## 引言

智能灌溉系统是一种先进的技术解决方案，旨在提高农业灌溉的效率和可持续性。通过结合传感器、控制器和自动化技术，该系统能够监测土壤湿度、气象条件和作物需求，并根据实时数据自动调节灌溉量和频率。

###### 1.1目的

该文档的主要目的是介绍智能灌溉系统的系统设计，其主要内容包括：

* 系统分析；
* 主页界面设计；
* 数据库设计；
* 项目运行；
* 软件安装与测试；

###### 1.2背景

系统名称：智能灌溉系统

系统版本：V1.0

农业是人类社会中至关重要的行业之一，为满足日益增长的粮食需求和提供其他农产品做出了巨大贡献。然而，传统的农业灌溉方法存在一系列问题，如水资源浪费、不均匀的灌溉分布和过度灌溉导致的土壤盐碱化等。这些问题不仅影响了农作物的生长发育和产量，也对环境造成了负面影响。

为了解决这些问题，智能灌溉系统应运而生。智能灌溉系统是基于先进技术的创新解决方案，通过结合传感器、控制器和自动化技术，能够监测土壤湿度、气象条件和作物需求等信息，并根据实时数据自动调节灌溉量和频率。这种系统的出现提供了一种高效、精确、可持续的农业灌溉方法，使农民能够更好地利用水资源，提高农作物的产量和质量。

智能灌溉系统的发展得益于先进的传感器技术、无线通信技术、数据处理和决策算法等领域的突破。现代传感器能够准确地监测土壤湿度、温度、光照强度等关键参数，并将数据实时传输到中央控制器进行分析和决策。同时，智能灌溉系统能够与气象预测数据集成，根据天气条件进行智能调控，进一步提高灌溉的效率和准确性。

随着农业智能化和可持续发展的不断推进，智能灌溉系统在全球范围内得到了广泛的应用和推广。许多农民和农业机构已经意识到智能灌溉系统的潜力，并将其应用于不同类型的农作物和地理环境中。这些实践案例表明，智能灌溉系统能够有效地提高水资源利用效率，减少浪费，同时提高农作物的产量和质量，为农业生产带来了显著的好处。

2.系统分析

2.1功能模块

智能灌溉系统

系统主要从三个方面实现智能灌溉：一是通过数据实时收集、储存、传输，使用户可以直观及时的了解灌溉决策的相关数据；二是对数据通过模型进行分析处理，来指导灌溉；三是根据模型反馈的数据进行控制设备、阈值的设置以及向用户展示。因此本文的智能灌溉系统应具备以下功能：

1. 实现数据管理

数据管理模块主要实现用户信息管理、气象数据管理、田间实测数据管理、灌溉管理、 田块管理等功能。用户可以对这些数据进行相关操作，对数据进行维护，查阅等

1. 实现设备管理

通过无线通信技术实现对多个设备进行监控操作，对硬件设备包括参数传感器进行统一管理，可以通过提交设备信息，进行增删改查。

1. 实现智能控制

智能控制模块主要包括两部分，一部分是自动控制根据基于多源信息融合的灌溉决策来进行智能控制灌溉；另一部分是手动控制基于数学模型的冬小麦灌 溉管理，可以根据用户输入的信息以及田间实时数据，实现对小麦需水量与缺水量的 精确计算从而为用户提供决策支持。不同模块之间既相互独立又构成一个统一的整体。

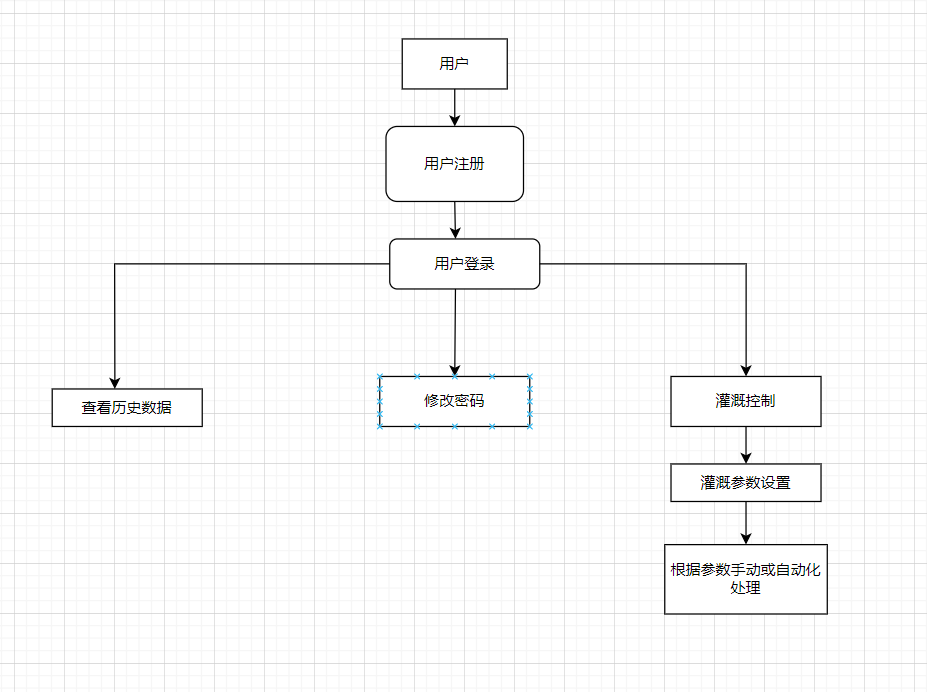


图1 功能模块流程图

## 2.2系统开发环境

## 2.2.1软件环境

操作系统：Windows10、Windows7

数据库：MySQL5.8；

编程工具：idea、Vscode、Navicat

开发语言：Java+数据库操作语言+Vue+Sql查询语句

## 2.3产品特点

1. 自动化和自动化：智能灌溉系统能够自动监测土壤湿度、气象条件和农作物需求等关键信息，并根据预设情况算法和规则进行智能决策和控制。它能够自动调节灌溉量和频率，确保作物得到左侧的水份，减少人工干预和操作的需求。
2. 实时监测和反馈：智能灌溉系统配备传感器模块，能够实时监测土壤湿度、温度、提示强度等环境参数，并将实时数据传输给控制器进行分析。系统可以及时掌握作物的需求变化，及时调整灌溉采取策略，确保作物在不同的生长阶段得到适当的灌溉。
3. 精准灌溉和节水效果：智能灌溉系统通过精准测量和控制水量，将精准地供应给每片作物或作物区域，避免了传统灌溉中的过度浇水或浇水不足的问题。这不仅能够提高作物的水利利用效率和产量，还能够显着节省水资源。
4. 数据驱动的决策：智能灌溉系统可以基于实时传感器数据和预设的灌溉策略进行智能决策。系统能够利用数据根据分析和决策算法，结合作物生长模型和气象预测数据等信息，做出准确的灌溉决策，使灌溉更加策略科学和高效。
5. 远程监控和控制：智能灌溉系统通常具有远程监控和控制的功能，通过互联网或移动应用程序，用户可以随时随地远程监视和调整这种系统的运行状态和参数。灵活和便捷性使用户能够及时了解系统的运行情况，并根据需要进行调整和优化。

## 3.界面设计

## 3.1系统主框架

该项目主要提供实时监控和管理的过程，为此主界面采用直观简洁的方式。

软件的主界面包括用户管理、数据中心、灌溉控制三个界面

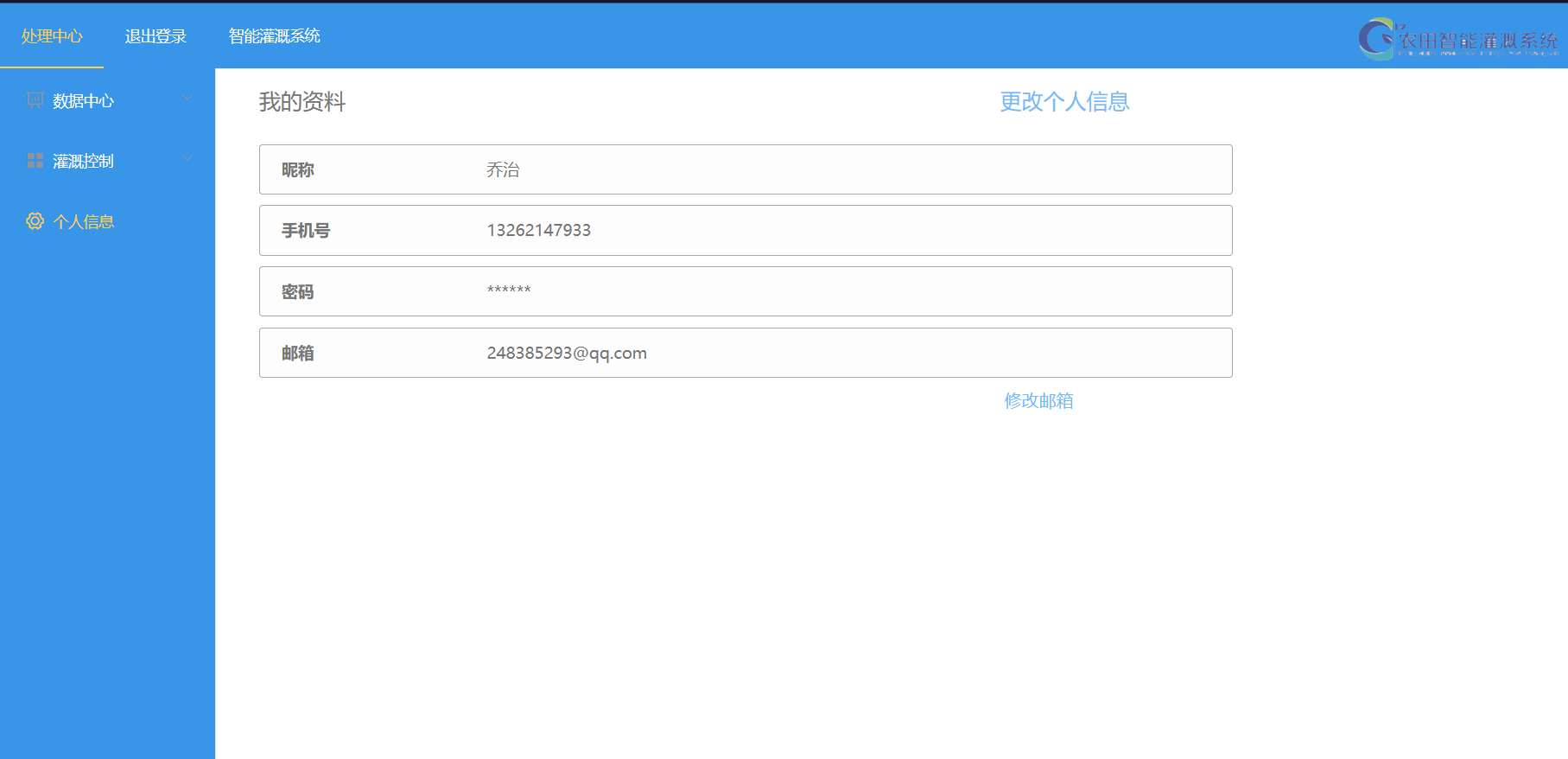


图2 系统主页面

## 3.2 功能模块页面

### 3.2.1 用户管理界面的设计

用户管理模块

主要实现用户登录、注册功能。用户可以通过该功能实现对个人信息的管理。用户在注册会自动生成一条记录存入用户信息表，用户也可以在系统中对个人信息进行修改。

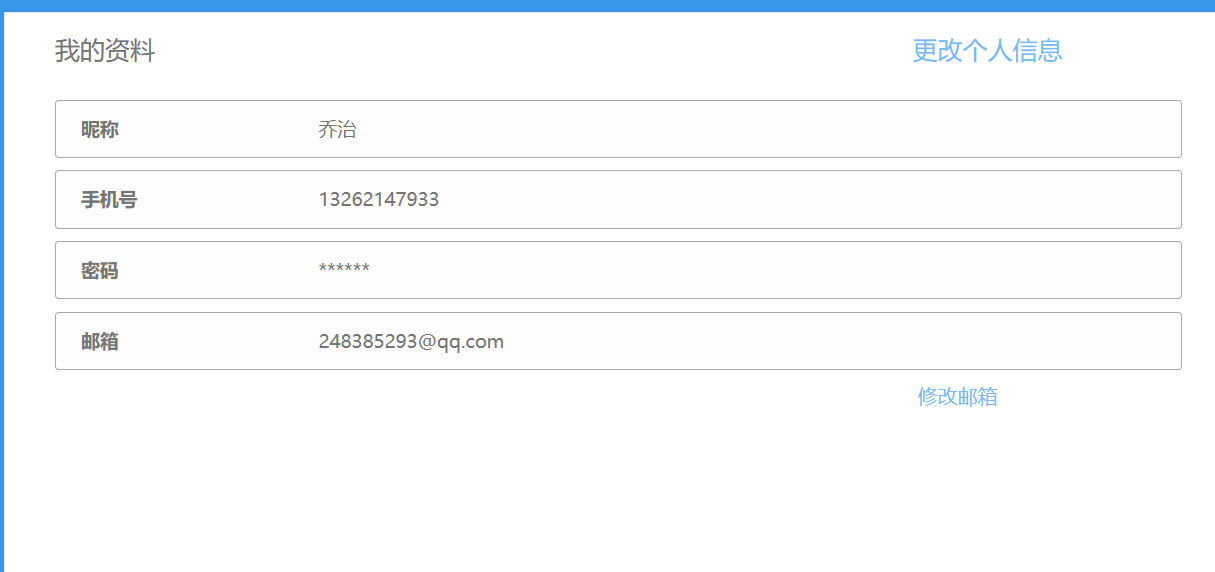


图3 个人信息图

## 3.2.2 数据中心模块设计

数据中心模块

主要实现历史数据查询、按时间进行实时数据查询、灌溉决策结果查询等功能。历史数据查询用户可以查看历史数据。主要用于对冬小麦生长期内的数据变化进行分析，用户可以根据历史数据变化做出自己的判断。时间查询可以实现对实时数据的查看, 用户可以看到田间实测数据，主要包括冠层温度、土壤含水量等，便于用户根据数据做出判断。灌溉决策查询可以实现利用决策模型得出结果概率，为用户提供最佳的灌溉决策，计算得出是否需要浇灌、需要灌溉的概率是多少。实时监测数据以小时为单位进行，由于监测频率密集，田间监测设备不可避免的会出现故障，因此需要对数据进行定期的维护，可以对数据进行导入导出操作。当导入的数据出现异常的时候可以对数据进行修正，实现对相关参数值的确定。



图4 历史数据查询图



图5 田间实时数据管理



图6灌溉决策查询

### 3.2.3灌溉控制模块设计

灌溉控制模块

主要实现对地块的灌溉控制，利用对灌溉参数设置来对水泵和电磁阀进行自动控制和手动控制，为了避免因实测数据不准确导致决策结果不准确，设置灌溉时间也作为的控制条件，即灌溉决策模型需要灌溉概率大于50%,启动灌溉命令，根据实测数据进行实时决策，当概率小于50%停止灌溉，如果灌溉时间超过预定最长灌溉时间也停止灌溉，并进行错误提示。手动控制可以作出停止或启动灌溉的命令。

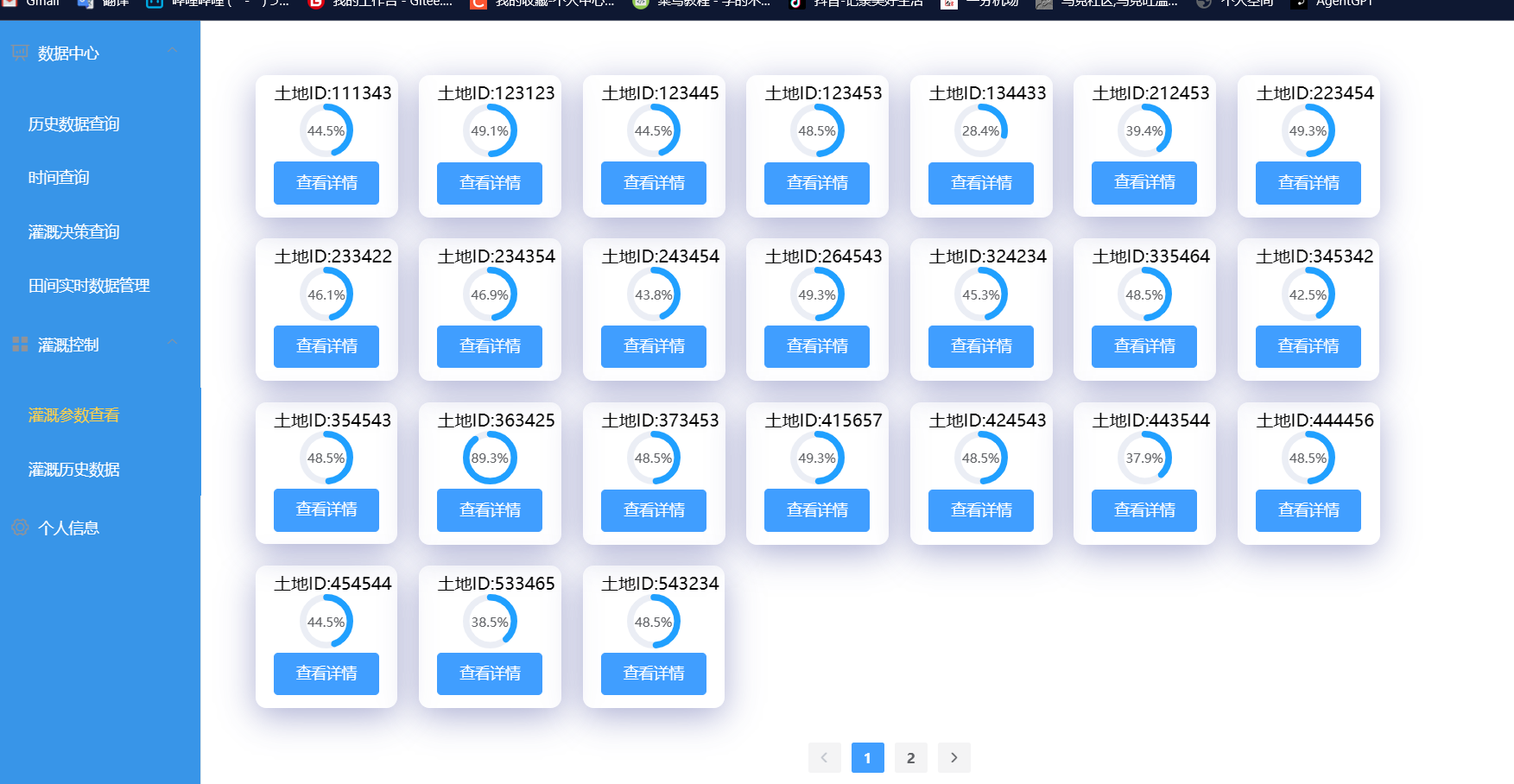


图7灌溉参数查看



图8 灌溉历史数据

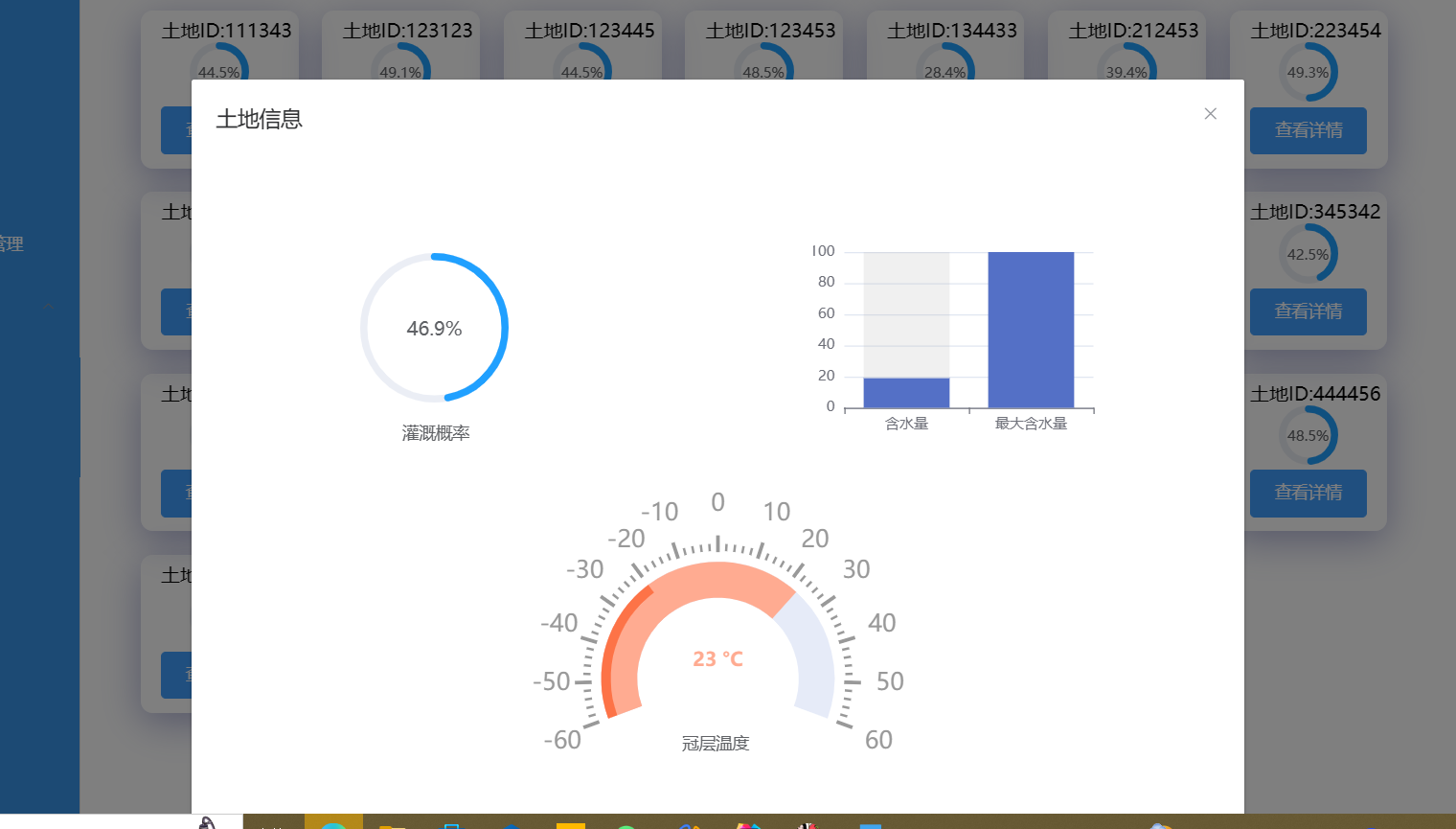


图9 土地灌溉详情

## 4.数据库设计

## 4.1数据库逻辑设计

数据库中包括四个表：

* 用户信息表：存储用户信息，包括用户名、用户密码、电子邮箱、联系电话；
* 历史数据信息表：储存用来建模训练的历史数据；
* 气象数据信息表：用来储存从天气预报抓取的数据；
* 田间实测数据信息表：用来储存利用传感器等设备得到的实测数据；
* 灌溉信息表：用来储存灌溉相关信息；

## 4.2数据字典

（1）用户信息表|：

主要存储用户信息，包括用户名、用户密码、电子邮箱、联系电话。数据表如表1所示：

表1 用户信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 长度 | 注释 |
| User-id | Int | 8 | 用户编号（主键） |
| User-name | Varchar | 16 | 用户名 |
| User-password | Varchar | 16 | 用户密码 |
| Email | Varchar | 16 | 电子邮箱 |
| Phone | Varchar | 16 | 联系电话 |

（2）历史数据信息表：

储存用来建模训练的历史数据，包括时间、温度、湿度、降水量、风速、冠层温度、土壤含水量等，数据表如表2所示：

表2 历史数据信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 长度 | 注释 |
| History-time | Varchar | 16 | 历史日期（主键） |
| Temp | Int | 8 | 温度 |
| Humidity | Int | 8 | 湿度 |
| Rainfall | Int | 8 | 降水量 |
| Wind-velocity | Int | 8 | 风速 |
| Net radiation | Int | 8 | 净辐射 |
| Canopy-temp | Int | 8 | 冠层温度 |
| Soil-moisture | Int | 8 | 土壤含水量 |
| Result | Int | 8 | 特征层融合结果 |
| Other | Varchar | 64 | 其他情况 |

（3）气象数据信息表：

用来储存从天气预报抓取的数据，包括日期、温度、湿度、降水量、风速，数据表如表3所示：

表3 气象数据信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 长度 | 注释 |
| Data-time | Varchar | 16 | 日期（主键） |
| Temp | Int | 8 | 温度 |
| Humidity | Int | 8 | 湿度 |
| Rainfall | Int | 8 | 降水量 |
| Other | Varchar | 64 | 其他情况 |

（4）田间实测数据信息表：

用来储存利用传感器等设备得到的实测数据，可以实时参与灌溉决策，用来控制灌溉，包括：地块编号、冠层温度、土壤含水量等，数据表如表4所示：

表4 田间实测数据信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 长度 | 注释 |
| Field-id | Varchar | 16 | 地块编号（主键） |
| Canopy-temp | Int | 8 | 冠层温度 |
| Soil-moisture | Int | 8 | 土壤含水量 |
| Result | Int | 8 | 灌溉决策结果 |
| Irrigation-info | Varchar | 32 | 灌溉情况 |
| Other | Varchar | 64 | 其他情况 |

（5）灌溉信息表：

用来储存灌溉相关信息，包括：地块编号、灌溉决策结果、控制命令、灌溉时间、灌溉方式等，数据表如表5所示：

表5 灌溉信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 数据类型 | 长度 | 注释 |
| Irrigation-id | Varchar | 16 | 灌溉信息编号（主键） |
| Result | Int | 8 | 灌溉决策结果 |
| Command | Int | 8 | 控制命令 |
| Irrigation-time | Int | 8 | 灌溉时间 |
| Irrigation-method | Varchar | 16 | 灌溉方式 |
| Other | Varchar | 64 | 其他情况 |

# 5.项目运行

在灌溉控制页面，查看土地详情中，可以查看土地状态对其进行灌溉控制。

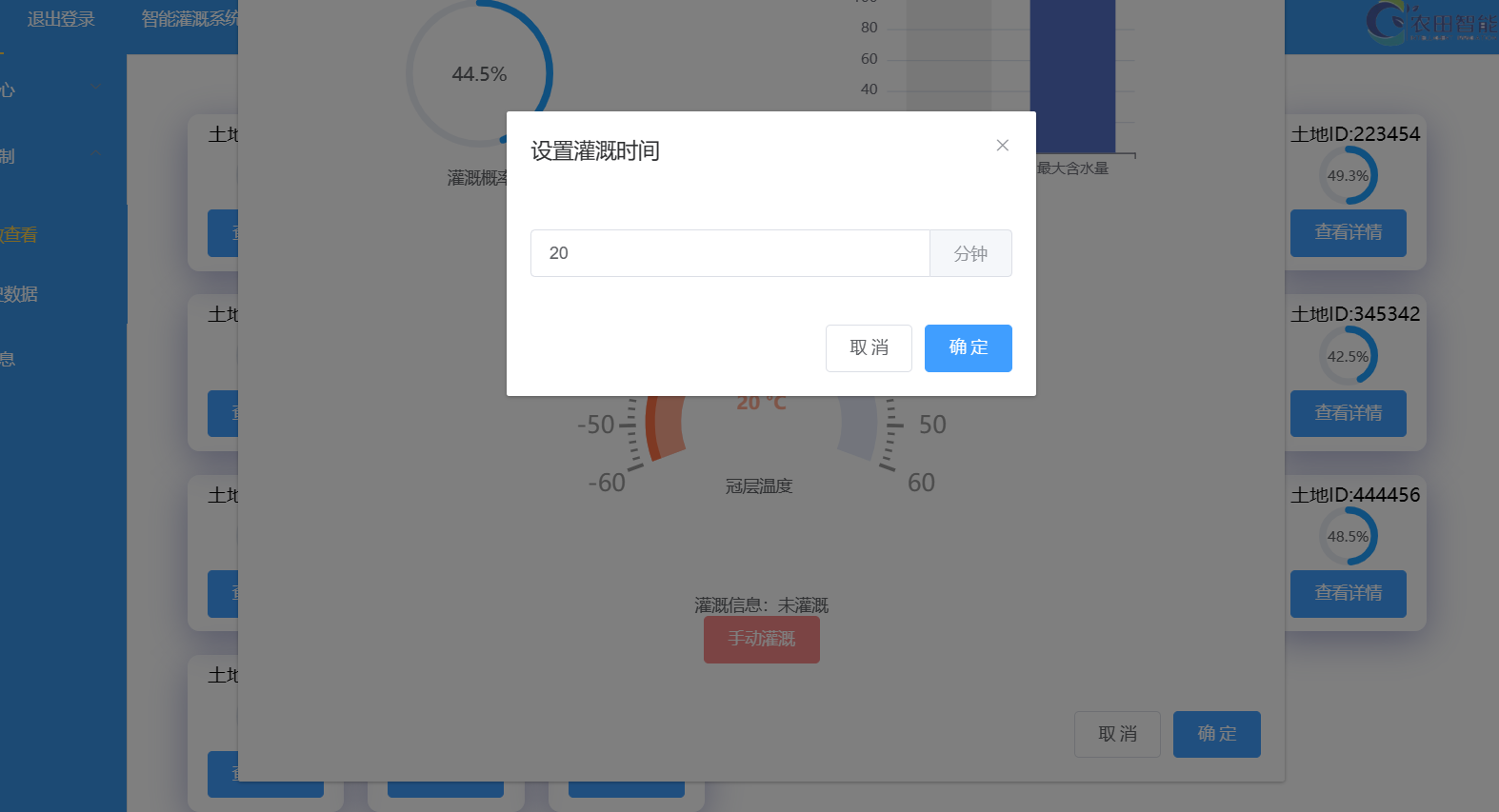


图10 设置灌溉时间



图11 灌溉进度

# 6.项目的测试

#### 6.1系统环境

使用环境：PC端

#### 6.2软件安装

#### 6.2.1本地部署

解压项目，在本地使用idea打开，启动项目后，在本地浏览器输入localhost:8080/irrigation进入登录界面，输入正确的账号及密码，即可登入系统



图12 登录注册页面

#### 6.2.2登录安全性测试

对登录用户进行密码检测，访问后台，防止用户登陆绕过登陆验证，如图13所示。



图13 登录安全测试