

CA392-ForestSmart - Intelligent Forestry Management System

森智管 - 智能山林管理系统

目录

CA392-ForestSmart - Intelligent Forestry Management System	- 1 -
森智管 - 智能山林管理系统	- 1 -
1. 项目简介.....	- 3 -
2. 系统功能模块.....	- 3 -
2.1 账户与权限管理.....	- 3 -
2.2 林地资源管理.....	- 3 -
2.3 植被与生态监测.....	- 4 -
2.4 养护与巡护管理.....	- 4 -
2.5 采伐与林业生产.....	- 4 -
2.6 财务与收益管理.....	- 5 -
2.7 设备与物资管理.....	- 5 -
2.8 报表与数据分析.....	- 5 -
3. 技术特点.....	- 6 -
4. 预期收益.....	- 6 -
5.PHP 程序文件架构.....	- 7 -
6.MYSQL8 数据库设计.....	- 11 -

森智管 - 智能山林管理系统

1. 项目简介

CA392ForestSmart（森智管 智能山林管理系统）是一款专为山林管理打造的信息化平台，结合 GIS（地理信息系统）、IoT（物联网）、智能数据分析等技术，提供森林资源的全面监控与管理。适用于国有林场、私人林地经营者、政府林业部门、环保组织等，帮助用户优化资源利用，提升森林经营效率，确保生态环境可持续发展。

2. 系统功能模块

2.1 账户与权限管理

用户管理：支持管理员、普通用户、巡护员等不同角色的账号管理。

权限控制：基于角色的权限分配，确保数据安全与访问控制。

日志审计：记录所有用户操作，提供日志查询功能。

2.2 林地资源管理

林地档案：管理林区基本信息（如面积、地形、树种分布、土壤状

况等)。

地图管理：GIS 电子地图展示林区情况，支持卫星图、地形图切换。

土地流转管理：记录林地承包、流转合同、租赁到期提醒等信息。

2.3 植被与生态监测

树种管理：登记林区内树种信息，包括生长情况、预计采伐周期、病虫害历史等。

生态监测：

气象数据（温度、湿度、降水量等）

土壤监测（pH 值、养分含量）

空气质量监测

病虫害管理：记录病虫害事件，提供防治方案推荐。

2.4 养护与巡护管理

巡护计划：设定巡护任务，自动提醒巡护员。

巡护记录：支持移动端数据采集，上传巡护照片和日志。

智能预警：对异常情况（如非法砍伐、火灾风险）进行预警。

2.5 采伐与林业生产

采伐申请管理：支持申请、审批流程，确保合法采伐。

木材产量统计：记录采伐数据，计算产量、经济价值。

加工与销售：

记录木材加工信息（如规格、用途）

订单管理（销售、发货）

2.6 财务与收益管理

成本核算：记录土地租赁、人工成本、设备维护费用。

收益管理：统计木材销售、补贴收入、碳汇交易等收益。

报表分析：生成财务报表，支持导出 Excel/PDF。

2.7 设备与物资管理

设备管理：登记无人机、摄像头、传感器等设备信息，监控状态与维护情况。

物资库存：管理种苗、肥料、农药等库存，支持自动补货提醒。

2.8 报表与数据分析

统计分析：

资源利用率分析

采伐 vs. 补植比率

收益/成本比对

可视化报表：提供折线图、柱状图、饼图等多种数据展示方式。

3. 技术特点

GIS 地理信息集成：支持地图标注林地位置，展示资源分布情况。

IoT 物联网监测：对气象、土壤、空气质量等数据实时采集。

智能预警系统：结合 AI 数据分析，识别异常情况并推送警报。

多端支持：可通过 Web 端、移动端（H5）、PC 端 访问。

4. 预期收益

提高管理效率：信息化管理减少人工记录的繁琐，提高林场管理效率。

降低运营成本：通过智能化设备监控，减少不必要的人工巡查。

提升资源利用率：优化采伐与种植计划，确保林业可持续发展。

生态保护：精准监控森林状况，及时发现病虫害、火灾等风险。

5. PHP 程序文件架构

/CA392-ForestSmart

```
| — /app
|   | — /controllers
|   |   | — AuthController.php           # 认证控制器（登
录、注册）
|   |   | — DashboardController.php     # 仪表盘控制器
|   |   | — ForestController.php       # 林地资源管理
|   |   | — PatrolController.php       # 巡护管理
|   |   | — EcologyController.php      # 生态监测
|   |   | — LoggingController.php      # 采伐管理
|   |   | — FinanceController.php      # 财务管理
|   |   | — EquipmentController.php    # 设备管理
|   |   | — ReportController.php       # 数据报表
|   |   | — UserController.php        # 用户管理
|   |   | — RoleController.php         # 角色权限管理
|   | — /models
|   |   | — User.php                   # 用户模型
|   |   | — Role.php                   # 角色权限模型
```

				—— Forest.php	# 林地资源模型
				—— Patrol.php	# 巡护记录模型
				—— Ecology.php	# 生态监测模
型					
				—— Logging.php	# 采伐管理模
型					
				—— Finance.php	# 财务管理模
型					
				—— Equipment.php	# 设备管理模
型					
				—— Report.php	# 报表数据模
型					
			——	/views	
				—— /auth	
				—— login.php	# 登录界面
				—— register.php	# 注册界面
				—— /dashboard	
				—— index.php	# 仪表盘
				—— /forest	
				—— index.php	# 林地资源列
表					

				—— edit.php	# 编辑林地信
息					
				—— create.php	# 添加新林地
				—— /patrol	
				—— index.php	# 巡护记录列
表					
				—— detail.php	# 巡护详情
				—— /ecology	
				—— index.php	# 生态监测数
据					
				—— charts.php	# 数据图表展
示					
				—— /logging	
				—— index.php	# 采伐管理
				—— /finance	
				—— index.php	# 财务管理
				—— /equipment	
				—— index.php	# 设备管理
				—— /report	
				—— index.php	# 统计报表
				—— /config	

			—— config.php	# 数据库配置
			—— routes.php	# 路由配置
		—— /core		
			—— Database.php	# 数据库连接
			—— Model.php	# 基础模型
类				
			—— Controller.php	# 基础控制器
类				
		—— /helpers		
			—— functions.php	# 辅助函数
	—— /public			
		—— /assets		
			—— /css	
			—— /js	
			—— /images	
		—— index.php		# 入口文件
	—— /storage			
		—— /logs		
		—— /uploads		
	—— .htaccess			
	—— composer.json			

| — README.md

6. MYSQL8 数据库设计

CA392ForestSmart MySQL 8 数据库架构设计

CA392ForestSmart 采用 MySQL 8 作为数据库管理系统，所有数据表按照 规范化设计，确保数据完整性、安全性、查询效率，并支持未来的扩展性。

1. 数据库名称

```
```sql
```

```
CREATE DATABASE CA392_ForestSmart DEFAULT CHARACTER SET
utf8mb4 COLLATE utf8mb4_unicode_ci;
```

```
```
```

2. 数据表设计

2.1 用户与权限管理

用户表（users）

```
```sql
```

```
CREATE TABLE users (
 id INT PRIMARY KEY AUTO_INCREMENT,
 username VARCHAR(50) UNIQUE NOT NULL,
 password_hash VARCHAR(255) NOT NULL,
 email VARCHAR(100) UNIQUE NOT NULL,
 phone VARCHAR(20),
 role_id INT NOT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
 FOREIGN KEY (role_id) REFERENCES roles(id) ON DELETE CASCADE
);
```
```

角色表（roles）

```
```sql
```

```
CREATE TABLE roles (
 id INT PRIMARY KEY AUTO_INCREMENT,
 name VARCHAR(50) UNIQUE NOT NULL,
 description VARCHAR(255) NOT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
 FOREIGN KEY (parent_role_id) REFERENCES roles(id) ON DELETE CASCADE
);
```
```

```
id INT PRIMARY KEY AUTO_INCREMENT,  
name VARCHAR(50) UNIQUE NOT NULL,  
permissions JSON NOT NULL COMMENT '存储权限配置 (JSON 格式) '  
);  
...
```

2.2 林地资源管理

林地信息表 (forests)

```
```sql
```

```
CREATE TABLE forests (
 id INT PRIMARY KEY AUTO_INCREMENT,
 name VARCHAR(100) NOT NULL,
 area DECIMAL(10,2) NOT NULL COMMENT '林地面积 (亩) ',
 location POINT NOT NULL COMMENT 'GIS 坐标点',
 owner_id INT NOT NULL COMMENT '所有者 (用户 ID) ',
 tree_species TEXT COMMENT '主要树种信息',
 soil_type VARCHAR(50) COMMENT '土壤类型',
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
```

```
updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
FOREIGN KEY (owner_id) REFERENCES users(id) ON DELETE
CASCADE
);
...
```

## 2.3 生态监测

生态监测数据表 (ecology\_data)

```
```sql
```

```
CREATE TABLE ecology_data (
    id INT PRIMARY KEY AUTO_INCREMENT,
    forest_id INT NOT NULL,
    temperature DECIMAL(5,2) NOT NULL COMMENT '温度 (°C)',
    humidity DECIMAL(5,2) NOT NULL COMMENT '湿度 (%)',
    soil_ph DECIMAL(3,2) NOT NULL COMMENT '土壤 pH 值',
    air_quality_index INT COMMENT '空气质量指数',
    recorded_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
COMMENT '数据记录时间',
```

```
FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE  
CASCADE  
);  
...
```

2.4 巡护管理

巡护记录表 (patrols)

```
```sql
```

```
CREATE TABLE patrols (
 id INT PRIMARY KEY AUTO_INCREMENT,
 patrol_user_id INT NOT NULL COMMENT '巡护员 ID',
 forest_id INT NOT NULL COMMENT '林地 ID',
 patrol_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
COMMENT '巡护时间',
 notes TEXT COMMENT '巡护情况',
 image_url VARCHAR(255) COMMENT '巡护图片',
 FOREIGN KEY (patrol_user_id) REFERENCES users(id) ON DELETE
CASCADE,
 FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE
```

---

CASCADE

);

...

## 2.5 采伐管理

采伐申请表 (logging\_requests)

```sql

```
CREATE TABLE logging_requests (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    forest_id INT NOT NULL,  
    request_user_id INT NOT NULL COMMENT '申请人 ID',  
    tree_species VARCHAR(100) NOT NULL COMMENT '采伐树种',  
    amount INT NOT NULL COMMENT '数量 (棵)',  
    status ENUM('pending', 'approved', 'rejected') DEFAULT 'pending'  
COMMENT '审批状态',  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at  TIMESTAMP  DEFAULT  CURRENT_TIMESTAMP  ON  
UPDATE CURRENT_TIMESTAMP,  
    FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE
```



```
CASCADE,  
    FOREIGN KEY (request_user_id) REFERENCES users(id) ON DELETE  
CASCADE  
);  
...
```

2.6 财务与收益管理

财务记录表 (finance_records)

```
```sql  
CREATE TABLE finance_records (
 id INT PRIMARY KEY AUTO_INCREMENT,
 forest_id INT NOT NULL,
 record_type ENUM('income', 'expense') NOT NULL COMMENT '收入
/支出',
 amount DECIMAL(10,2) NOT NULL COMMENT '金额',
 description TEXT COMMENT '备注',
 recorded_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE
CASCADE
```

---

);

...

## 2.7 设备与物资管理

设备管理表 (equipments)

```sql

```
CREATE TABLE equipments (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(100) NOT NULL COMMENT '设备名称',  
    type VARCHAR(50) NOT NULL COMMENT '设备类型',  
    status ENUM('working', 'repair', 'retired') DEFAULT 'working'  
COMMENT '状态',  
    forest_id INT NOT NULL COMMENT '所属林地',  
    purchase_date DATE COMMENT '购入日期',  
    maintenance_date DATE COMMENT '最近维护日期',  
    FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE  
CASCADE  
);  
...
```

2.8 数据报表

统计数据表（reports）

```
```sql
```

```
CREATE TABLE reports (
 id INT PRIMARY KEY AUTO_INCREMENT,
 forest_id INT NOT NULL,
 report_type ENUM('finance', 'ecology', 'logging', 'patrol') NOT NULL
COMMENT '报表类型',
 data JSON NOT NULL COMMENT '存储统计数据（JSON 格式）',
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 FOREIGN KEY (forest_id) REFERENCES forests(id) ON DELETE
CASCADE
);
...
```

## 3. 数据索引优化

为了提高查询效率，我们对部分表字段添加索引：

```
```sql
```

```
ALTER TABLE users ADD INDEX idx_role_id (role_id);
```

```
ALTER TABLE forests ADD SPATIAL INDEX idx_location (location);
```

```
ALTER TABLE ecology_data ADD INDEX idx_forest_id (forest_id);
```

```
ALTER TABLE patrols ADD INDEX idx_patrol_user_id (patrol_user_id);
```

```
ALTER TABLE finance_records ADD INDEX idx_forest_id (forest_id);
```

```
ALTER TABLE equipments ADD INDEX idx_forest_id (forest_id);
```

```
```
```

#### 4. 未来扩展

增加物联网数据表（接入传感器实时数据）

优化权限管理（细粒度权限控制）

增加 API 支持（提供 RESTful API 以支持移动端）

#### 总结

数据表结构清晰，严格遵循规范化设计

引入 GIS 坐标支持，更精准管理森林位置

所有业务数据均可关联，提高查询效率

采用 JSON 存储部分数据（如权限、统计报表），方便扩展

索引优化，保证查询速度

这个数据库架构可以支撑 CA392ForestSmart 的完整功能，你觉得是否还需要补充或修改？

Powered by Xander Ray , caffz.com