**开发的硬件环境（50字内）**

Intel Core i7-12700K处理器(12核心)、32GB DDR5 RAM、1TB NVMe SSD、NVIDIA GeForce RTX 3080Ti显卡(12GB VRAM)、双4K显示器、2.5Gbps网络环境、高性能散热系统。

**运行的硬件环境（50字内）**

服务器：双路Intel Xeon Gold 6346处理器(16核32线程)、128GB ECC RAM、4TB NVMe SSD RAID 10阵列、10Gbps网络；客户端：Intel Core i5-10400以上、16GB RAM、集成显卡、千兆网络连接。

**开发该软件的操作系统（50字内）**

主要开发环境：Windows 11 Pro 22H2(64位)、Ubuntu 22.04 LTS服务器版；辅助开发环境：macOS Ventura 13.4、CentOS 8.5，采用虚拟化技术确保跨平台兼容性测试。

**软件开发环境/开发工具（50字内）**

Visual Studio Code 1.78、IntelliJ IDEA 2023.1 Ultimate、Git 2.40、npm 9.6、Webpack 5.82、Docker 23.0、Kubernetes 1.27、Jenkins 2.401、Figma设计工具、Postman 10.14、Chrome DevTools。

**该软件的运行平台/操作系统（50字内）**

服务端：Ubuntu 22.04 LTS、Rocky Linux 9.1、Windows Server 2022；客户端：支持Chrome 112+、Firefox 113+、Edge 112+、Safari 16.4+浏览器，兼容Windows、macOS、Android 10+、iOS 15+系统。

**软件运行支撑环境/支持软件（50字内）**

Node.js 18.16 LTS、JDK 17.0.7、PostgreSQL 15.3(含PostGIS扩展)、Nginx 1.24、Redis 7.0.11、RabbitMQ 3.12、Docker 23.0、Docker Compose 2.17、Elasticsearch 8.8、Kibana 8.8。

**用到的编程语言有哪些（120字内）**

前端：TypeScript 5.0(主要语言)、JavaScript ES2022、HTML5、CSS3、SCSS；

后端：Java 17(核心业务逻辑)、SQL(数据操作)；辅助语言：Python 3.10(数据处理脚本)、Shell脚本(自动化部署)；配置语言：YAML、JSON、XML；前端采用TypeScript实现类型安全，React框架构建组件化界面，后端使用Java Spring Boot框架开发微服务，PostgreSQL存储关系数据和地理信息数据，Python用于数据分析和处理。

**源程序量**

前端代码：28,500行（包含React组件、工具函数、样式文件）后端代码：23,700行（包含业务逻辑、数据访问层、API接口）配置文件：4,800行（包含项目配置、部署脚本、数据库脚本）总计：57,000行

**开发目的（50字内）**

构建智能化森林防火监控平台，整合多源监测数据，实现火情早期发现、精准预警、高效处置，提升森林防火科学化、信息化、智能化水平，有效保护森林资源和生态安全。

**面向领域/行业（50字内）**

国家林业和草原局、省市县级林业管理部门、森林防火指挥中心、自然保护区管理机构、国家公园管理部门、生态环境监测站、应急管理部门、森林消防队伍、林区管理单位。

**软件的主要功能（200字内）**

1. 多源监测数据融合：整合卫星遥感、无人机航拍、地面监测站、红外热成像等多源数据，构建全方位监测网络。
2. 分级预警系统：基于气象数据、植被状况、历史火情，建立多级火险预警机制，支持自定义预警规则和阈值。
3. 精细化巡护管理：科学规划巡护路线，智能分配任务，GPS实时定位跟踪，记录巡护发现，评估巡护效果。
4. 告警闭环处理：支持告警分级分类，规范处理流程，全程记录处理过程，提供完整追溯。
5. 资源智能调度：管理防火人员、车辆、设备、物资，支持就近调度，优化资源配置。
6. 地理信息可视化：高精度GIS地图展示，标注火点位置、蔓延趋势、防火设施、水源点、交通网络。
7. 气象信息分析：接入气象数据，分析温度、湿度、风向、风速，评估火灾风险，预测火势蔓延。
8. 多维数据分析：提供火情趋势分析、区域风险评估、资源利用分析，生成专业报表，辅助科学决策。
9. 移动协同作业：支持移动终端访问，实现内外业协同，提高应急响应速度。

**软件的技术特点（100字）**

1. 前后端分离架构：采用React+TypeScript构建SPA应用，实现高性能交互体验；后端基于Spring Boot微服务框架，提供RESTful API。
2. 地理信息处理：集成PostGIS空间数据库和OpenLayers地图引擎，实现精准地理定位和空间分析。

3. 实时数据处理：采用WebSocket技术实现监测数据实时推送，Redis缓存提升数据访问性能。

1. 多级安全保障：实现基于JWT的身份认证，细粒度RBAC权限控制，数据传输加密，确保系统安全可靠。
2. 高可用部署：采用Docker容器化部署，支持水平扩展和故障自动恢复。