

AI智慧农场管理系统的设计与实现

郭文君, 颜丽, 张鑫岚, 钟应招

萍乡学院, 信息与计算机工程学院, 江西 萍乡

收稿日期: 2023年12月22日; 录用日期: 2024年1月19日; 发布日期: 2024年1月30日

摘要

随着经济和现代技术的快速发展, 现代农业也开始逐步向智能化方向发展。传统的生产方式, 对环境参数的获取和控制都无法做到及时性, 且无法对农作物生长状况、生长阶段等作出系统的记录与判断。结合现阶段的功能需求和功能需求, 设计智慧农业环境监测及信息管理系统, 将有助于实现农业现代化管理, 推动我国乡村振兴。智慧农业环境监测及信息管理系统是一个集成人工智能技术、传感器技术、大数据分析等多种技术手段的综合性农业生产管理系统, 旨在帮助农民提高生产效率, 降低成本, 提高农产品的质量和产量。该系统可以对农场进行全面的的管理, 包括土地利用、作物生长管理、农产品质量监测、天气预报、设备管理、农情通知、农资管理等方面。

关键词

智慧农业, 传感器, 阿里云, SpringBoot, Vue

Design and Implementation of AI Intelligent Farm System

Wenjun Guo, Li Yan, Xinlan Zhang, Yingzhao Zhong

School of Information and Computer Engineering, Pingxiang University, Pingxiang Jiangxi

Received: Dec. 22nd, 2023; accepted: Jan. 19th, 2024; published: Jan. 30th, 2024

Abstract

With the rapid development of the economy and modern technology, modern agriculture is also gradually moving towards intelligence. Traditional production methods cannot achieve timely acquisition and control of environmental parameters, and cannot make systematic records and judgments on crop growth status and stages. Combining the current situation and functional requirements, designing a smart agricultural environmental monitoring and information manage-

ment system will help achieve modern agricultural management and promote rural revitalization in China. The Smart Agricultural Environmental Monitoring and Information Management System is a comprehensive agricultural production management system that integrates various technological means such as artificial intelligence, sensor technology, and big data analysis. It aims to help farmers improve production efficiency, reduce costs, and improve the quality and yield of agricultural products. This system can comprehensively manage farms, including land use, crop growth management, agricultural product quality monitoring, weather forecasting, equipment management, agricultural information notification, agricultural material management, and other aspects.

Keywords

Smart Agriculture, Sensors, Alibaba Cloud, SpringBoot, Vue

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 问题提出

如今的 AI 智慧农场系统总体已经初步完成，但是很多细节还仍需我们继续修改和完善。AI 智慧农场系统现今所有功能只有七个模块，还需要根据农场的具体需求去增加和修缮，并在原有基础上更改地更加贴合使用者的习惯并且便利使用者[1]。AI 智慧农场系统现在所使用的部分技术和设备都较为基础，在今后资金充足的条件下必将提高技术的严谨性以及设备的精密度[2]。本系统暂时只注重于技术以及功能部分，对其后续的风险维护和商业价值方面还略有欠缺，需要在老师的进一步指导下更加精益求精的去做好这部分的整理。在科技日新月异的背景下，我们更需要跟进时代的洪流，让系统具有独特的优势，保障好系统不被淘汰，才能在未来越来越好！

2. 技术路线

AI 智慧农场系统主要由传感器、摄像头、阿里云平台组成的环境监测模块与信息管理系统平台几部分组成[3]。环境监测模块的实现方式，主要是采用 ESP8266 开发版通过 MQTT 通信协议将各类传感器数据传输给阿里云平台并通过信息管理系统监测与展示数据[4]。本系统的信息管理系统采用前后端分离的开发模式，后端基于 Java 语言开发，采用目前比较流行的 SpringBoot + MyBatis-plus 框架，前端则使用 Vue 框架以及 Element-UI 组件[5]。框架技术设计，对农业生产过程中所涉及的事务进行信息化管理，主要包括环境数据监控大屏、农事管理、设备管理、农情通知、农资管理等模块。在 AI 智慧农业系统的业务设计时，我们需要注重成本，设备的防水，防漏电的问题[6]。而 WiFi 通信则基本满足农业生产的需求，所以本系统数据传输采用的是 WiFi 通信。由于本系统要实现无线远程环境监测，需要将传感器数据传输至阿里云平台，故使用 MQTT 协议实现数据通信，并使用同样的方式将数据从阿里云平台传输至本系统[7]。

3. 系统设计

3.1. 系统功能分析

AI 智慧农场系统在从阿里云平台接收数据，并对数据进行可视化监测与记录。同时使用摄像头实时将种植地的情况远程监控，便于生产人员远程了解农作物生长情况。包括对农作物生长环境管理，通过

传感器获取空气温湿度，土壤温湿度，土壤 PH 值等数据后将数据传输到阿里云平台。也可以对传感器，监控设备等进行管理，包括控制上线与下线，查询设备的状态，设备监测日志，配置新设备等。具体的系统整体架构如图 1 所示。

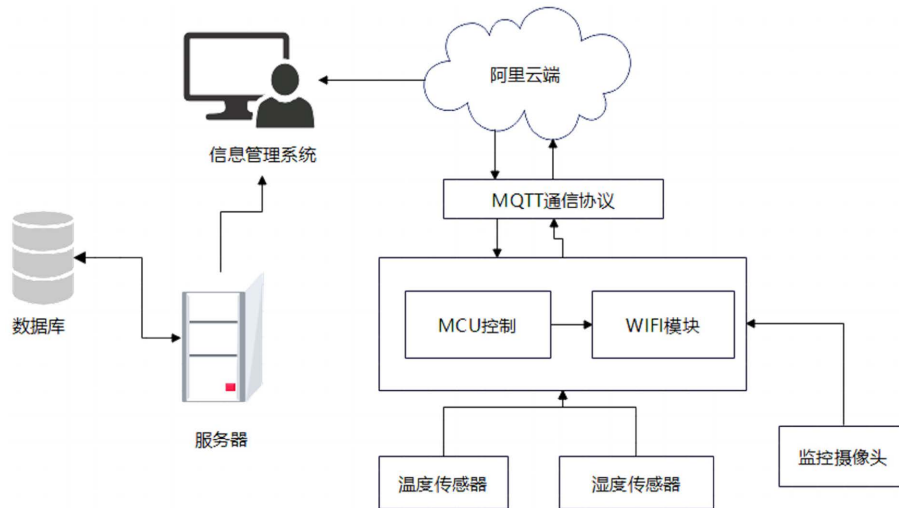


Figure 1. Overall system architecture
图 1. 系统整体架构

3.2. 数据库设计

根据实体设计，可以得到各个实体的 E-R 图。

3.2.1. 管理员实体

显示管理员的属性，E-R 图如图 2 所示。

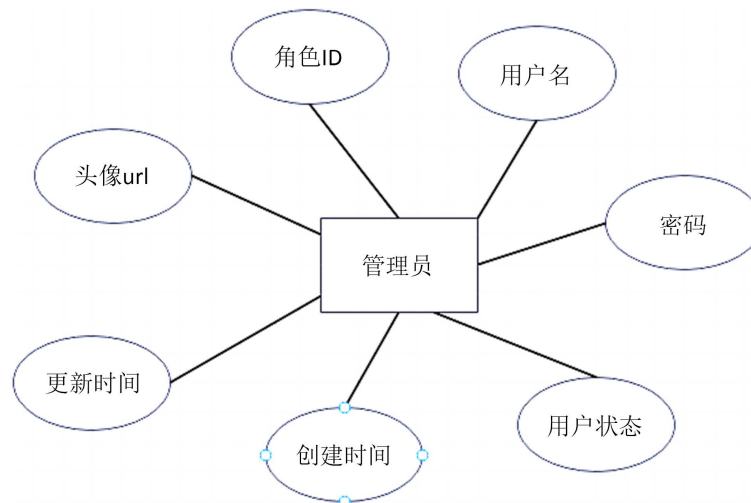


Figure 2. System E-R diagram
图 2. 系统 E-R 图

3.2.2. 农作物实体

显示农作物的属性，E-R 图如图 3 所示。

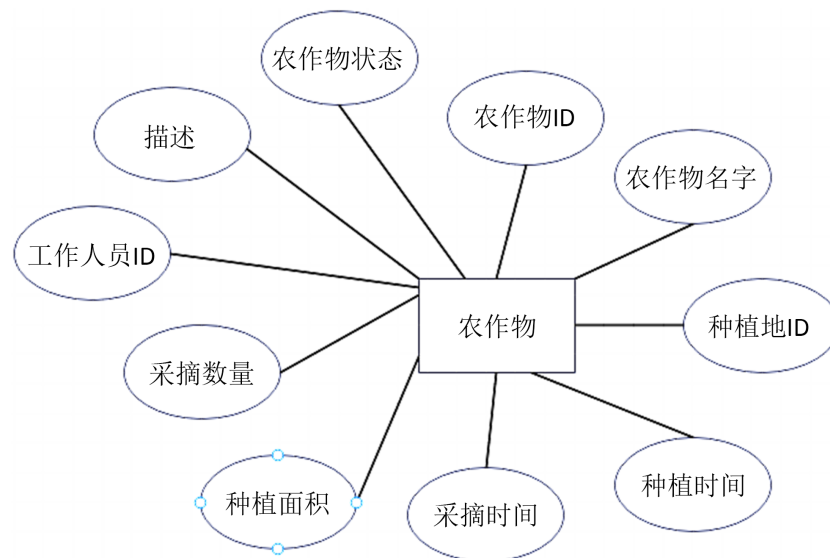


Figure 3. Crop entity E-R diagram

图 3. 农作物实体 E-R 图

3.2.3. 种植地实体

显示种植地的属性，E-R 图如图 4 所示。

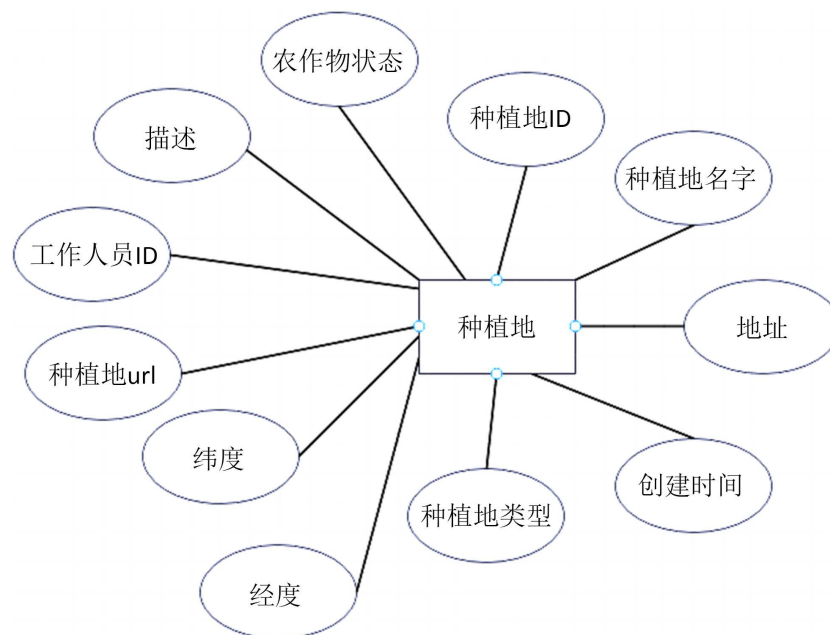


Figure 4. Planting land entity E-R diagram

图 4. 种植地实体 E-R 图

3.2.4. 工作人员实体

显示工作人员的属性，E-R 图如图 5 所示。

3.2.5. 设备实体

显示设备的属性，E-R 图如图 6 所示。

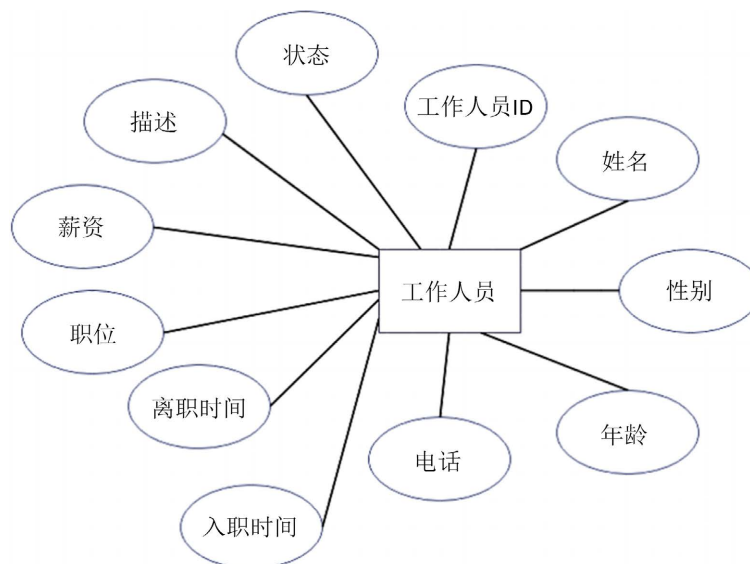


Figure 5. Staff entity E-R diagram

图 5. 工作人员实体 E-R 图

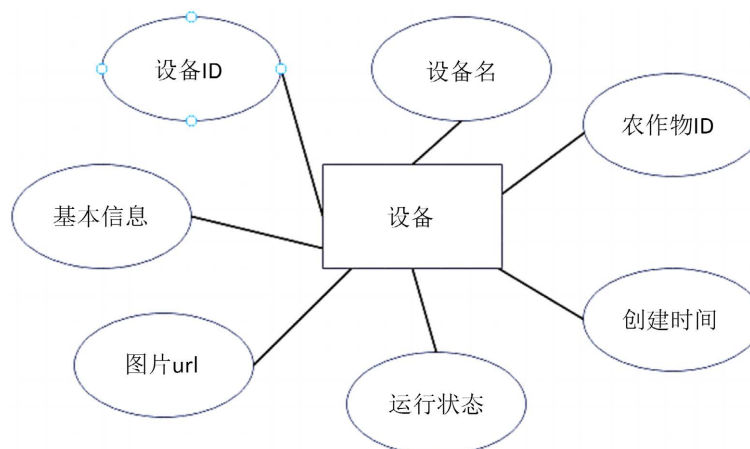


Figure 6. Equipment entity E-R diagram

图 6. 设备实体 E-R 图

4. 系统功能实现

4.1. 系统登录页面设计与实现

在系统的登录页面，输入用户表以后的用户名和密码，登录界面加入了图形验证码登录，以此来防止非法网络攻击带来的服务器资源浪费。每次图形验证码登录时会携带唯一的 key，作为登录凭证，点击登录后就可以将用户名、密码、图形验证码、验证码唯一标识 key 通过表单提交的方式发送给后台，验证成功后才能进入系统，否则提示登录失败。

系统登录页面如图 7 所示。

4.2. 系统首页设计与实现

系统首页是一块可视化监控大屏，可视化监控了室外环境、土壤环境、天气预报、种植详情、设备运行情况、环境数据变化等多个指标。能够一目了然地掌握当前种植地的环境变化和农作物的生长情况。

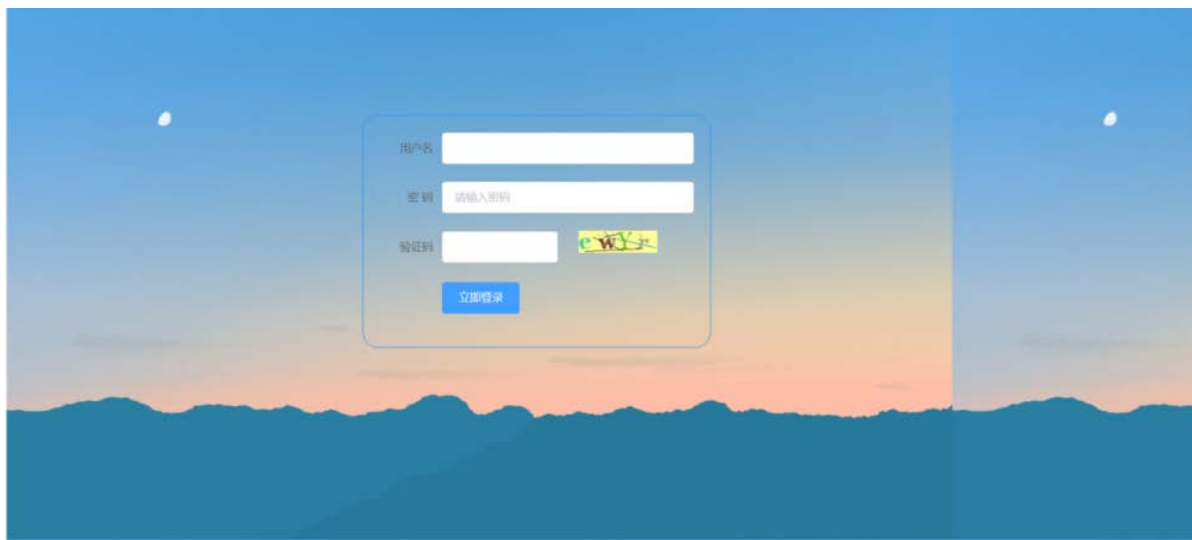


Figure 7. System login page
图 7. 系统登录页面



Figure 8. System homepage
图 8. 系统首页

系统首页如图 8 所示。

为了实现环境数据的远程监测，本系统需要先将传感器数据上传到云平台，系统再与云平台通信。为了使设备、云平台、设备进行通信，首先需要在阿里云物联网平台创建实例，并填写设备信息以创建设备，获取三元组信息。

创建成功后设备信息如图 9 所示。

之后进入设备烧录阶段，将 ESP8266 开发版与 DHT11 温湿度传感器连接，并使用 Arduino 客户端编写程序，将带有三元组、WIFI 账号密码等信息的程序烧录至开发版之后，设备激活，云平台设备显示在线。

设备激活在线如图 10 所示。

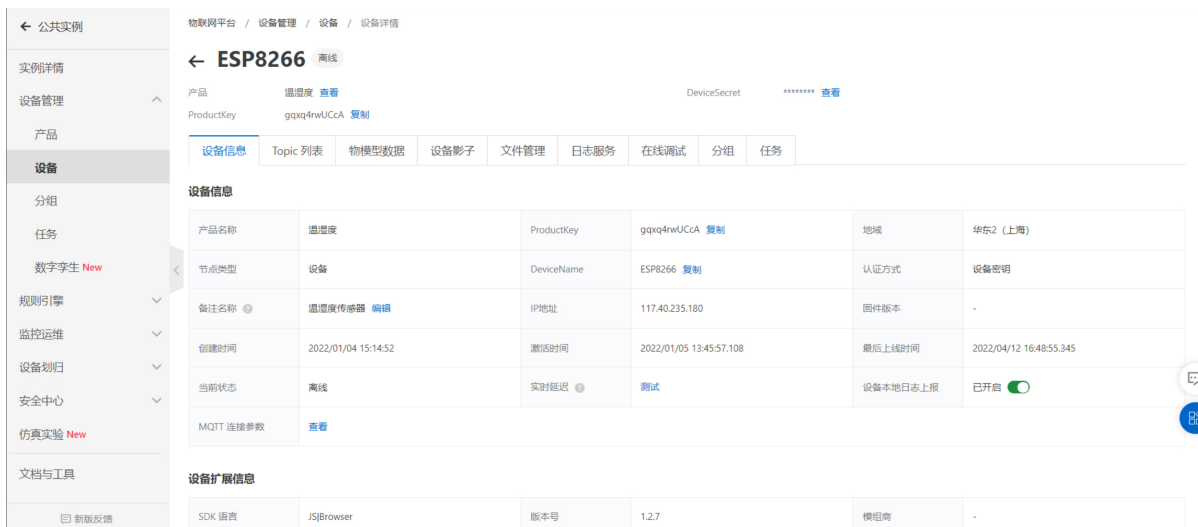


Figure 9. Device information after creation
图 9. 创建后设备信息



Figure 10. Device activation online
图 10. 设备激活在线

至此，阿里云的物联网平台已经和传感器实现了通讯，并且可以对其进行实时采集。物模型数据如图 11 所示。



Figure 11. Object model data
图 11. 物模型数据

设备与云平台完成通信后，需要使用 MQTT 通信协议实现云平台与本系统的双向实时通信，将物模型数据实时传输到本系统。

系统向云平台订阅消息如图 12 所示。

```

连接成功
> Packet {cmd: 'connack', retain: false, qos: 0, dup: false, length: 2, ...}
订阅成功
收到来自 /sys/gaxq4rwUcCA/ESP8266/thing/event/property/post 的消息 {"id":"dht11","version":"1.0","method":"thing.event.property.post","params":{"CurrentHumidity":55.000000}}
收到来自 /sys/gaxq4rwUcCA/ESP8266/thing/event/property/post_reply 的消息 {"code":200,"data":{},"id":"dht11","message":"success","method":"thing.event.property.post","version":"1.0"}
连接成功
> Packet {cmd: 'connack', retain: false, qos: 0, dup: false, length: 2, ...}
订阅成功
重新连接...
收到来自 /sys/gaxq4rwUcCA/ESP8266/thing/event/property/post_reply 的消息 {"code":200,"data":{},"id":"dht11","message":"success","method":"thing.event.property.post","version":"1.0"}
连接成功

```

Figure 12. The system subscribes to information from the cloud platform

图 12. 系统向云平台订阅信息

接收到消息后，我们可以根据需要进行处理，例如对接收到的数据进行进一步的分析和处理，并采取相应的控制动作来调整农场环境等。重要代码以 MQTT 通信为例，阿里云物联网平台与本系统实时通信的代码如下：

```

# 导入必要的库
import paho.mqtt.client as mqtt
import numpy as np

# 定义 MQTT 客户端回调函数
def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe("topic") # 订阅指定的主题

def on_message(client, userdata, msg):
    data = np.array(msg.payload.decode("utf-8").split(',')) # 将接收到的消息转换为 NumPy 数组
    # 处理接收到的数据，例如进行数据分析和控制动作
    # ...
    print(data)

# 创建 MQTT 客户端实例
client = mqtt.Client()

# 设置客户端回调函数
client.on_connect = on_connect
client.on_message = on_message

# 连接到阿里云物联网平台
client.username_pw_set("username", "password") # 设置用户名和密码，如果需要的话
client.connect("iot.example.com", 1883, 60) # 连接到指定的 MQTT 服务器和端口号，并设置超时时间

```



```
# 发布消息到主题，例如上报农场环境数据
```

```
client.publish("topic", "1.23,4.56,7.89") # 发布消息到指定的主题，例如上报农场环境数据
```

```
# 循环接收消息
```

```
while True:
```

```
client.loop() # 处理网络流量和接收消息
```

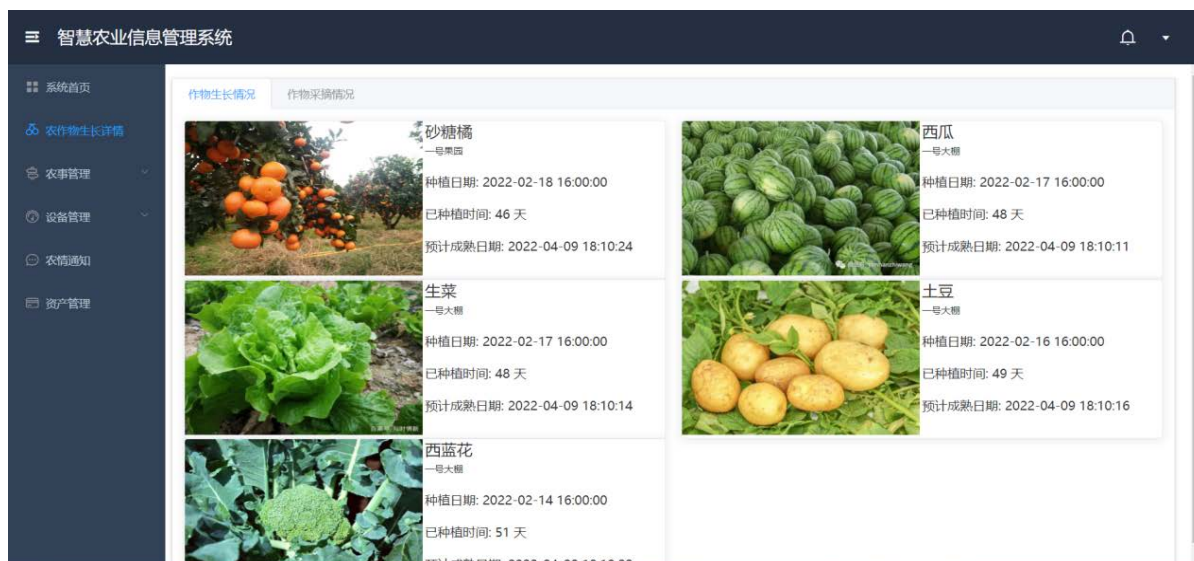


Figure 13. Crop growth detailed page

图 13. 农作物生长详细页

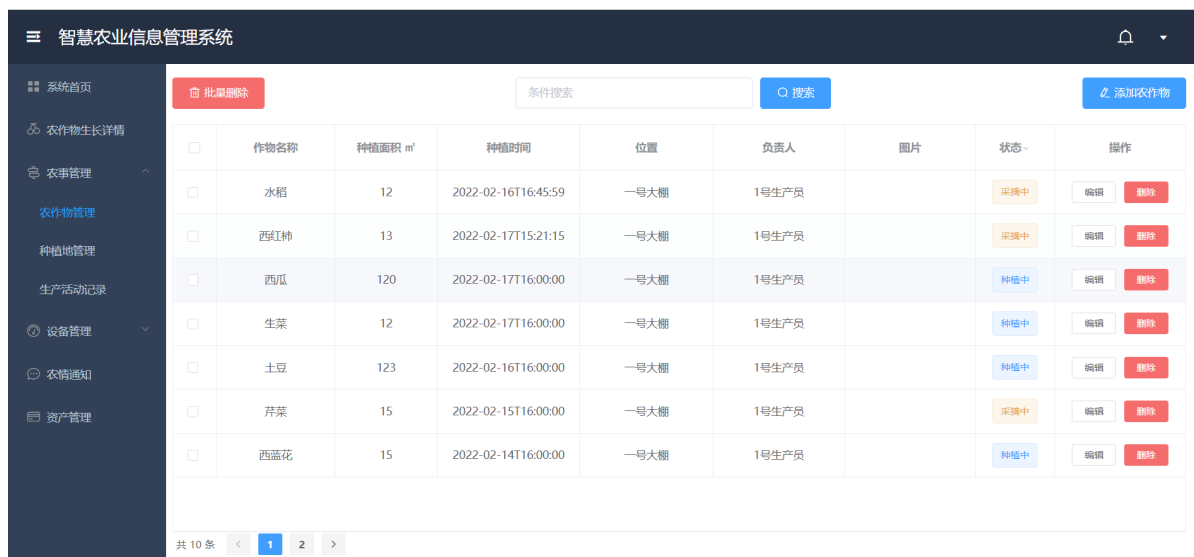


Figure 14. Crop management detailed page

图 14. 农作物管理详细页

4.3. 系统模块设计与实现

农作物生长详情页展示了农作物生长与成熟采摘的详细情况，包括生长时间，近期图片，预计成熟

日期等。

农作物生长详情页如图 13 所示。

农事管理模块包括了农作物管理、种植地管理、生产活动记录三个部分。

农作物管理展示各个农作物的基本信息，如图 14 所示

5. 结论

AI 智慧农场系统的设计主要包括信息管理系统、阿里云物联网平台、与 ESP8266 开发板连接的各类传感器。传感器采集数据后,通过 MQTT 协议与阿里云 IOT 平台通信,此时数据会实时上传到阿里云 IOT 平台,信息管理系统再通过 MQTT 协议订阅消息,接收阿里云 IOT 平台发布的物模型数据,在 AI 智慧农场系统进行监控与展示。这样做的好处是克服了 WIFI 通信距离短的不足,使得信息管理系统不管在何地登录都能够实时获取传感器数据。AI 智慧农场系统在系统测试时选择了黑白盒测试。先使用白盒测试,根据系统过程逻辑,逐个进行测试,以此来达到系统是否符合用户的预期的目的。然后使用黑盒测试对系统进行随机输入的检验,看是否能够达到需求规格说明书的要求。每一次的测试过程都需要编写和保存好每一次的测试用例和测试数据,以便后期维护人员对系统的维护。传统的生产方式,对环境参数的获取和控制都无法做到及时性,且无法对农作物生长状况、生长阶段等作出系统的记录与判断。目前网络上虽然存在包含显示作物生长状况和生长环境数据等功能的软件,但是大部分软件的针对性都很强,例如只提供某些农作物生长状况或生长阶段等。结合现阶段的状况和功能需求,设计发明 AI 智慧农场,将有助于实现农业现代化管理,推动我国乡村振兴。

基金项目

本文系 2023 年江西省大学生创新创业训练计划项目 AI 智慧农场(项目编号: S202310895031)。

参考文献

- [1] 陈相余. 云平台下消息推送系统的设计与实现[D]: [硕士学位论文]. 成都: 电子科技大学, 2021.
- [2] 张寅生. 新一代人工智能计算模型的创新及其哲学意义[J]. 学术界, 2021(5): 59-69.
- [3] 谢晋飞. 基于嵌入式的智慧农业监测系统设计[D]: [硕士学位论文]. 太原: 中北大学, 2021.
- [4] 郝刚, 陈淑花. 基于 Arduino 与 Blinky 云平台的农业物联网系统设计[J]. 武汉工程职业技术学院学报, 2021, 33(4): 13-16.
- [5] 李志伟, 东伟, 黄双成. 基于 DHT11 的农业大棚温湿度监控系统设计[J]. 工业仪表与自动化装置, 2021(1): 39-43.
- [6] 赵春江. 智慧农业的发展现状与未来展望[J]. 中国农业文摘-农业工程, 2021, 33(6): 4-8.
- [7] 李寒. 基于物联网的智慧农业大棚控制系统设计[D]: [硕士学位论文]. 保定: 河北大学, 2021.