
-H "Authorization: Bearer YOUR_ADMIN_TOKEN" \
-d '{
  "user_id": 2,
  "shift_date": "2025-07-01",
  "shift_type": "morning",
  "start_time": "06:00:00",
  "end_time": "14:00:00",
  "status": "scheduled"
}'
```

## 排班統計

管理員可以查看排班統計：

```
curl -X GET "http://your-server/api/schedules/stats?start_date=2025-07-01&end_date=2025-07-31" \
-H "Authorization: Bearer YOUR_ADMIN_TOKEN"
```

# 系統維護

---

## 日誌管理

系統日誌位置： - Nginx 日誌： `/var/log/nginx/` - PHP 日誌： `/var/log/php8.1-fpm.log` - 應用日誌： `/var/log/recipe_schedule_app/`

## 數據庫備份

定期備份數據庫：

```
# 創建備份
mysqldump -u recipe_app -p recipe_schedule_db > backup_$(date +%Y%m%d_%H%M%S).sql

# 恢復備份
mysql -u recipe_app -p recipe_schedule_db < backup_file.sql
```

## 性能監控

使用提供的性能測試腳本監控系統性能：

```
node performance_test.js
```

## 系統更新

1. 備份數據庫和配置文件
2. 停止服務
3. 更新代碼
4. 運行數據庫遷移（如需要）
5. 重啟服務
6. 驗證功能

# 安全設置

---

## JWT 密鑰

更改 JWT 密鑰以增強安全性：

1. 編輯 `backend/utils/JWT.php`
2. 修改 `$secret_key` 變量
3. 重啟服務

## 數據庫安全

1. 使用強密碼
2. 限制數據庫用戶權限
3. 定期更新密碼
4. 啟用 SSL 連接（生產環境）

## 防火牆設置

```
# 允許 HTTP/HTTPS
sudo ufw allow 'Nginx Full'

# 允許 SSH
sudo ufw allow ssh

# 啟用防火牆
sudo ufw enable
```

## HTTPS 配置

生產環境建議使用 HTTPS：

1. 獲取 SSL 證書（Let's Encrypt）
2. 配置 Nginx SSL
3. 強制 HTTPS 重定向



# 故障排除

---

## 常見問題

### API 無法訪問

1. 檢查 Nginx 狀態： `sudo systemctl status nginx`
2. 檢查 PHP-FPM 狀態： `sudo systemctl status php8.1-fpm`
3. 查看錯誤日誌： `sudo tail -f /var/log/nginx/error.log`

### 數據庫連接失敗

1. 檢查 MySQL 狀態： `sudo systemctl status mysql`
2. 驗證數據庫配置： `backend/config/database.php`
3. 測試數據庫連接： `mysql -u recipe_app -p`

### 移動應用無法連接

1. 確認 API 服務器可訪問
2. 檢查防火牆設置
3. 驗證 CORS 配置
4. 檢查網絡連接

## 調試模式

啟用 PHP 錯誤顯示（僅開發環境）：

```
// 在 API 文件頂部添加
ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);
```

## 性能優化

1. 啟用 PHP OPcache
2. 配置 MySQL 查詢緩存

3. 使用 Nginx 靜態文件緩存
4. 優化數據庫索引

## 聯繫支持

---

如需技術支持，請提供以下信息： - 系統版本 - 錯誤日誌 - 重現步驟 - 服務器環境信息

---

本指南版本：1.0

最後更新：2025年6月30日