大理大学本科毕业设计

基于SpringBoot和Vue3的云南主要农作物产量分析与预测系统的设计与实现

**Design and Implementation of a Yunnan Major Crop Yield Analysis and Forecasting System Based on Spring Boot and Vue 3**

学 院： 数学与计算机学院

学生 姓名： 恭浩杰

学 号： 20221108900212

指导 教师： 杨润标

专 业： 计算机科学与技术

年级（班级）： 2022级2班

起止 日期： 2025年 6月— 2025年 12 月

摘 要

农作物产量受到气候波动、耕作制度以及市场变化等多种因素共同影响，如果只是依靠经验来做估算，往往难以达到精细化管理与科学决策所提出的要求。围绕云南省主要农作物统计数据，这篇文章完成了一套产量预测与可视化分析系统的设计并加以实现，用于支撑数据管理、预测建模以及结果展示等业务环节。

系统采用前后端分离方式进行开发。后端借助Spring Boot给出REST接口服务，结合Spring Security与JWT完成登录认证与权限控制，并借助MySQL开展数据存储与管理；前端基于Vue3与Vite加以构建，配合Element Plus实现交互界面开发，使用ECharts展示产量统计与预测结果，系统给出数据集导入与管理、模型参数配置、预测任务创建与执行、结果对比分析与报表导出等功能，并支持对不同模型的预测效果开展指标评价。

在典型业务场景下开展功能与性能测试之后，系统能够稳定完成数据导入、批量预测与图表渲染等关键流程，降低了预测分析过程中重复操作带来的成本。这篇文章工作为农作物产量预测的工程化实现给出可供参考的思路，并可为智慧农业场景下的生产规划与决策支持提供一定借鉴。

**关键词：**农业产量预测；数据可视化；SpringBoot；Vue3；LSTM

**Abstract**

Crop yield is jointly influenced by factors such as climate variability, cultivation practices, and market changes. If estimation relies only on experience, it is often difficult to meet the requirements of refined management and evidence-based decision-making. Based on statistical data of major crops in Yunnan Province, this paper designs and implements a yield forecasting and visual analytics system to support key business processes, including data management, predictive modeling, and result presentation.

The system is developed with a front–back separation architecture. The back end provides REST APIs built on Spring Boot, using Spring Security and JWT for authentication and access control, with MySQL for data storage and management. The front end is built with Vue 3 and Vite; Element Plus is used to develop the interactive user interface, and ECharts is used to visualize yield statistics and forecasting results. The system offers functions such as dataset import and management, model parameter configuration, forecasting task creation and execution, result comparison and analysis, and report exporting, and it supports metric-based evaluation of forecasting performance across different models.

After functional and performance testing under typical business scenarios, the system proved capable of reliably completing key workflows such as data import, batch forecasting, and chart rendering, reducing the cost caused by repetitive operations during forecasting and analysis. This work provides a practical reference for engineering crop-yield forecasting into real-world applications and offers insights for production planning and decision support in smart agriculture contexts.

**Keywords:** crop yield forecasting; data visualization; Spring Boot; Vue 3; LSTM

目 录

摘 要 I

Abstract II

目 录 III

第一章 绪 论 1

1.1 选题的背景 1

1.1.1 选题的背景 1

1.1.2 选题的意义 1

1.2 国内外研究状态 1

1.2.1 国内研究现状 1

1.2.2 国外研究现状 2

1.3 这篇文章的组织架构 3

第二章 需求获取与分析 4

2.1 功能需求 4

2.2 标识参与者 4

2.3 标识用例 5

2.3.1 求精用例 5

第三章 系统总体设计 22

3.1 设计原则与目标 22

3.2 系统架构设计 22

3.3 前后端技术选型对比 23

3.4 部署架构说明 23

第四章 系统详细设计 24

4.1 将用例映射到时序图 24

4.1.1 用户注册时序图 24

4.1.2用户登录时序图 24

4.1.3添加作物信息时序图 25

4.1.4新增区域信息时序图 25

4.1.5 导入历史农作物数据时序图 26

4.1.6配置预测模型时序图 26

4.1.7执行产量预测时序图 27

4.1.8 生成分析报告时序图 28

4.1.9 农户提交咨询时序图 28

4.1.10 农业专家回复咨询时序图 29

4.2 将系统对象映射到编程环境 30

4.2.1 用户注册类图 30

4.2.2 用户登录类图 30

4.2.3 添加作物信息类图 31

4.2.4添加区域信息类图 31

4.2.5 导入农作物数据类图 31

4.2.6 配置预测模型类图 32

4.2.7 执行产量预测类图 32

4.2.8 生成分析报告类图 32

4.2.9农户提交咨询类图 33

4.2.10专家回复咨询类图 33

4.3 将持久性对象映射到数据库 34

4.3.1 用户表 34

4.3.2 区域对象表 34

4.3.3 作物对象表 35

4.3.4 系统配置对象表 35

4.3.5 数据文件对象表 36

4.3.6 预测模型对象表 37

4.3.7 报告章节表 37

4.3.8 在线咨询对象表 38

4.3.9 咨询消息对象表 39

第五章 系统实现与测试 40

5.1 系统实现 40

5.1.1 公有类实现 40

5.1.2 用户登录页面实现 41

5.1.3 仪表盘页面实现 43

5.1.4 数据中心页面实现 44

5.1.5 在线咨询管理页面实现 45

5.1.6 报告中心页面实现 46

5.1.7 系统设置页面实现 47

5.1.8 用户管理页面实现 48

5.1.9 系统日志管理页面实现 49

5.1.10 预测中心页面实现 50

5.1.11 数据可视化页面实现 52

5.1.12 天气监测页面实现 53

5.1.13 气象分析页面实现 54

5.2 系统测试 55

5.2.1 测试方法 55

5.2.1 测试用例 56

5.3 测试结果 58

5.4 测试小结 61

第六章 总 结 62

6.1 工作总结 62

6.2 不足总结 62

6.3 展望与未来 62

6.4 结论 62

参考文献 63

致 谢 65

第一章 绪 论

1.1 选题的背景

1.1.1 选题的背景

农业和国计民生有紧密联系，粮食安全同样属于国家安全的重要部分，近些年来，人口数量增加、耕地资源紧张以及气候条件波动这些因素叠加到一起，给农业生产带来更大压力：既需要提高单位面积产量、稳定主要粮食供应，也要考虑到种植结构优化。但在实际中，很多决策仍然主要依靠经验判断，对历史产量数据、气象资料和市场信息的系统利用不够充分，这容易造成产量预测不够准确，资源投入也难以做到更加合理、更加精细[7][9]。

在农业信息化和数字乡村建设推动下，各类农作物产量及气象数据不断积累，但由于缺乏统一管理与可视化平台，数据多以分散表格形式存在，难以转化为有效决策依据。时间序列分析和机器学习方法在其他领域已得到验证，将其引入农作物产量预测，可在既有数据基础上为决策提供量化参考。因此，构建集历史数据管理、多模型预测和可视化展示于一体的农作物产量预测与分析系统，既能为政府制定粮食生产与储备政策提供依据，也能为合作社和种植大户安排播种面积、选择品种提供决策支持，具有现实需求和应用价值[3][10][17]。

1.1.2 选题的意义

本课题拟构建农作物产量预测与可视化系统，通过集成历史产量及相关影响因素，利用模型对主要作物未来产量进行预测，并以图表直观展示结果，帮助管理者及时掌握产量变化趋势，提前识别减产风险或增产机会。农业主管部门可据此调整种植结构和政策方向，粮食储备与流通企业可优化收储和调控，农户也能据预测信息合理安排生产，减少因盲目种植和供需失衡造成的损失，从而提高决策科学性，服务粮食安全和农民增收[3][17]。

这套系统可以被看作是推动农业信息化、服务数字乡村的一个具体做法。它把原本分散的产量数据集中到一块儿统一管理，再拿来做分析处理，让数据真正“用起来”，也能让相关人员慢慢养成借助数据来做判断的习惯，对云南来讲，该系统可以作为数字化转型的试点应用，为智慧农业建设积累经验，并对农业高质量发展和数字乡村战略的落地起到一定支持作用[3][7]。

1.2 国内外研究状态

1.2.1 国内研究现状

在国内，对农作物产量的预测研究已经积累了不少成果，常用的手段仍然以传统的统计模型和基于经验的分析为主。其中，时间序列方法用得比较普遍，ARIMA（自回归积分滑动平均）模型可以捕捉产量序列的趋势和季节性波动，因此在粮食作物的预测里使用得较多，相关研究一般以水稻、小麦等作物的历史统计数据作为基础来构建模型，并借助对预测结果和实际数值的比对，验证这类模型在短期预测中的可用性，针对一些经济作物，也有研究把 ARIMA 用于波动预警，比如在云南咖啡等产量分析中，用它来帮助判断产量变化及其可能带来的市场影响。

灰色预测方法在国内同样比较常见。农业生产受到自然条件的影响很明显，而有些地区数据样本有限，灰色系统理论在这类情境下更容易落地，以 GM(1, 1) 为代表的模型能在“小样本、贫信息”条件下提取趋势特征，所以经常被用于省或者县一级的粮食产量走势预测，也有研究在灰色模型基础上做改进，例如带进马尔可夫链或者遗传算法，以增强对复杂系统的拟合能力并提升预测精度，总的来看，传统统计模型仍然是国内产量预测研究的重要基础，方法相对成熟、实现成本不高，便于在实际部门中推广使用[1][9]。

除了模型研究，农业数据可视化与信息系统的建设近年受到更多在意，各地在推进农业大数据平台和农情监测系统建设时，开始把统计分析和可视化展示结合起来，用于辅助管理决策，一些地区构建了集数据采集、分析与展示为一体的平台，用来跟踪主要作物的种植面积、生长状况与产量变化。也有基于 GIS 的可视化应用，用于呈现产量的空间分布与历史变化，方便管理人员快速把握农情信息。

与此同时，部分研究尝试将预测模型嵌入业务系统，通过交互界面向用户给出预测服务，使模型结果能够在业务里直接使用。用户上传本地数据后就能得到预测输出，一定程度上拉近了“研究模型”和“业务应用”之间的距离，但从整体看，国内产量预测与可视化系统仍以局部应用为主，多数面向特定地区或单一作物，功能通用性和完善程度还有提升余地，所以，围绕实际需要构建预测与可视化一体的平台，仍然具有较大的改进空间和应用价值[3][17]。

1.2.2 国外研究现状

国外对农作物产量预测的研究开始得比较早，研究方法也更多样。随着计算能力的提高，机器学习的方法也开始被应用到农业上，并且和传统的统计分析一起使用来提高预测的精度和稳定性。LSTM 经常被用在处理产量时间序列上，因为它可以刻画出较长时间跨度内的依赖关系。有研究把历史气象数据同作物生长信息融合起来，对美国玉米和大豆单产展开预测，其结果比传统方法要好。

除了深度学习以外，集成学习模型在产量预测中也很常见。以XGBoost为例，该类模型能够同时处理多维特征，加入气候、土壤、管理措施等变量后可以用来做区域尺度作物产量回归预测，具有较好的稳定性。另一方面，Prophet等开源工具也被用于农业时间序列建模，提供更便捷的预测实现途径。从总体上讲，国外的研究方法由传统的统计模型向机器学习模型扩展，并且经常通过多模型对比或者组合来提高预测效果。

从数据的角度来讲，国外研究的一个重点就是多源遥感融合。卫星遥感可以获得大面积作物的生长信息，结合地面观测数据和历史统计数据，可以在较大的空间尺度上进行产量的监测与预测。在美国、欧洲等地已经用在了农业监测当中。利用作物生长期内NDVI、EVI等植被指数的变化，结合气象数据，可以对区域产量进行阶段性的估计，为产量预报提供依据。

从应用的角度来说，美国农业部和欧盟的相关机构已经建立起了比较完善的农情监测和产量预测体系，使得预测结果可以被应用到农业管理和政策制定当中。近些年计算机视觉的方法也被引入到遥感估产的研究中，一些研究尝试用CNN 处理遥感影像，从图像中提取特征用于产量预测，在空间分辨率和预测精度上取得一定的进展。总体上，国外的相关研究已经形成了“模型方法-多源数据-业务应用”的路径，对我国开展产量预测和可视化系统研究有借鉴意义[6][18][20]。

1.3 这篇文章的组织架构

全文严格按照高校毕业设计模板来进行组织，共分为6章：

第一章为绪论，交代研究背景与意义，并对农作物产量预测、智慧农业信息化和数据可视化等相关研究进行简要梳理，最后说明这篇文章的研究内容、技术路线和章节结构。

第二章为需求获取与分析，从农业管理的实际场景出发明确系统建设目标与使用对象，整理功能需求与典型用例，并从技术、操作与经济三个角度评估系统可行性，为后续设计提供依据。

第三章为系统设计，在需求分析基础上给出总体方案与技术选型，包含系统架构、前后端模块划分、数据库设计、部署方案，以及 ARIMA、Prophet、LSTM 等模型的选型思路。

第四章为系统实现，根据设计方案说明核心功能的实现过程，包括前端页面、后端接口、业务流程、数据导入与预处理、预测任务执行和结果展示，用关键代码来说明实现要点。

第五章为系统测试，验证主要功能和性能，给出测试方法、测试用例、测试结果，并关注系统稳定性、可用性、异常处理的表现。

第六章为总结与展望，总结本文的工作及系统实现的效果，指出不足并提出后续可以改进和扩展的方向。

第二章 需求获取与分析

2.1 功能需求

（1）账号与权限管理

系统支持用户注册与登录，并采用基于角色的权限控制机制。管理员负责账号与权限配置，不同角色用户可执行的操作范围有所区别。

（2）基础数据维

系统支持对作物和行政区域等基础信息进行管理，包括作物属性及地区层级信息，为数据分析和预测提供基础支撑。

（3）数据导入与清洗

系统支持导入 Excel、CSV 等数据文件，并在导入过程中自动完成字段匹配、单位处理和异常数据过滤，同时提供导入进度与结果反馈。

（4）模型配置与预测任务

用户可配置预测模型及相关参数，提交预测任务后系统在后台完成计算，并生成相应的预测结果。

（5）预测结果管

系统集中展示预测结果及评估指标，并支持与历史数据进行对比分析，便于评估预测效果。

（6）数据可视化分

系统通过多种图表形式展示产量数据与预测趋势，支持按作物和区域进行交互式分析。

（7）报告生成与导出

系统可生成作物分析报告，并支持将结果导出为 PDF 或 Excel 文件。

（8）系统设置

系统提供基础配置功能，管理员可根据实际需求调整运行参数，提升系统的可维护性和灵活性。

2.2 标识参与者

根据系统功能设计，本系统涉及以下四类主要参与者：

（1）系统管理员

系统管理员拥有最高操作权限，主要负责用户与角色管理、基础数据维护以及系统参数配置等工作。同时，管理员还可进行模型配置、预测任务的创建与执行、结果查看和报告导出，是系统运行和日常维护中的核心角色。

（2）农业部门用户

来自农业主管机构的业务人员，具备基础数据维护、数据导入、模型创建与执行、结果查看、报告生成等主要业务权限，但无法进行系统配置与用户管理。

（3）农户用户

权限较低，可浏览系统提供的公开数据分析与可视化图表，获取地区产量趋势、价格信息和预测结果，用于辅助生产决策，无法导入数据或创建预测任务。

（4）游客用户

未注册或未登录用户，仅可访问注册与登录页面。注册后成为正式用户，方可根据分配角色使用系统功能。

2.3 标识用例

主要用例包括：

（1）用户注册；

（2）用户及权限管理；

（3）基础数据维护；

（4）数据导入；

（5）数据清洗；

（6）模型训练与预测；

（7）预测结果查看；

（8）数据可视化分析；

（9）报告生成与导出；

（10）系统参数设置

2.3.1 求精用例

**用例1：用户注册用例**

**用例名**：用户注册用例

**范围**：农作物产量预测与可视化分析系统

**级别：**系统用户层（访客端）

**主要参与者：**访客（Visitor）

**涉众关注点：**

**用户：**用户输入正确账号密码即可登录，并在登录后访问其权限内功能。系统需快速响应，并明确提示成功或失败原因。

**管理员：**关注系统登录的安全性，防止未授权用户进入，登录失败次数过多时系统应有防范，并记录登录日志以备审计。

**系统管理员：**关注认证机制的健壮性（如验证码校验、密码强度）、登录流程对服务器性能的影响，以及日志记录用于监控异常登录行为

**前置条件：**

1.系统允许开放注册功能，访客可以访问注册页面；

2.（如果启用验证码）系统已为注册页面准备好图形验证码供访客填写

**后置条件：**

1.系统新建一个用户账户记录，并赋予默认的系统角色（例如农户角色）；

2.新用户账户已可用于登录

**主事件流：**

1.访客打开系统的用户注册页面。

2.系统显示注册所需的表单字段（用户名、密码等），并呈现一张图形验证码供验证。

3.访客填写用户名、密码等必要信息，输入图形验证码后提交注册请求。

4.系统校验用户填写的信息和验证码是否合法、有效；如果校验通过，则调用后端注册接口创建新用户账户。

5.系统将新用户的默认角色设定为“农户”或预设的基础角色，保存用户基本信息到数据库。

6.系统向访客反馈注册成功的信息（例如跳转到登录页面或直接提示注册成功）。  
**备用事件流：**

1.若提供的信息无效或验证码错误：

2.在步骤4，系统发现验证码不匹配或必填信息有遗漏/格式错误，注册失败，提示相应错误信息（例如“验证码不正确”或“用户名已被占用”）；

3.访客返回修改注册信息或刷新验证码后重新提交注册申请，用例重新从步骤3开始。

如图2-1所示，用户注册用例的用例图展示了访客与系统交互完成新用户注册的过程。

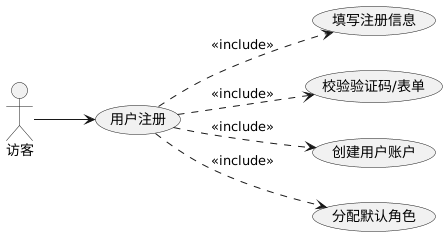


图2-1 用户注册用例图

表2-1 用户注册用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 访客 | Visitor | 尚未注册或登录的用户，只能访问注册和登录页面。 |
| 2 | 注册页面 | Registration Page | 供新用户填写信息以创建账户的网页。 |
| 3 | 图形验证码 | CAPTCHA | 用于验证操作人为真实用户的图片验证码。 |
| 4 | 用户账户 | User Account | 系统中保存用户凭证和基本信息的账户记录。 |
| 5 | 系统角色 | System Role | 授予用户的一组预定义权限（如管理员或农户）。 |
| 6 | 农户角色 | Farmer Role | 默认授予新用户的基础系统角色。 |

**用例2：用户登录用例**

**用例名：**用户登录用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（用户端）

**主要参与者：**用户（User）

**涉众关注点：**

**用户：**用户输入正确的账号密码就可以登录，在登录之后可以访问其权限范围内的功能，系统应该快速响应，并明确提示成功或者失败的原因。

**管理员：**管理员要求只有合法用户才能进入系统，登录时需要进行身份校验并记录登录日志；连续失败次数较多的时候可以启用验证码等其它校验来提高安全性。

**前置条件：**

1.用户已拥有系统账户（例如已经完成注册，或由管理员创建了账户）；

2.系统运行正常，登录页面可供访问。

**后置条件：**

3.用户成功登录系统后，获得一个有效的会话令牌或Session，后续请求将被授权；

4.系统生成一条登录日志，记录该用户的登录时间、IP等信息

**主事件流：**

1.用户打开系统的登录页面。

2.系统显示用户名、密码输入框（以及在需要时显示的验证码输入框）。

3.用户输入账户用户名、密码（以及验证码，如有要求）后提交登录请求。

4.系统验证用户名和密码是否匹配现有账户，必要时一并验证验证码；若验证通过，生成JWT令牌或建立登录会话，并更新用户上次登录时间等信息。

5.系统为该用户建立登录会话环境，返回登录成功的状态（例如重定向到仪表盘页面）。

6.用户进入系统主界面，可以根据其角色访问相应功能模块

**备用事件流：**

1.若用户名或密码不正确：

2.在步骤4，系统发现凭证无效，则拒绝登录并反馈“用户名或密码错误”的提示；

3.系统记录一次失败尝试（可能在多次失败后要求输入验证码），用户可返回步骤3重新输入正确的凭证后再尝试登录。

如图2-2所示，用户登录用例的用例图展示了用户与系统进行账户认证的交互过程。

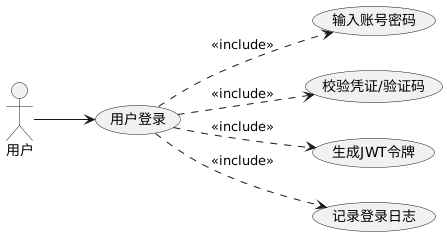


图2-2 用户登录用例图

表2-2 用户登录用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 用户 | User | 已拥有账户并可以登录系统的人员。 |
| 2 | 登录页面 | Login Page | 供用户输入用户名、密码（及验证码）的网页界面。 |
| 3 | JWT | JSON Web Token | 基于JSON的用户身份认证令牌。 |
| 4 | 会话 | Session | 维持用户登录状态的服务器会话。 |
| 5 | 登录日志 | Login Log | 记录用户登录时间、IP等信息的系统日志条目。 |
| 6 | 验证码 | CAPTCHA | 用于区分真人用户、提高登录安全性的验证码。 |

**用例3：添加作物信息用例**

**用例名：**添加作物信息用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（管理员端）

**主要参与者：**管理员（Admin）

**涉众关注点：**

**管理员：**管理员负责维护作物基础信息，新增品种时可快速录入，并确保名称与编码不重复。

**农业部门专员：**专员关注作物数据的完整与准确，新作物出现时需及时更新，避免影响后续数据录入和预测分析。

**系统：**系统需保证作物基础信息的唯一性和一致性，避免重复记录，并确保新增作物可被数据导入和预测等功能正常识别

**前置条件：**

1.管理员已成功登录系统并拥有维护基础数据的权限；

2.系统作物信息管理界面正常可用

**后置条件**

系统成功新增一条作物基础信息（包含名称、编码等），并立即生效，可在数据导入及预测功能中选择使用。

**主事件流**

1.管理员进入系统基础资料管理模块，选择作物管理查看作物列表。

2.管理员点击新增作物。

3.系统弹出新增作物表单，包含作物名称、编码、类别、描述等字段。

4.管理员填写作物信息并提交。

5.系统校验数据合法性（必填项、名称/编码唯一性）。

6.校验通过后，系统保存作物信息并返回作物列表，显示新添加的作物记录。

**备用事件流**

1.当作物名称或编码已存在时，系统提示“作物已存在”，管理员可选择修改信息后重新提交，或取消本次操作

2.当输入信息不完整或不符合校验规则时，系统给出相应提示，管理员补充或更正后可再次提交。

添加作物信息的用例图如图2-3所示，展示了管理员通过系统完成作物新增操作的交互过程。

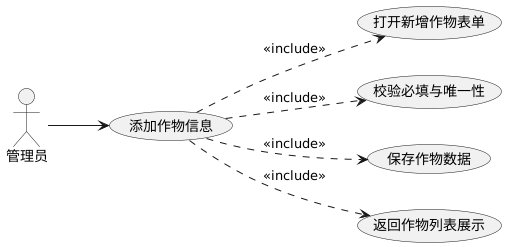


图2-3 添加作物信息用例图

表2-3 添加作物信息用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 管理员 | Admin | 具备最高权限，负责系统用户和数据管理的用户。 |
| 2 | 作物 | Crop | 指用于农业生产的植物，如水稻、小麦等。 |
| 3 | 作物名称 | Crop Name | 作物的名称，在系统中须唯一。 |
| 4 | 作物编码 | Crop Code | 用于唯一标识某种作物的代码。 |
| 5 | 作物品种 | Crop Variety | 某一作物的具体品种类型。 |

**用例4：添加区域信息用例**

**用例名：**添加区域信息用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（管理员端）

**主要参与者**：管理员（Admin）

**涉众关注点：**

**管理员：**管理员对行政区域信息进行维护，新增或者细分区域时可以及时录入，保证数据完整且不重复。

**农业部门专员：**农业部门专员要覆盖全面的区域数据，使农作物数据可以准确归属，出现新的种植区域时应该同步更新。

**系统：**保证区域名称、编码的唯一性，维护好层级关系；新添加的区域可以被数据导入、统计分析、报表等功能识别。

**前置条件：**

管理员已登录系统并具备区域管理权限；新增子区域时，其上级区域已存在。

**后置条件：**

系统成功新增一条行政区域信息（含名称、编码、级别及隶属关系），并立即生效，可用于下拉选择、数据录入及统计分析

**主事件流：**

1. 管理员进入基础资料管理 → 区域管理，查看区域列表（层级展示）。

2. 管理员点击新增区域。

3. 系统显示新增区域表单（名称、编码、级别、上级区域、描述等）。

4. 管理员填写信息并提交。

5. 系统校验必填项及同级唯一性，通过后保存区域及层级关系。

6.系统提示新增成功，并在区域列表中显示新区域节点。

**备用事件流：**

1.区域名称或者编码重复的时候，系统会提示“区域已存在”，管理员可以修改之后再提交，也可以取消操作。

2.信息缺失或者格式不符合要求的时候，系统提示校验错误，管理员修改之后再次提交。

添加区域信息用例图如图 2-4 所示，是管理员新增区域记录的交互流程图。

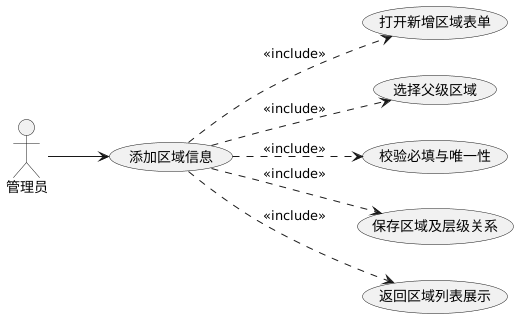


图2-4 添加区域信息用例图

表2-4 添加区域信息用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 农业部门用户 | Agricultural Department User | 来自农业主管部门，具备数据导入等权限的系统用户。 |
| 2 | Excel文件 | Excel File | Microsoft Excel 格式的电子表格文件（.xlsx）。 |
| 3 | CSV文件 | CSV File | 以逗号分隔值形式存储表格数据的纯文这篇文章件。 |
| 4 | 字段映射 | Field Mapping | 将导入数据的字段对应到系统预定义字段的过程。 |
| 5 | 单位统一 | Unit Standardization | 将数据中的不同计量单位转换为统一单位。 |
| 6 | 异常过滤 | Outlier Filtering | 筛除数据中明显异常或错误的记录。 |

**用例5：导入农作物数据用例**

**用例名：**导入农作物数据用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（农业部门端）

**主要参与者：**农业部门专员（Agriculture Dept Staff）

**涉众关注点：**

**农业部门专员：**专员希望导入的数据要完整并且更新及时，导入后可以直接用来做报表分析和预测建模，保证后续结果可靠。

**管理员：**管理员关心导入过程的稳定性，需要能识别并拦截重复、错误的数据；在必要的时候自动补充新增加作物、新区域的信息。导入过程产生日志便于追溯。

**前置条件：**用户已登录系统并具备数据导入权限，且已准备符合模板要求的 CSV/Excel 数据文件。

**后置条件**：系统生成一条数据导入任务记录；文件中的历史数据成功入库；缺失的作物或区域基础数据被自动创建；导入数据立即可用于分析、报表与预测功能。

**主事件流：**

1.专员进入数据中心 → 导入数据，上传符合要求的数据文件。

2.系统创建导入任务并开始解析，任务状态显示为“进行中”。

3.系统逐行校验并处理数据，自动匹配或新增作物与区域基础信息。

4.数据按统一单位规则处理后写入业务数据库，并持续更新导入进度。

5.导入完成后，系统将任务标记为“已完成”，显示成功数量及提示信息，导入数据对分析模块可见。

**备用事件流：**

1. 当数据文件格式或结构不符合要求时，系统中止导入并将任务标记为失败，同时提示错误原因，用户修正后可重新导入。

2. 当仅部分记录存在异常时，系统对异常数据进行跳过或标记，其余数据继续导入，并在任务详情中列出异常原因，便于后续处理。

导入农作物数据用例图如图2-5所示，用于说明农业专员发起导入及系统处理流程。

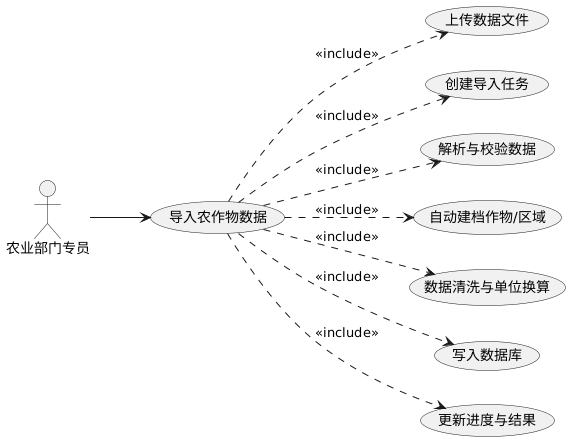


图2-5 导入农作物数据用例图

表2-5 导入农作物数据用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 数据清洗 | Data Cleansing | 通过清除错误、缺失等问题来提高数据质量的过程。 |
| 2 | 缺失值 | Missing Value | 指数据集中缺失或未填写的数据项。 |
| 3 | 异常值 | Outlier | 指偏离大部分数据且可能有误的数据值。 |
| 4 | 数据格式统一 | Data Format Normalization | 将数据格式（如日期格式）调整为一致的规范。 |
| 5 | 重复数据 | Duplicate Data | 在数据集中出现多次的相同记录。 |

**用例6. 配置预测模型用例**

**用例名：**配置预测模型用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（农业部门端）

**主要参与者：**农业部门专员（Agriculture Dept Staff）

**涉众关注点：**

**农业部门专员：**专员希望能方便地选择预测算法并设置参数，以适配不同作物和区域的预测需求。

**管理员：**管理员统一维护可用模型范围，避免使用不稳定或未验证的算法，保障系统运行稳定。

**前置条件：**用户已登录并具备模型配置权限，系统中存在可用于训练的历史数据。

**后置条件：**系统新增一条模型配置记录，并在模型列表中生效，后续创建预测任务时可直接选用。

**主事件流：**

1.专员进入预测中心的模型管理页面，查看现有模型列表。

2.点击新增模型。

3. 系统弹出配置表单，填写模型名称，选择算法类型（ARIMA、Prophet、LSTM等），设置参数。

4.专员提交配置。

5.系统对名称、参数和算法可用性进行校验，校验通过后保存配置。

6.保存成功后，新模型出现在模型列表中，可用于后续预测任务选择。

**备用事件流：**

1.模型名称重复时，系统提示错误信息，专员修改后可再次提交。

2.参数不符合规则时，系统给出具体校验提示，专员修正后重新提交。

3.所选算法不可用时，系统提示该类型不可用，专员需更换算法或取消操作。

配置预测模型用例图如图2-6所示，用于说明农业专员新增模型配置的交互流程。

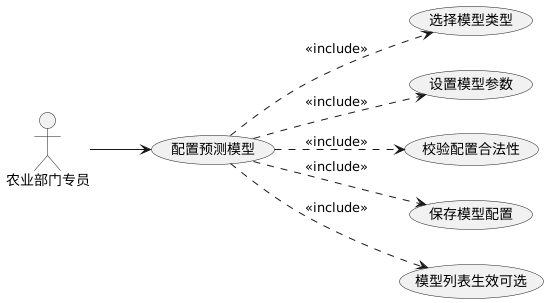


图2-6 配置预测模型用例图

表2-6 配置预测模型用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 算法类型 | Algorithm Type | 所选用的预测模型算法类别（如ARIMA、LSTM）。 |
| 2 | 预测年份 | Prediction Year | 需要预测的目标年份。 |
| 3 | 气象场景 | Meteorological Scenario | 预测中假设的天气情景（如正常年景、干旱）。 |
| 4 | 预测任务 | Prediction Task | 提交的包含模型训练和预测过程的作业。 |
| 5 | 后端引擎 | Backend Engine | 后端执行模型训练和预测计算的服务或模块。 |

**用例7：执行产量预测用例**

**用例名：**执行产量预测用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（农业部门端）

**主要参与者：**农业部门专员（Agriculture Dept Staff）

**涉众关注点：**

**农业部门专员**：农业部门专员希望预测任务创建流程简单，可以及时看到执行进度，可以查询历史记录和结果。

**决策者**：决策者更关心预测结果是否可信，以便于生产安排和管理决策。

**管理员/运维**：管理员/运维要求预测以异步的方式进行，尽量不影响系统整体的性能，并保留任务记录以便于追溯。

**前置条件：**系统具备可用历史数据与预测模型，用户已登录并拥有预测权限

**后置条件：**系统生成预测任务并完成计算，预测结果及评估信息入库，可在前端页面或报表中查看，并记录至历史预测列表。

**主事件流：**

1.专员进入预测中心，新建预测任务。

2.选择作物、区域、预测周期及预测模型，配置相关参数并提交。

3.系统校验数据与配置合法性，通过后后台执行预测计算。

4.系统完成模型训练与推理，保存预测结果并更新任务状态。

5.系统通知专员预测完成，结果可视化展示。

**备用事件流：**

1.历史数据不足：系统拒绝预测并提示原因。

2.预测执行失败：任务标记失败并返回错误信息。

3.任务耗时较长：预测以异步方式执行，用户可查看任务状态，完成后查看结果。

如图2-7所示，执行产量预测用例图描述了农业专员配置并运行预测任务的过程。

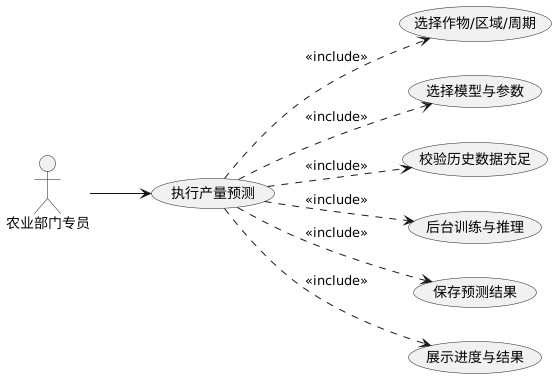


图2-7 执行产量预测用例图

表2-7 执行产量预测用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 置信区间 | Confidence Interval | 围绕预测值给出的范围，用于表示预测结果的不确定性。 |
| 2 | 评估指标 | Evaluation Metrics | 用于衡量预测准确度的量化指标（如误差率）。 |
| 3 | 历史实绩数据 | Historical Actual Data | 历史记录的真实产量数据，用于与预测结果比较。 |
| 4 | 多维度分析 | Multidimensional Analysis | 从作物、区域、时间等多个角度对数据进行分析。 |

**用例8：生成分析报告用例**

**用例名：**生成分析报告用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（农业部门端）

**主要参与者：**农业部门专员（Agriculture Dept Staff）

**涉众关注点：**

**农业部门专员**：专员希望能快速生成包含历史数据与预测结果的分析报告，并支持在线查看与导出保存。

**决策者**：决策者希望通过报告直观了解不同作物、不同区域的产量现状与趋势，用于辅助决策。

**管理员**：管理员关注报告数据的一致性与准确性，并能维护报告模板及导出格式。

**前置条件：**

系统中已有足够的数据和预测结果可供汇总（如已经完成若干次数据导入和预测任务）；

用户已登录并能够访问“报告中心”页面（农业部门专员、管理员或有权限的农户均可访问报告功能）

**后置条件：**

系统生成了一份最新的产量分析报告实体，包括选定范围内的历史产量概况、区域对比分析和未来产量预测结果；

报告内容在前端呈现，供用户查看和交互（如切换视图、查看详细数据）；用户可选择将报告导出为文件（PDF、Excel 等）保存

**主事件流：**

1.专员进入报告中心页面。

2.选择作物、区域等报告范围（也可沿用默认条件）。

3.点击生成报告。

4.系统汇总历史数据与预测结果，按模板生成图表、表格和结论摘要。

5.报告在页面中展示，专员可查看并选择导出。

6.系统按所选格式生成并提供报告文件下载

**备用事件流：**

1.数据不足时，系统提示无法生成报告。

2.生成过程中出现异常时，系统终止生成并返回错误原因。

3.导出失败时，系统提示失败信息，用户可重试或更换导出格式。

生成分析报告用例图如图2-8所示，用于说明农业专员按条件生成产量分析报告的操作流程。

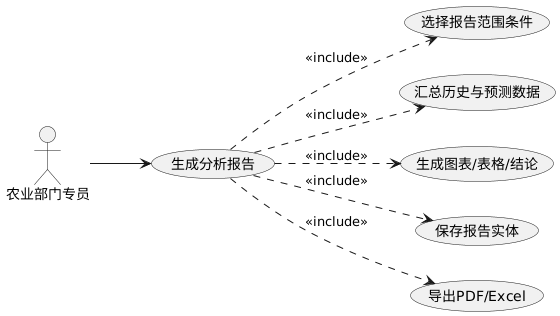


图2-8 生成分析报告用例图

表2-8 生成分析报告用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 柱状图 | Bar Chart | 用矩形柱条表示数据大小对比的图表。 |
| 2 | 折线图 | Line Chart | 用折线连接数据点以反映趋势变化的图表。 |
| 3 | 热力图 | Heatmap | 以颜色深浅显示数据强度分布的图表。 |
| 4 | 交互式选择 | Interactive Selection | 用户可动态选择作物、区域等条件实时更新可视化的功能。 |

**用例9：农户提交咨询用例**

**用例名：**农户提交咨询用例

**范围：**农作物产量预测与可视化分析系

**级别：**系统用户层（农户端）

**主要参与者：**农户（Farmer）

**涉众关注点：**

**农户**：农户希望可以方便地提交问题，并且能够尽快收到可靠的答复。

**农业部门专员**：农业部门专员希望咨询的内容要表述清楚、信息全面，便于快速判断和处理。

**管理员**：管理员需要对咨询流程进行跟踪管理，不能遗漏或者延误。

**前置条件：**农户登录系统成功，咨询类别和处理规则已经配置。

**后置条件：**系统生成咨询记录并标注为待回复，同时分配给对应处理队列；农户可以在系统中查看咨询的状态。

**主事件流：**

1.农户进入咨询中心，点击发起咨询。

2.选择咨询类别，填写问题主题与详细描述并提交。

3.系统校验信息有效性，创建咨询记录并分配处理对象。

4.系统提示提交成功，咨询显示在农户咨询列表中。

**备用事件流：**

1.信息不完整：系统提示补充后重新提交。

2.无法自动分配：咨询记录保存为“未指派”，由管理员后续处理。

3.用户取消：不生成咨询记录。

如图2-9所示，农户提交咨询用例图描述了农户在系统中提交咨询的基本过程。

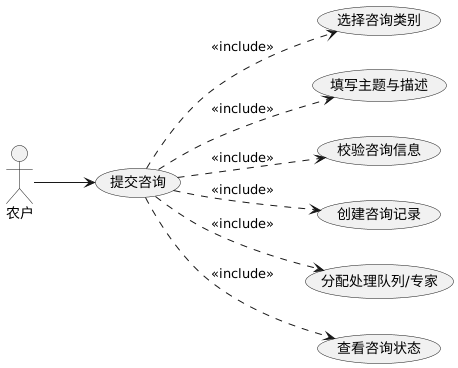


图2-9 农户提交咨询用例图

表2-9 农户提交咨询用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 分析报告 | Analysis Report | 汇总作物数据、预测结果和对比分析的报告文档。 |
| 2 | 统计数据 | Statistical Data | 报告中呈现的汇总数值（如平均值、总和等）。 |
| 3 | 区域对比 | Regional Comparison | 比较不同地区数据或结果差异的分析方式。 |
| 4 | PDF | PDF | 可移植文档格式，一种保持版式的通用文档格式。 |
| 5 | 导出 | Export | 将报告以指定格式文件输出保存的操作。 |

**用例10：专家回复咨询用例**

**用例名：**专家回复咨询用例

**范围：**农作物产量预测与可视化分析系统

**级别：**系统用户层（农业部门端）

**主要参与者：**农业部门专员（Agriculture Dept Staff）

**涉众关注点：**

**农业部门专员**：高效查看并回复农户咨询，记录回复内容与时间，可标记问题状态。

**农户**：及时获得专业、有效的回复，支持必要的后续沟通。

**管理员**：咨询过程可追踪、可审计，确保回复及时且质量可控。

**前置条件：**

1.系统中存在至少一条待回复的农户咨询问题（状态为待回复，已分配给相应部门或专家）；

2.农业部门专员已登录系统，并有权限查看和回复咨询问题（通常对应其负责的类别或由管理员赋予权限）

**后置条件：**

1.农户的咨询记录更新为已回复状态，系统在该咨询会话中新增一条专家的回复消息，记录回复内容和时间；

2.咨询的状态可能更新为“已解决”或“已回复”。如果支持多轮交流，咨询状态可暂时标记为“跟进中”，等待农户的确认或进一步提问；一旦问题解决，咨询可被标记为关闭并记录关闭时间

**主事件流：**

1.专员进入咨询管理，查看待处理咨询列表。

2.选择咨询并查看详情及历史记录。

3.填写回复内容（可含附件）并发送。

4.系统保存回复、更新咨询状态并通知农户。

5.专员可将咨询标记为已解决或关闭。

**备用事件流：**

1.信息专员向农户询问未结的情况。

2.已被他人处理：不需要重复回复，用例结束。

3. 发送失败，系统提示失败，专员稍后再试。

专家回复咨询用例图如图 2-10 所示，是农业专家查看、回复农户咨询的流程。

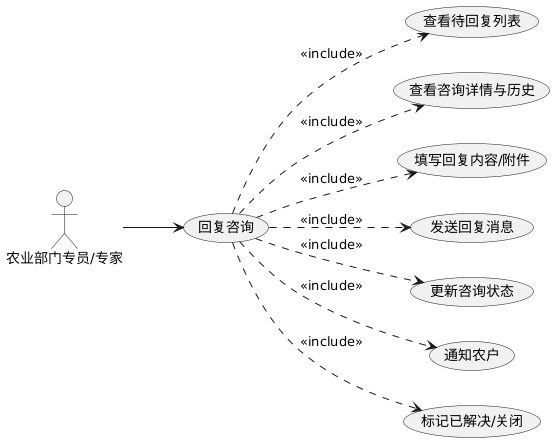


图2-10 专家回复咨询用例图

表2-10 专家回复咨询用例术语表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 标识 | 说明 |
| 1 | 默认区域 | Default Region | 系统预设的默认地区选择。 |
| 2 | 平台公告 | Platform Announcement | 由管理员发布的面向所有用户的系统通知。 |
| 3 | 安全策略 | Security Policy | 系统设置的安全规则，如密码强度要求、登录失败限制等。 |
| 4 | 通知邮箱 | Notification Email | 系统用于发送通知或告警的电子邮箱地址。 |

第三章 系统总体设计

3.1 设计原则与目标

在农作物产量预测经营系统的设计当中，这篇文章按照模块清晰而且彼此独立的思路来组织功能，减小模块之间的耦合度，这样就能方便后期的维持和升级，其界面设计着重于易用性，削减操作成本并改善使用者的感受，从安全角度来讲，采用依靠角色的权限控制，并且对关键输入执行校验，以此确保系统运行稳定，还给后续的功能拓展以及部署调整保留空间。

3.2 系统架构设计

农作物产量预测经营系统经由B / S架构来达成，其后端由Spring Boot供应REST接口服务，融合Spring Web，Spring Data JPA以及Spring Security以执行业务处理和权限控制，数据存储利用MySQL；而且具备CSV / Excel数据导入功能并能导出PDF报告，借助整合模型库来执行预测计算，前端依托Vue3和Vite开发，加上Element Plus，Pinia和ECharts，做到页面交互并且把预测结果以图表形式表现出来。

从功能划分角度看，该系统包含诸多模块，账号与权限运作，基础数据守护，数据录入及查询，预测运作，报表分析以及日志与通知等模块都是其一部分，这些模块合成了系统的总体功能架构，具体内容参照图3 - 1。



图3-1 系统总体架构图

例如，系统集成的时间序列预测模型之一ARIMA模型可表示为

(3-1)

式中：

—— 时刻的产量值；

—— 常数项；

—— 自回归项系数；

—— 自回归阶数；

—— 移动平均项系数；

—— 移动平均阶数；

—— 时刻的随机误差项（白噪声）；

—— 滞后 阶的随机误差项。

3.3 前后端技术选型对比

本系统前后端技术选型对比情况如表3-1所示。最终，后端采用Spring Boot框架，前端采用Vue3框架，数据库选用MySQL，图表库选用ECharts等技术组合，以充分发挥各自优势。

表3-1 前后端主要技术选型对比

|  |  |  |  |
| --- | --- | --- | --- |
| 层次 | 候选方案 | 最终选用 | 选用理由 |
| 前端框架 | Angular、React、Vue3 | Vue3 | 渐进式框架，易上手且生态成熟，具备高性能和模块化开发优势 |
| 数据可视化库 | D3.js、Chart.js、ECharts | ECharts | 提供丰富的可视化图表，配置简单，易于与前端框架集成 |
| 后端框架 | Spring MVC、Django、Spring Boot | Spring Boot | 基于Spring生态，开箱即用配置简化开发，生态完善且社区支持度高 |
| 数据库 | MySQL、PostgreSQL、MongoDB | MySQL | 支持复杂查询和事务，性能稳定，易于部署维护，在项目需求下足够胜任 |

3.4 部署架构说明

农作物产量预测经营系统采取 B/S 三层架构部署，用户经由浏览器访问前端页面，其请求被后端 Spring Boot 服务所处理，并与 MySQL 数据库执行读写操作。在实际部署过程中，前端静态资源往往由 Nginx 等 Web 服务器供应，而后端应用则运行于独立服务器或者云主机环境当中；数据库一般位于同一局域网内，以此来缩减访问时延并优化读写性能，此方案利于按照访问量实施扩展，而且能够凭借负载均衡加强并发能力，进而保障系统平稳运行。

第四章 系统详细设计

4.1 将用例映射到时序图

4.1.1 用户注册时序图

图4 - 1显示了用户注册时序图，该图表现出访客完成注册的主要流程，访客在注册页面填写信息并提交之后，请求被认证控制器（AuthController）所接收，并转交给认证服务（AuthService）去处理。服务端按序执行验证码校验，用户名/邮箱是否唯一以及获取默认角色这些操作；当校验通过以后，就创建并保存用户信息，还要生成 JWT 访问令牌与刷新令牌，而且要把注册日志记录下来，系统把注册结果反馈给前端，页面则提示注册已经完成。

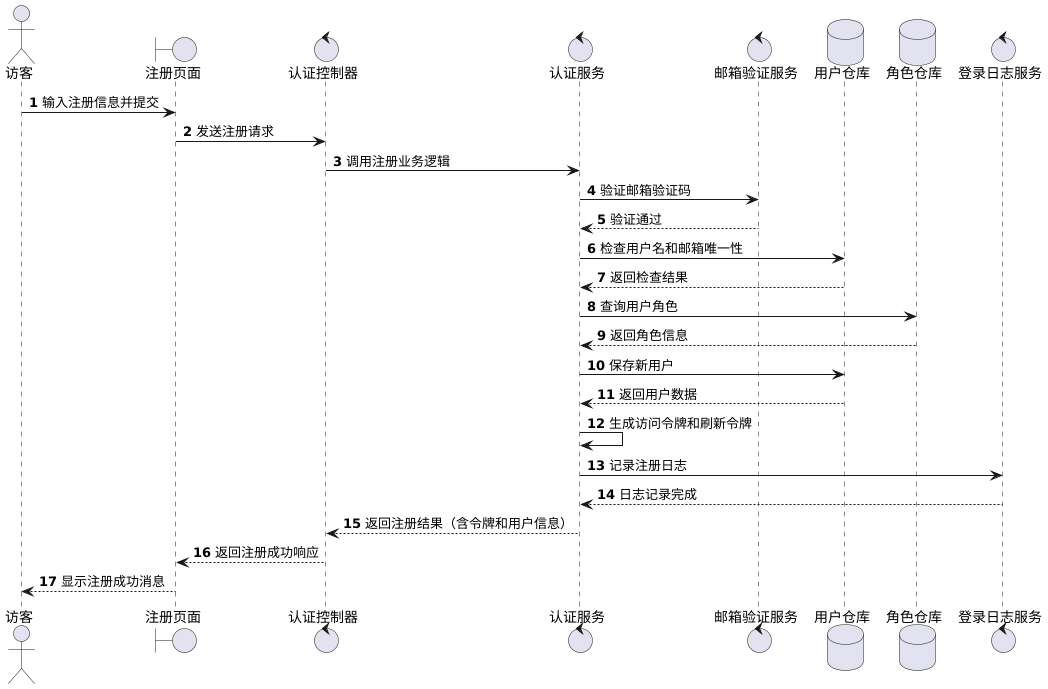


图4-1用户注册时序图

4.1.2用户登录时序图

图4 - 1显示了用户注册时序图，该图表现了访客完成注册的主要流程，访客在注册页面填写相关信息之后提交，此请求被认证控制器（AuthController）接收，并转交给认证服务（AuthService）去处理。服务端按照顺序执行验证码校验，用户名或者邮箱的唯一性核查以及默认角色的获取这些操作，当校验通过以后，会创建用户信息并且保存下来，还要生成JWT访问令牌和刷新令牌，而且要把注册事件记入日志当中，系统把注册的结果反馈给前端，页面则提示注册已完成。

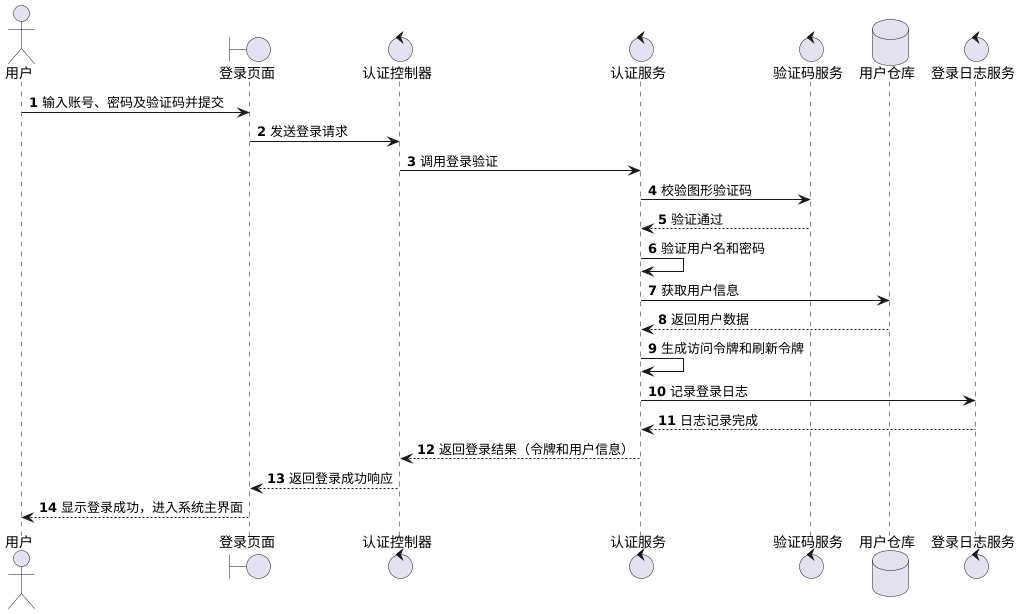


图4-2 用户登录时序图

4.1.3添加作物信息时序图

如图4-3所示，添加作物信息时序图刻画了管理员新增作物的过程：管理员在前端作物管理界面填写新作物信息并提交，前端调用后端的添加作物接口；CropController收到请求后转由CropService处理新增逻辑。CropService首先通过CropRepository查询待添加作物编码是否已存在。如果存在，则返回表示“重复编码”的失败结果；如果不存在，则将新的作物实体 (Crop) 保存到数据库表（如base\_crop表），保存成功后将该作物信息返回前端，前端提示操作成功并刷新作物列表。

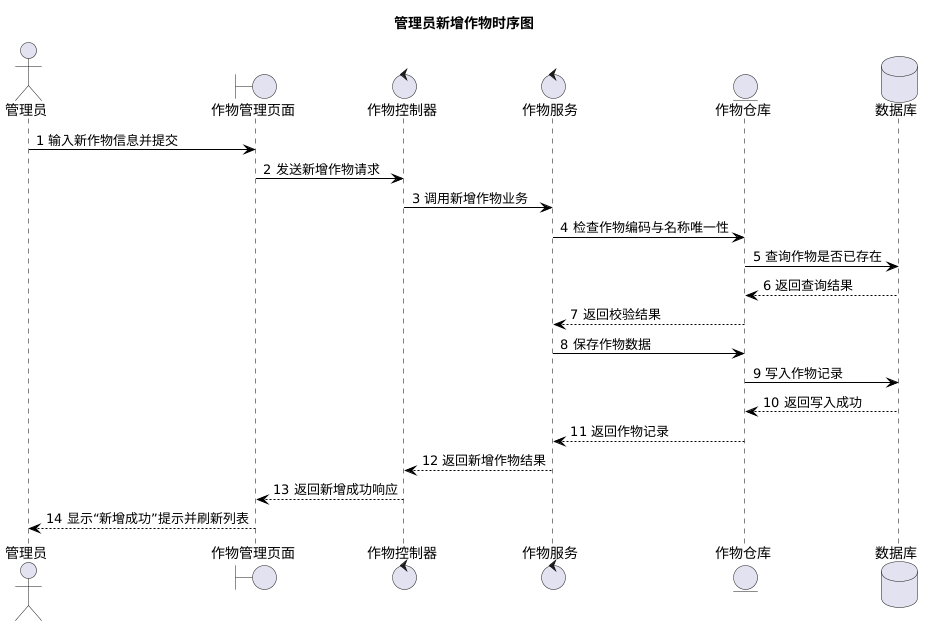


图4-3 添加作物信息时序图

4.1.4新增区域信息时序图

如图4 – 4所示，新增区域信息时序图阐述了管理员新增区域的交互流程，即管理员在区域运作页面填写区域名称，编码等相关信息然后提交，此时RegionController接收该请求，并经由调用RegionService来处理。服务层首先借助RegionRepository验证名称和编码是否存在，若校验无误则保存区域记录并反馈结果，之后，控制器把新增结果告知前端，页面提示新增完成并且更新区域列表。

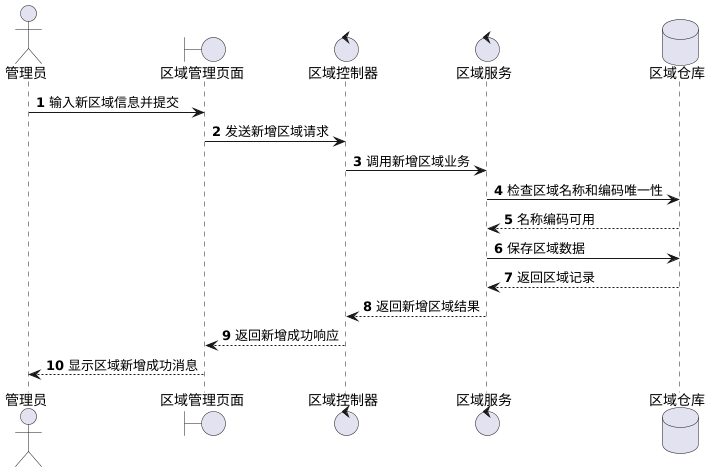


图4-4 新增区域信息时序图

4.1.5 导入历史农作物数据时序图

图4 - 5显示了历史农作物数据导入时序图，其中包含数据异步导入的基本流程，农业专员在数据导入页面上传文件之后，该请求被DataImportController接收，并转交给DataImportService处理。系统会先生成导入任务，并给出任务编号，然后在后台异步执行文件的分析与数据录入工作，在此期间，对于新出现的作物或者地区，系统会自动增添基本资料，而且不断更新任务的状态，等到导入结束以后，任务就会被标记为“已完成”，农业专员能够凭借任务编号来查看导入的结果。

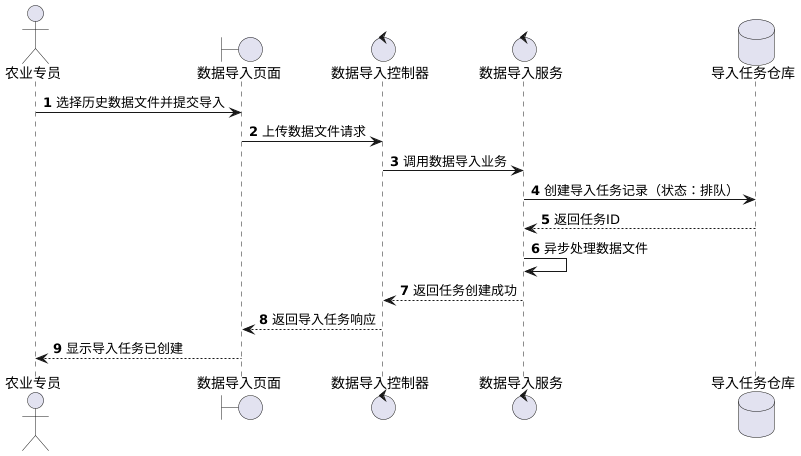


图4-5 导入历史农作物数据时序图

4.1.6配置预测模型时序图

如图4 – 6所示，创建预测模型时序图表明了管理员增添预测模型的流程，即管理员在模型设置页面填写模型名称，类型等相关信息然后提交，ModelController接收到该请求之后会调用ModelService来处理。服务层把模型信息存入模型仓库并且给出保存结果，接着，系统把设置完成的通知传递给前端，这个新模型就能被用于后面的产量预测任务当中。

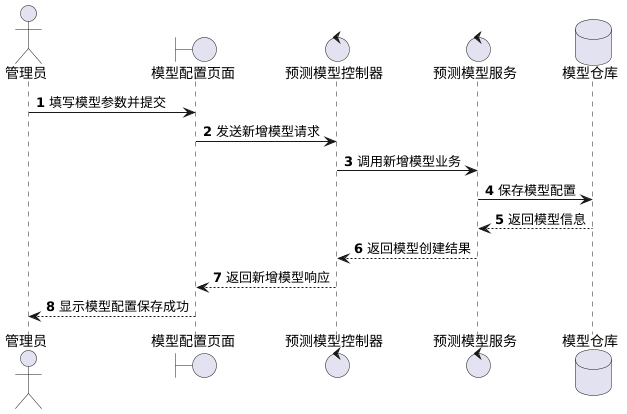


图4-6 配置预测模型时序图

4.1.7执行产量预测时序图

如图4-7所示，执行产量预测时序图刻画了农业专员发起预测任务到系统返回结果的过程：专员在前端预测执行页面选择作物、区域和预测所用模型后提交预测请求。PredictionController收到请求后调用PredictionService进行产量预测处理。PredictionService首先从区域仓库和作物仓库加载所选区域和作物的基础信息，并从产量记录仓库 (YieldRecordRepository) 检索该区域作物的历史产量数据。随后，PredictionService在预测运行仓库 (ForecastRunRepository)中新建一条预测运行记录（状态设为“运行中”），并调用内部集成的预测引擎 (PredictionEngine，封装调用模型计算的组件) 执行模型计算。预测引擎利用历史数据和选定模型计算未来产量的预测结果。服务收到结果后，将预测输出保存到预测结果仓库 (ForecastResultRepository)，并将预测结果数据返回给控制器。PredictionController最终将响应发送给前端页面，前端即时显示预测的结果图表和相关数据供专员查看分析。

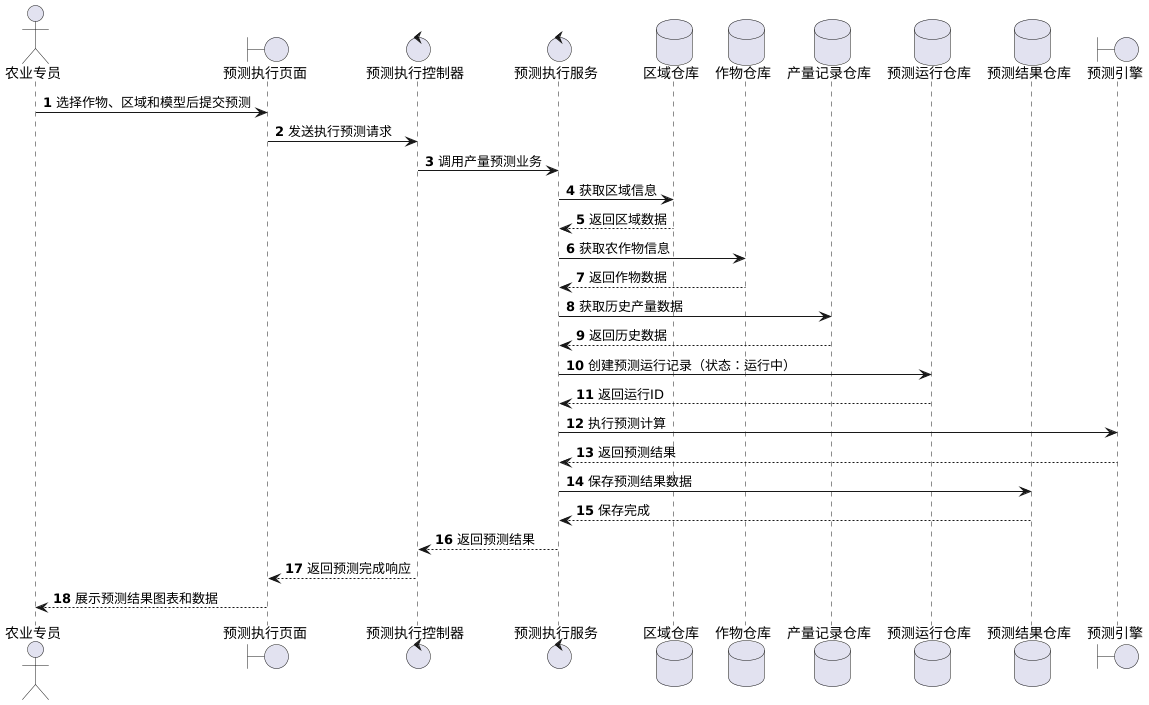


图4-7 执行产量预测时序图

4.1.8 生成分析报告时序图

如图4 – 8所示，生成分析报告时序图阐述了农业专员生成产量分析报告的流程，专员在报告页面选定区域，作物以及时间范围之后提交请求，ReportController收到请求之后会调用ReportService来处理。服务层读取对应的基本信息和历史产量数据，如果有预测结果的话，还会一同加载用来对比分析的数据，接着按照模板生成报告内容并保存到报告仓库当中，系统把报告结果反馈给前端，页面显示出报告并且支持导出保存。

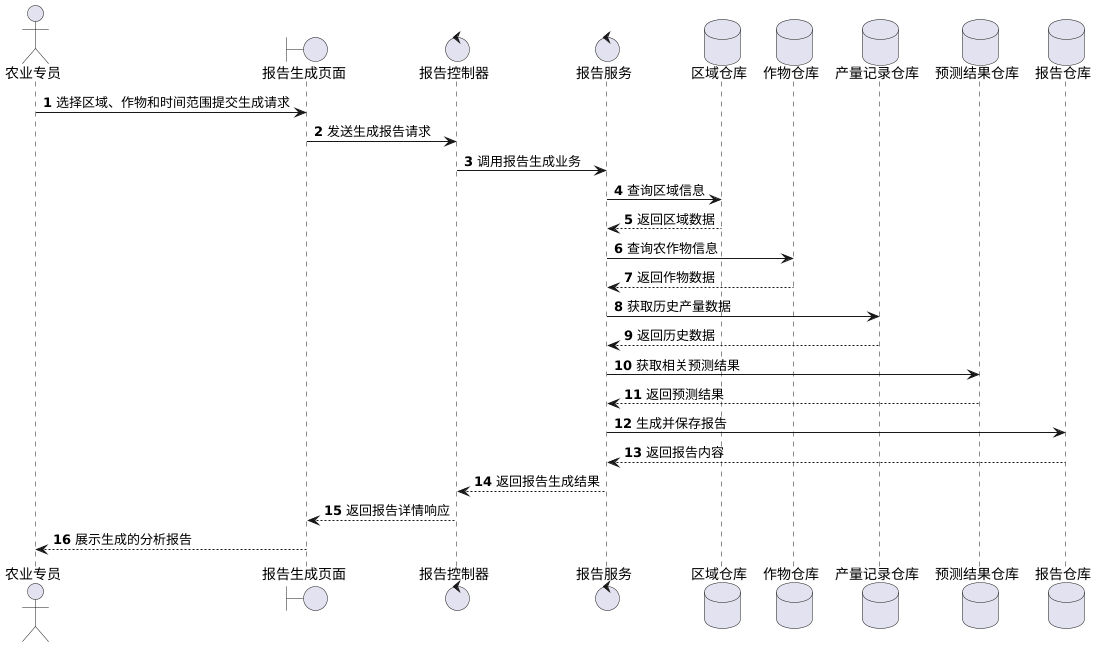


图4-8 生成分析报告时序图

4.1.9 农户提交咨询时序图

如图4-9所示，农户提交咨询时序图描述了农户通过系统提交咨询问题的过程：农户在咨询页面填写咨询的主题和问题描述后提交咨询请求。ConsultationController接收到请求后，调用ConsultationService处理新咨询的创建逻辑。ConsultationService首先根据提交的部门代码验证咨询所属部门的有效性，然后在咨询仓库 (ConsultationRepository)中新建一条咨询记录，状态设为“待回复”，并记录当前农户作为该咨询的发起人。同时，系统为该咨询指定相关领域的农业专家作为参与者（若有可用专家），将专家参与信息保存到参与者仓库 (ParticipantRepository)。接着，ConsultationService将农户提交的首条提问内容保存到消息仓库 (MessageRepository) 中作为咨询会话的第一条消息。完成以上操作后，ConsultationService返回新创建的咨询概要信息给控制器，ConsultationController再将成功创建的结果返回前端页面。前端提示农户咨询提交成功，咨询状态为待回复，农户等待农业专家后续回复。

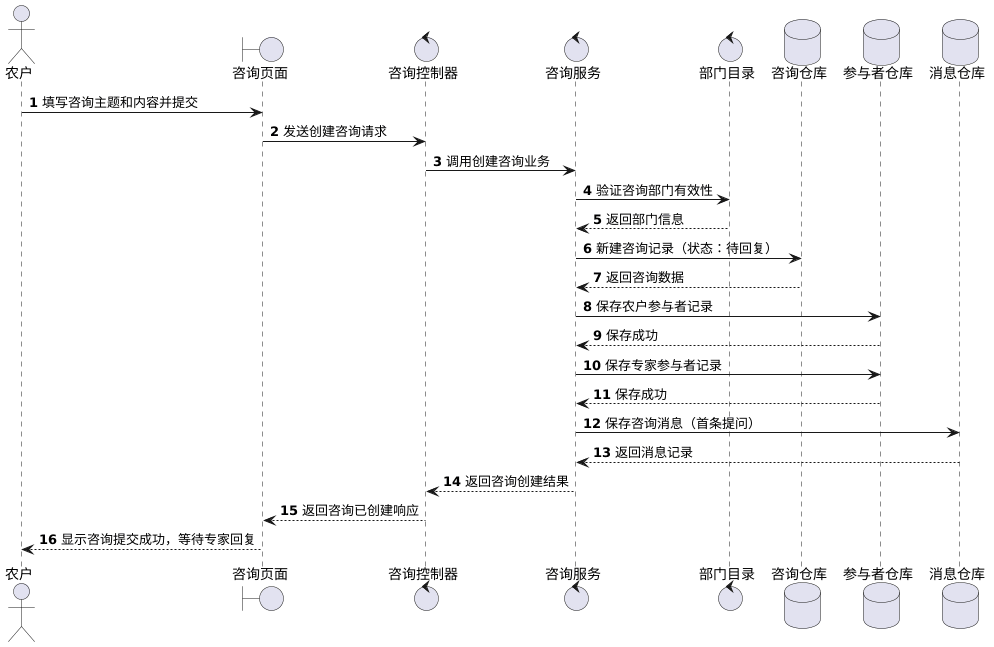


图4-9 农户提交咨询时序图

4.1.10 农业专家回复咨询时序图

如图4-10所示，农业专家回复咨询时序图描述了农业专家针对农户咨询进行回复的过程：专家在咨询页面看到农户的提问后，输入回复内容并点击提交。前端将回复消息发送到ConsultationController，后者调用ConsultationService执行专家回复的业务逻辑。ConsultationService首先从咨询仓库加载该咨询会话的详细信息，确认当前专家有权限回复且会话未关闭。然后将专家的回复内容保存为新的消息记录存入消息仓库。保存成功后，ConsultationService更新参与者仓库 (ParticipantRepository) 中该专家与会话的关联记录（将最后阅读消息的时间更新，表示专家已查看并回复）。完成以上操作后，ConsultationService将回复发送结果返回给控制器，ConsultationController将成功回复的结果返回前端页面。前端随即提示专家回复已成功发送，同时农户用户也会在其界面上收到新的回复内容通知。

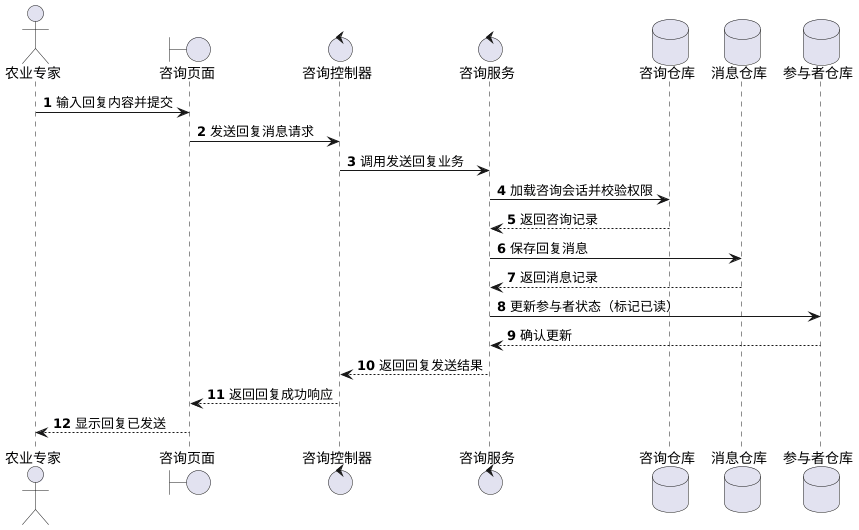


图4-10 农业专家回复咨询时序图

4.2 将系统对象映射到编程环境

4.2.1 用户注册类图

参见图 4 - 11，用户注册流程包含五个关键类，UserController用来接收注册请求，AuthService处理相关的业务逻辑，UserRepository负责用户数据的存储，RoleRepository给出默认的角色信息，User实体用以封装用户的基本数据。

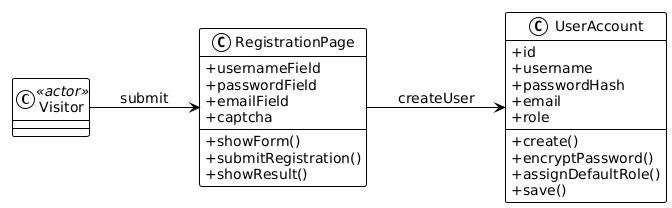


图4-11 用户注册类图

4.2.2 用户登录类图

如图4-12所示，用户登录涉及的类包括AuthController、AuthService、CaptchaService、UserRepository和JwtUtil，用于校验验证码、验证凭证并生成访问令牌。

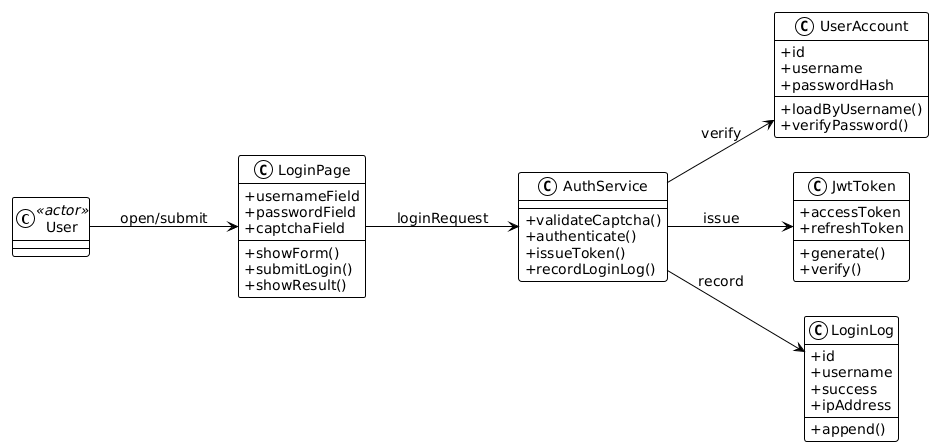


图4-12 用户登录类图

4.2.3 添加作物信息类图

如图4-13所示，添加作物信息功能主要由CropController、CropService、CropRepository、Crop实体类组成，实现作物基本信息的新增与持久化存储。



图4-13 添加作物信息类图

4.2.4添加区域信息类图

如图4-14所示，新增区域信息功能涉及RegionController、RegionService、RegionRepository以及Region实体类，用于维护行政区域的数据。

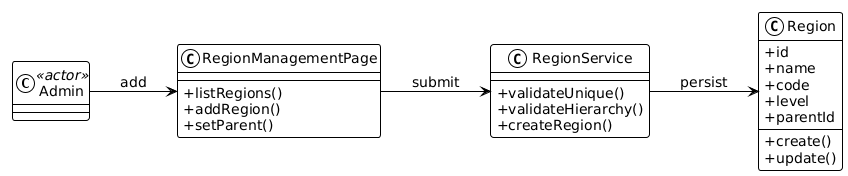


图4-14 添加区域信息类图

4.2.5 导入农作物数据类图

如图4-15所示，历史农作物数据导入功能由DataImportController、DataImportService、CropRepository、RegionRepository等组件协作完成，实现文件解析、数据清洗及入库。

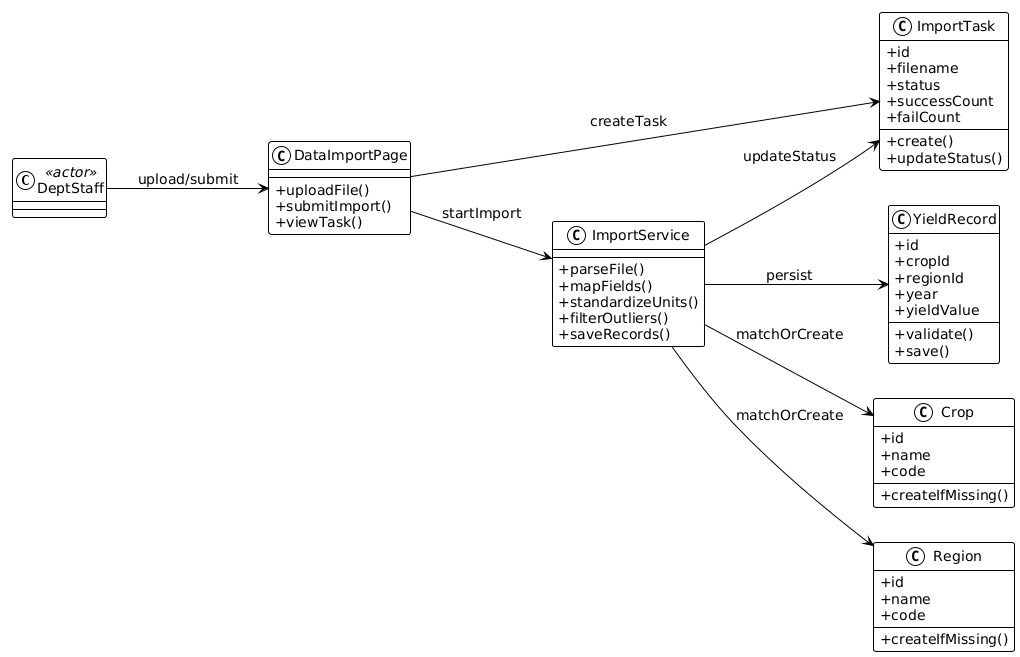


图4-15 导入农作物数据类图

4.2.6 配置预测模型类图

如图4-16所示，配置预测模型功能涉及ModelController、ModelService、ModelRepository和Model实体类，通过提供模型参数持久化配置预测模型。

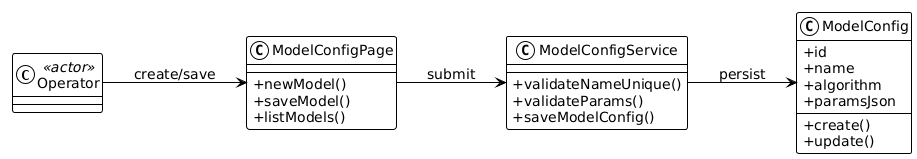


图4-16 配置预测模型类图

4.2.7 执行产量预测类图

如图4-17所示，执行产量预测功能由ForecastController、ForecastService、CropRepository、YieldRecordRepository等类协同完成，获取历史数据并调用预测算法生成结果。

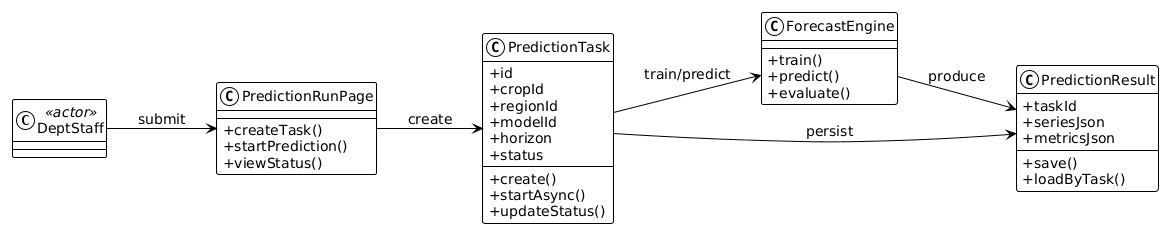


图4-17 执行产量预测类图

4.2.8 生成分析报告类图

如图4-18所示，生成分析报告功能涉及ReportController、ReportService、数据仓库接口等组件，将历史数据和预测结果整合生成可视化的报告实体。

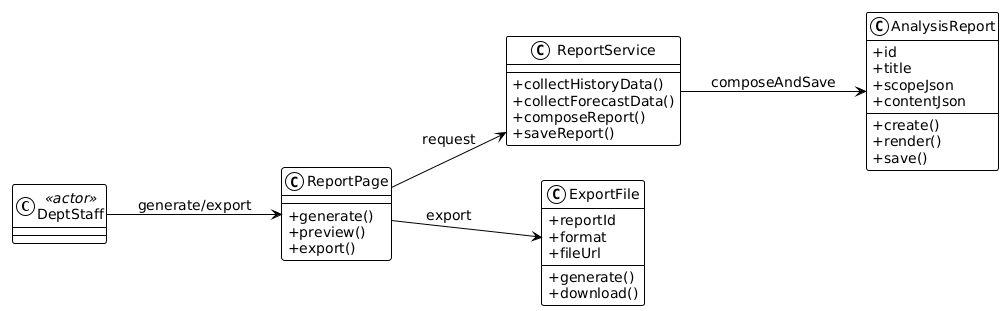


图4-18 生成分析报告类图

4.2.9农户提交咨询类图

如图4-19所示，农户提交咨询功能由ConsultationController、ConsultationService、ConsultationRepository、MessageRepository等类实现，用于创建咨询记录并保存消息内容。

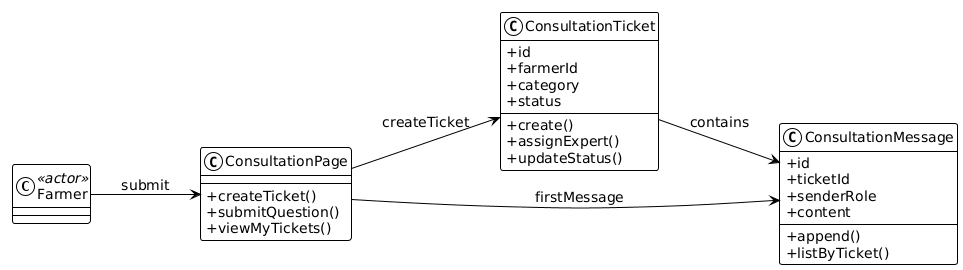


图4-19 农户提交咨询类图

4.2.10专家回复咨询类图

如图4-20所示，农业专家回复咨询功能涉及ConsultationService和MessageService等类，专家通过系统回复农户咨询，系统将回复内容保存并更新咨询状态。

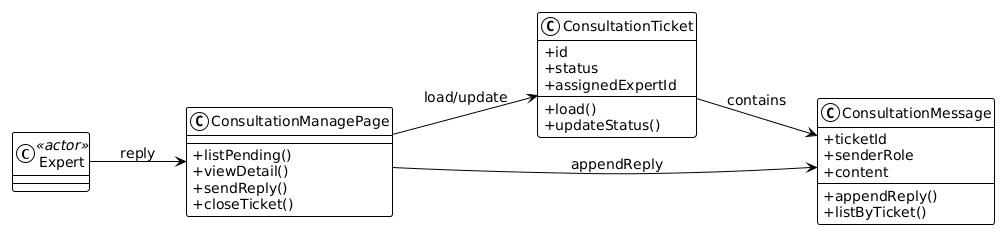


图4-20 专家回复咨询类图

4.3 将持久性对象映射到数据库

4.3.1 用户表

表4-1 用户表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说 明 | 数据类型 | 备注 |
| id | 用户主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 自增 |
| username | 登录用户名 | VARCHAR(64) | NOT NULL；唯一 uq\_user\_username |
| password | 登录密码（加密存储） | VARCHAR(128) | NOT NULL |
| full\_name | 用户姓名/显示名 | VARCHAR(128) | 可为空 |
| email | 邮箱 | VARCHAR(128) | 可为空；唯一 uq\_user\_email |
| department\_code | 部门编码（农业部门用户） | VARCHAR(64) | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.2 区域对象表

表4-2 区域对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 区域主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| code | 区域编码 | VARCHAR(64) | NOT NULL；唯一 uq\_base\_region\_code |
| name | 区域名称 | VARCHAR(128) | NOT NULL |
| level | 行政层级 | VARCHAR(32) | NOT NULL；索引 idx\_base\_region\_level |
| parent\_code | 上级编码 | VARCHAR(64) | 可为空 |
| parent\_name | 上级名称 | VARCHAR(128) | 可为空 |
| description | 描述 | VARCHAR(256) | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.3 作物对象表

表4-3作物对象

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说 明 | 数据类型 | 备注 |
| id | 作物主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| code | 作物编码 | VARCHAR(64) | NOT NULL；唯一 uq\_base\_crop\_code |
| name | 作物名称 | VARCHAR(128) | NOT NULL |
| category | 作物分类 | VARCHAR(64) | 可为空 |
| harvest\_season | 收获季/周期 | VARCHAR(32) | NOT NULL；默认 'ANNUAL' |
| description | 描述 | VARCHAR(256) | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.4 系统配置对象表

表4-4系统配置对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说 明 | 数据类型 | 备注 |
| id | 配置主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| default\_region\_id | 默认区域ID | BIGINT UNSIGNED | 外键→base\_region(id)；ON DELETE SET NULL |
| notify\_email | 通知邮箱 | VARCHAR(128) | 可为空 |
| cluster\_enabled | 聚类功能开关 | TINYINT(1) | NOT NULL；默认 1 |
| pending\_change\_count | 待处理变更数 | INT | NOT NULL；默认 0 |
| security\_strategy | 安全策略 | VARCHAR(64) | 可为空 |
| announcement\_title | 公告标题 | VARCHAR(128) | 可为空 |
| announcement\_message | 公告内容 | VARCHAR(512) | 可为空 |
| announcement\_status | 公告状态 | VARCHAR(32) | NOT NULL；默认 'INACTIVE' |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.5 数据文件对象表

表4-5 数据文件对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 文件主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| name | 文件名称 | VARCHAR(128) | NOT NULL；唯一 uq\_dataset\_file\_name |
| type | 文件类型 | VARCHAR(32) | NOT NULL |
| storage\_path | 存储路径 | VARCHAR(256) | NOT NULL |
| description | 描述 | VARCHAR(256) | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.6 预测模型对象表

表4-6预测模型对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 模型主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| name | 模型名称 | VARCHAR(128) | NOT NULL；唯一 uq\_forecast\_model\_name |
| type | 模型类型 | VARCHAR(32) | NOT NULL（如 ARIMA/LSTM 等） |
| description | 模型描述 | VARCHAR(512) | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.7 报告章节表

表4-7 报告章节

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 章节主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| report\_id | 报告ID | BIGINT UNSIGNED | NOT NULL；外键→report\_summary(id)；ON DELETE CASCADE；索引 idx\_section\_report |
| type | 章节类型 | VARCHAR(64) | NOT NULL |
| title | 章节标题 | VARCHAR(128) | NOT NULL |
| description | 章节描述 | VARCHAR(512) | 可为空 |
| data | 章节数据 | TEXT | 可为空（通常存 JSON/富文本） |
| sort\_order | 排序号 | INT | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.8 在线咨询对象表

表4-8 在线咨询对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 咨询会话主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| subject | 主题 | VARCHAR(128) | NOT NULL |
| crop\_type | 作物类型 | VARCHAR(64) | NOT NULL |
| description | 问题描述 | TEXT | 可为空 |
| status | 状态 | VARCHAR(32) | NOT NULL；默认 'pending'；索引 idx\_consultation\_status |
| priority | 优先级 | VARCHAR(32) | NOT NULL；默认 'normal' |
| department\_code | 部门编码 | VARCHAR(64) | 可为空 |
| created\_by | 发起人用户ID | BIGINT UNSIGNED | NOT NULL；外键→sys\_user(id)；ON DELETE CASCADE；索引 idx\_consultation\_creator |
| assigned\_to | 指派人用户ID | BIGINT UNSIGNED | 可为空；外键→sys\_user(id)；ON DELETE SET NULL；索引 idx\_consultation\_assignee |
| last\_message\_at | 最后消息时间 | DATETIME | 可为空 |
| closed\_at | 关闭时间 | DATETIME | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

4.3.9 咨询消息对象表

表4-9 咨询消息对象表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 说明 | 数据类型 | 备注 |
| id | 消息主键 | BIGINT UNSIGNED AUTO\_INCREMENT | 主键 PK |
| consultation\_id | 咨询会话ID | BIGINT UNSIGNED | NOT NULL；外键→consultation(id)；ON DELETE CASCADE；索引 idx\_message\_consultation |
| sender\_id | 发送者用户ID | BIGINT UNSIGNED | NOT NULL；外键→sys\_user(id)；ON DELETE CASCADE；索引 idx\_message\_sender |
| sender\_role | 发送者角色 | VARCHAR(64) | 可为空 |
| content | 消息内容 | TEXT | 可为空 |
| recalled | 是否撤回 | TINYINT(1) | NOT NULL；默认 0 |
| recalled\_at | 撤回时间 | TIMESTAMP | 可为空 |
| created\_at | 创建时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP |
| updated\_at | 更新时间 | TIMESTAMP | NOT NULL；默认 CURRENT\_TIMESTAMP；ON UPDATE CURRENT\_TIMESTAMP |

第五章 系统实现与测试

5.1 系统实现

5.1.1 公有类实现

本项目在后端采用统一响应与统一异常两类“公有类”来支撑接口返回规范，同时在配置文件中集中维护数据库连接参数；前端则通过分页组件完成数据分页展示，并用消息弹框提示操作结果。

1.数据库连接（配置文件集中管理）：项目通过application.yml统一配置MySQL连接信息，例如：spring.datasource.url/username/password

spring:

datasource:

url: jdbc:mysql://localhost:3306/database-schema?useSSL=false&serverTimezone=Asia/Shanghai&characterEncoding=UTF-8

username: root

password: 123456

driver-class-name: com.mysql.cj.jdbc.Driver

2.用于数据分页（后端分页 + 统一分页返回）：后端控制器通过 PageRequest.of(page,size)发起分页查询

，再用PageResponse(records,total,page,size)返回给前端 ；PageResponse本身是通用分页返回结构

@GetMapping

public ApiResponse<PageResponse<UserResponse>> listUsers(@RequestParam(defaultValue="0") int page,

@RequestParam(defaultValue="10") int size) {

Page<UserResponse>result=userService.listUsers(PageRequest.of(page, size));

return ApiResponse.success(new PageResponse<>(result.getContent(), result.getTotalElements(), page, size));

}

3.用于数据显示（前端表格 + 分页组件渲染）：前端页面使用el-table展示列表数据，并用el-pagination绑定current-page/page-size/total实现分页渲染与翻页事件回调

<el-table :data="users" v-loading="loading">...</el-table>

<el-pagination

:current-page="pagination.page"

:page-size="pagination.size"

:total="pagination.total"

layout="prev, pager, next"

background

@current-change="handlePageChange"

/>

4.用于消息返回弹框（成功/失败提示）：前端统一使用Element Plus的ElMessage/ElMessageBox在关键操作（如删除、请求失败）后进行提示，例如删除成功/失败、参数不合法警告等

if (!ids.length) {

ElMessage.warning('未找到所选数据集的标识，无法删除')

return

}

...

ElMessage.success('已删除选中的数据集')

...

ElMessage.error(error?.response?.data?.message || '删除数据集失败')

5.1.2 用户登录页面实现

页面功能说明

用户登录页面是系统的统一入口，用于完成用户身份认证与权限校验。用户需输入用户名、密码及验证码信息，系统对输入内容进行校验后提交至后端认证接口。认证成功后，系统生成登录凭证并跳转至系统首页；认证失败时，页面给出相应的错误提示。该页面为系统的安全访问控制提供基础保障。登录页面界面如图5-1所示。



图5-1 登录页面界

关键代码实现（后端 + 前端）

（1）登录接口控制器

@PostMapping("/login")

public ApiResponse<LoginResponse> login(@RequestBody LoginRequest request) {

return ApiResponse.success(authService.login(request));

}

（2）登录业务核心逻辑

public LoginResponse login(LoginRequest request) {

Authentication authentication = authenticationManager.authenticate(

new UsernamePasswordAuthenticationToken(

request.getUsername(), request.getPassword())

);

SecurityContextHolder.getContext().setAuthentication(authentication);

String token = jwtTokenProvider.createToken(authentication);

return new LoginResponse(token);

}

（3）前端登录请求

export function login(data) {

return request.post('/api/auth/login', data)

}

实现说明

登录页面通过调用后端认证接口完成用户身份校验，系统基于 JWT 实现无状态登录控制。

5.1.3 仪表盘页面实现

页面功能说明

系统首页以仪表盘形式展示系统运行概况，包括数据规模、作物/区域数量及分析结果摘要等，帮助用户快速掌握系统状态并进入其他功能模块。系统首页展示效果如图5-2所示。

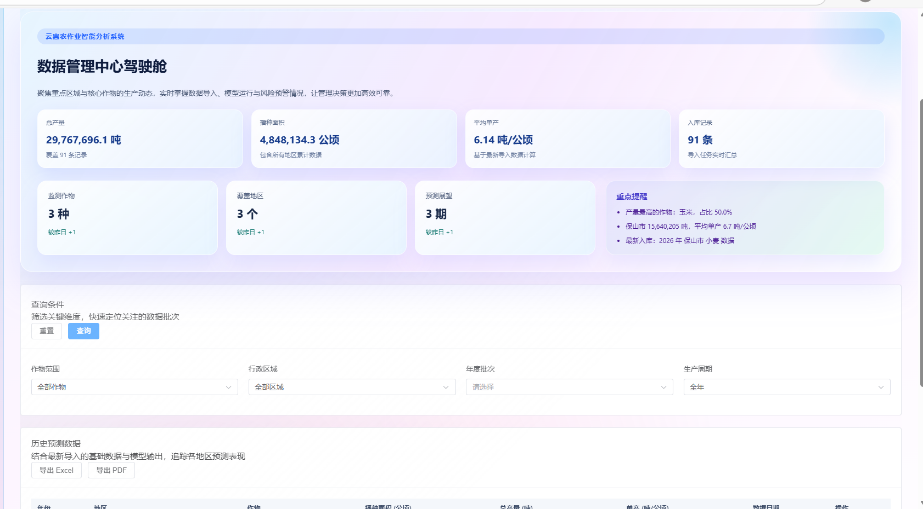


图5-2 仪表盘页面

关键代码实现

（1）仪表盘数据接口

@GetMapping("/dashboard/summary")

public ApiResponse<DashboardSummaryResponse> summary() {

return ApiResponse.success(dashboardService.getSummary());

}

（2）仪表盘数据汇总逻辑

public DashboardSummaryResponse getSummary() {

long cropCount = cropRepository.count();

long regionCount = regionRepository.count();

long recordCount = yieldRecordRepository.count();

return new DashboardSummaryResponse(cropCount, regionCount, recordCount);

}

（3）前端页面加载调用

onMounted(() => {

fetchDashboardSummary().then(res => {

dashboard.value = res.data

})

})

实现说明

系统首页通过统一接口获取统计信息，并以图表方式展示系统运行概况。

5.1.4 数据中心页面实现

页面功能说明

数据中心用于数据文件上传、数据集管理以及导入任务状态查看。系统采用异步任务方式处理数据导入，避免大文件解析造成页面阻塞，并在页面中展示导入进度与结果摘要。数据中心页面界面如图5-3所示。

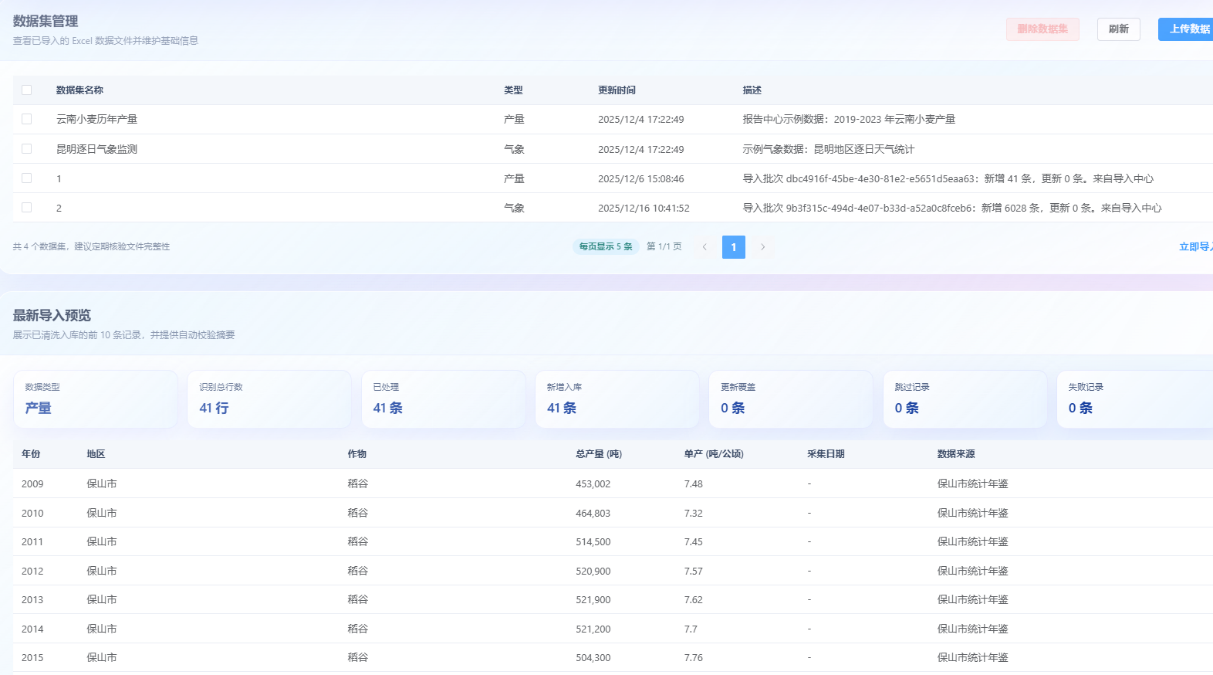


图5-3 数据中心页面

关键代码实现

（1）数据上传接口

@PostMapping("/data-import/upload")

public ApiResponse<DataImportJobView> upload(@RequestParam MultipartFile file) {

return ApiResponse.success(dataImportService.submitImport(file));

}

（2）导入任务创建

public DataImportJobView submitImport(MultipartFile file) {

DataImportJob job = new DataImportJob();

job.setStatus("RUNNING");

jobRepository.save(job);

processAsync(job.getId(), file);

return DataImportJobView.from(job);

}

（3）前端上传文件

const formData = new FormData()

formData.append('file', file)

request.post('/api/data-import/upload', formData)

实现说明

系统采用异步方式处理数据导入，避免大文件上传造成页面阻塞。

5.1.5 在线咨询管理页面实现

页面功能说明

在线咨询管理页面用于处理用户提交的农业咨询信息。普通用户可提交咨询问题，管理员和农业专员可在该页面查看咨询内容并进行回复处理，实现在线互动交流。在线咨询管理页面如图5-4所示。

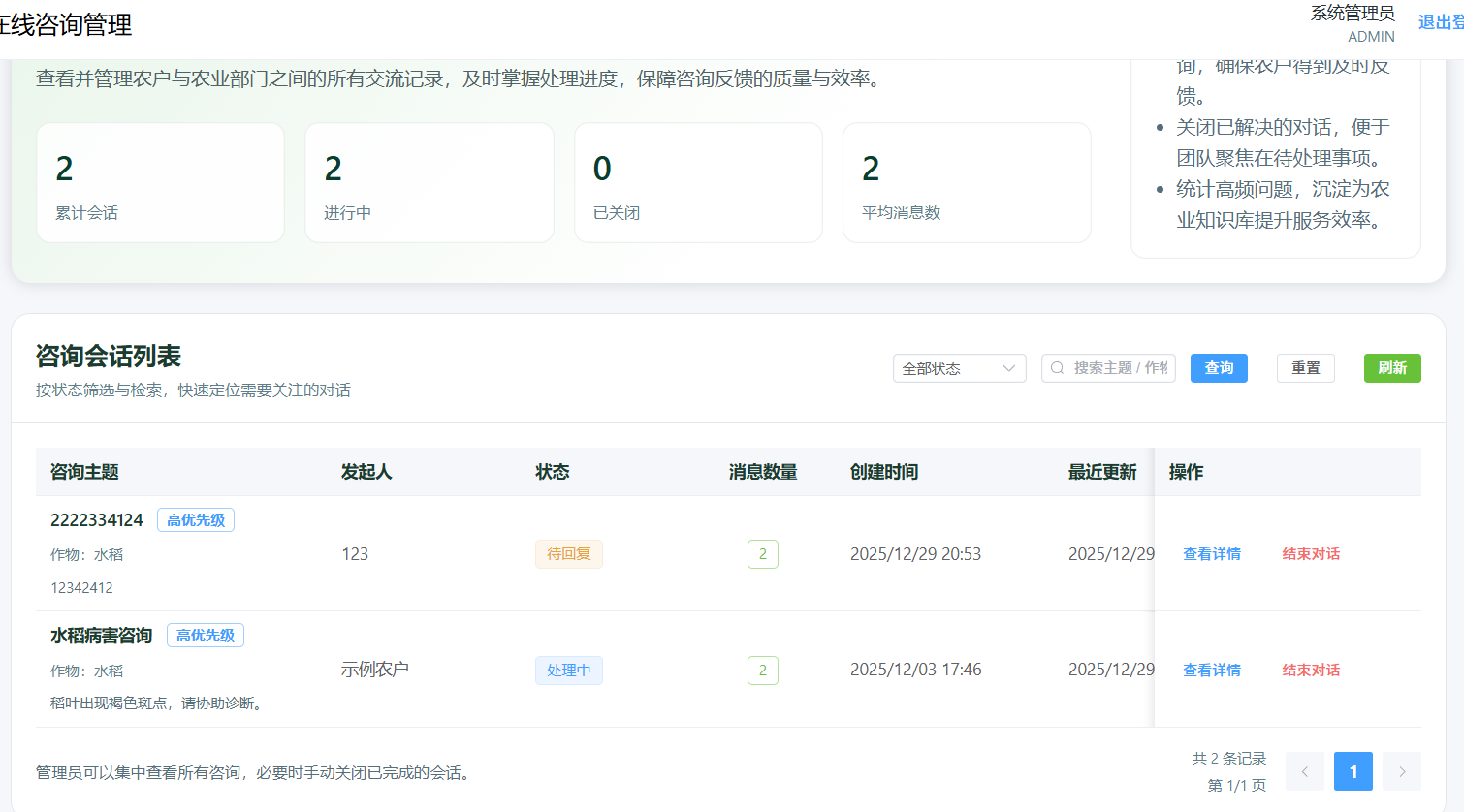


图5-4 在线咨询管理页面

关键代码实现

（1）提交咨询接口

@PostMapping("/consultation")

public ApiResponse<Void> submit(@RequestBody ConsultationRequest request) {

consultationService.submit(request);

return ApiResponse.success();

}

（2）查询咨询列表

@GetMapping("/consultation")

public ApiResponse<List<Consultation>> list() {

return ApiResponse.success(consultationService.listAll());

}

（3）前端提交咨询

request.post('/api/consultation', form)

实现说明

系统通过咨询管理模块实现用户与管理人员之间的在线交流，提高农业信息服务的及时性。

5.1.6 报告中心页面实现

页面功能说明

报告中心用于生成并展示农作物产量分析报告。用户选择作物、区域与时间范围后，系统加载对应数据并展示分析结果（表格/图表），支持按条件进行对比查看。报告中心页面如图5-5所示。

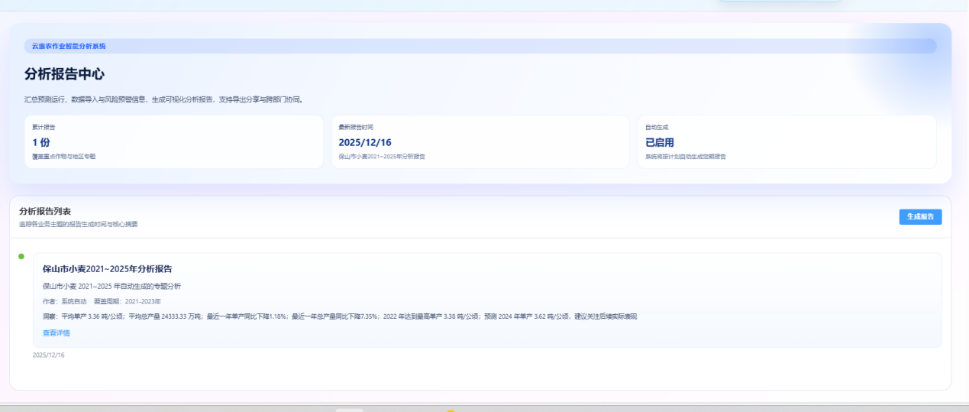


图5-5 报告中心页面

关键代码实现

（1）获取作物列表（用于条件选择）

@GetMapping("/base/crops")

public ApiResponse<List<Crop>> listCrops() {

return ApiResponse.success(cropService.listAll());

}

（2）前端加载作物下拉框

fetchCrops().then(res => {

cropOptions.value = res.data

})

（3）提交报告生成条件

request.post('/api/report/generate', {

cropId, regionId, startYear, endYear

})

实现说明

报告中心通过组合条件查询接口，实现多维度产量分析。

5.1.7 系统设置页面实现

页面功能说明

系统设置页面负责设置系统运行的基本参数，区域经营模块具备增加，修改和删除区域信息的功能，这会给后续的数据导入，统计分析以及产量预测给予区域方面的支撑，系统设置页面见图5 - 6，区域经营模块的界面见图5 - 7。



图5-6 系统设置页面

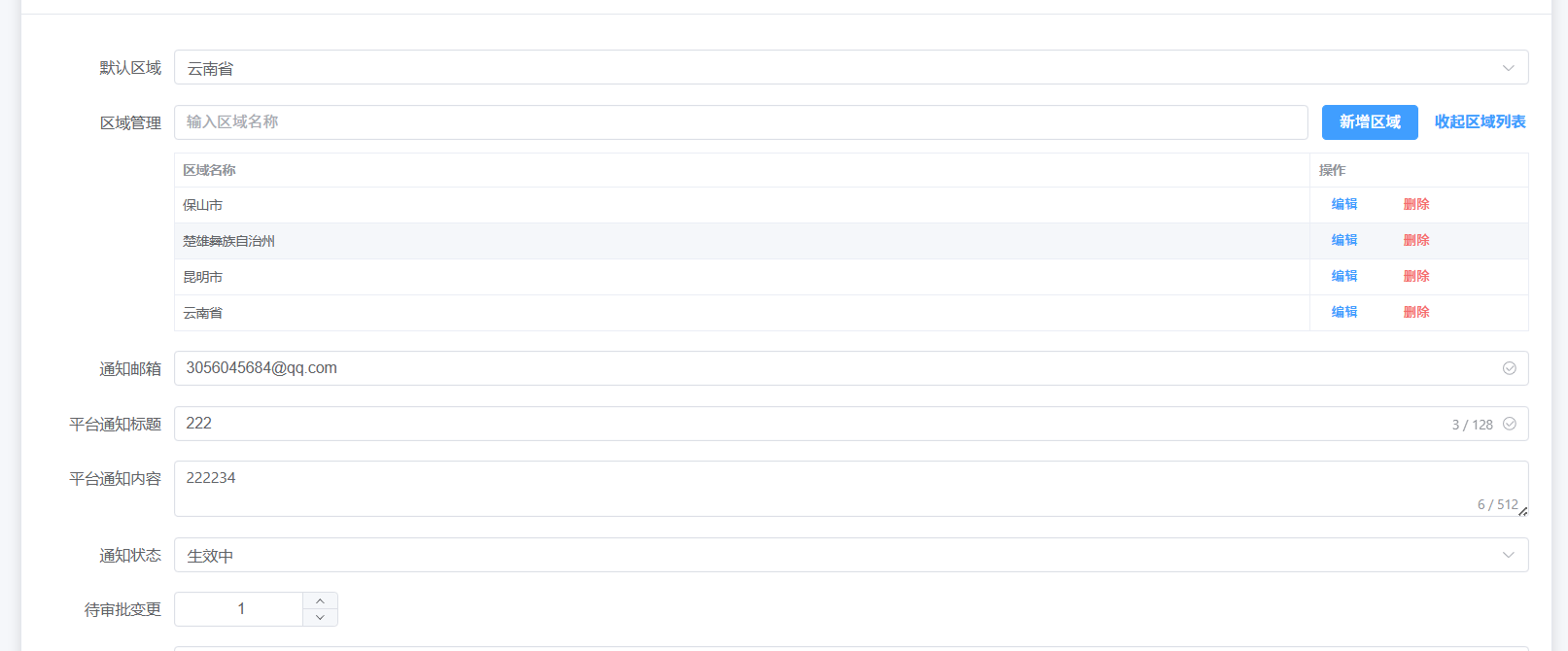


图5-7 区域管理模块界面

关键代码实现

（1）查询区域

@GetMapping("/base/regions")

public ApiResponse<List<Region>> list() {

return ApiResponse.success(regionService.listAll());

}

（2）新增区域

@PostMapping("/base/regions")

public ApiResponse<Region> create(@RequestBody Region region) {

return ApiResponse.success(regionService.create(region));

}

（3）前端提交新增/修改

request.post('/api/base/regions', form)

实现说明

系统设置页面实现了基础区域数据的集中维护。

5.1.8 用户管理页面实现

页面功能说明

用户管理页面用于维护系统用户账号信息（新增、修改、删除及角色分配），仅管理员角色可访问。通过该页面可实现对系统权限的统一管理与安全控制。用户管理页面如图4-9所示。



图5-8 用户管理页面

实现说明

用户管理模块通过角色控制限制访问权限，确保系统安全。

5.1.9 系统日志管理页面实现

页面功能说明

日志管理页面用于记录和查询系统运行日志与用户操作日志。管理员可通过该页面查看用户登录情况和关键操作记录，便于系统运维与安全审计。日志管理页面展示效果如图5-9所示。

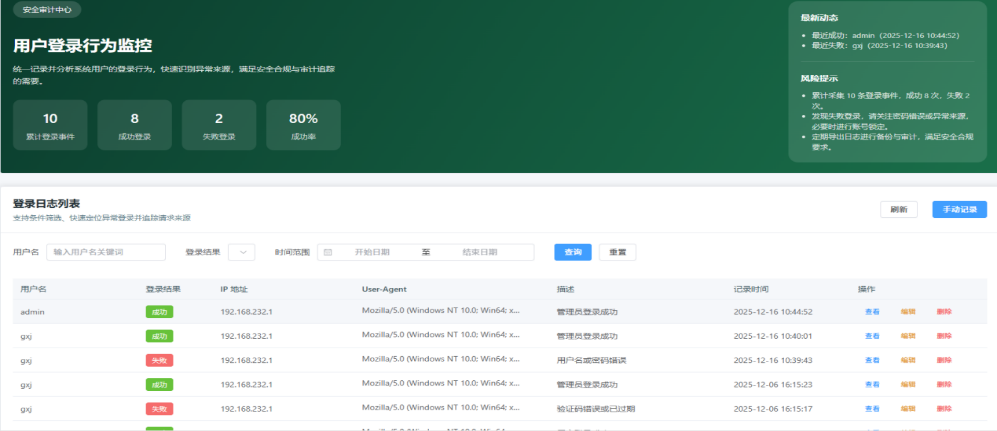


图5-9 系统日志管理页面

关键代码实现

（1）日志记录示例

public void record(String username, String action) {

SysLog log = new SysLog(username, action, LocalDateTime.now());

sysLogRepository.save(log);

}

（2）日志查询接口

@GetMapping("/system/logs")

public ApiResponse<List<SysLog>> listLogs() {

return ApiResponse.success(sysLogService.listAll());

}

（3）前端加载日志

onMounted(() => {

loadLogs()

})

实现说明

系统在关键操作节点记录日志信息，并通过日志管理页面集中展示，为系统安全和问题追踪提供支持。

5.1.10 预测中心页面实现

页面功能说明

预测中心页面可用来启动农作物产量预测，用户在此页面选定作物，区域等条件并提交之后，系统就会调用相关的预测模型来对历史数据执行运算，从而得出将来的产量预测结果，然后以图表的形式表现出来，预测中心页面的效果见图 5 - 10，而预测结果的显示情况则见图 5 - 11。



图5-10 预测中心页面

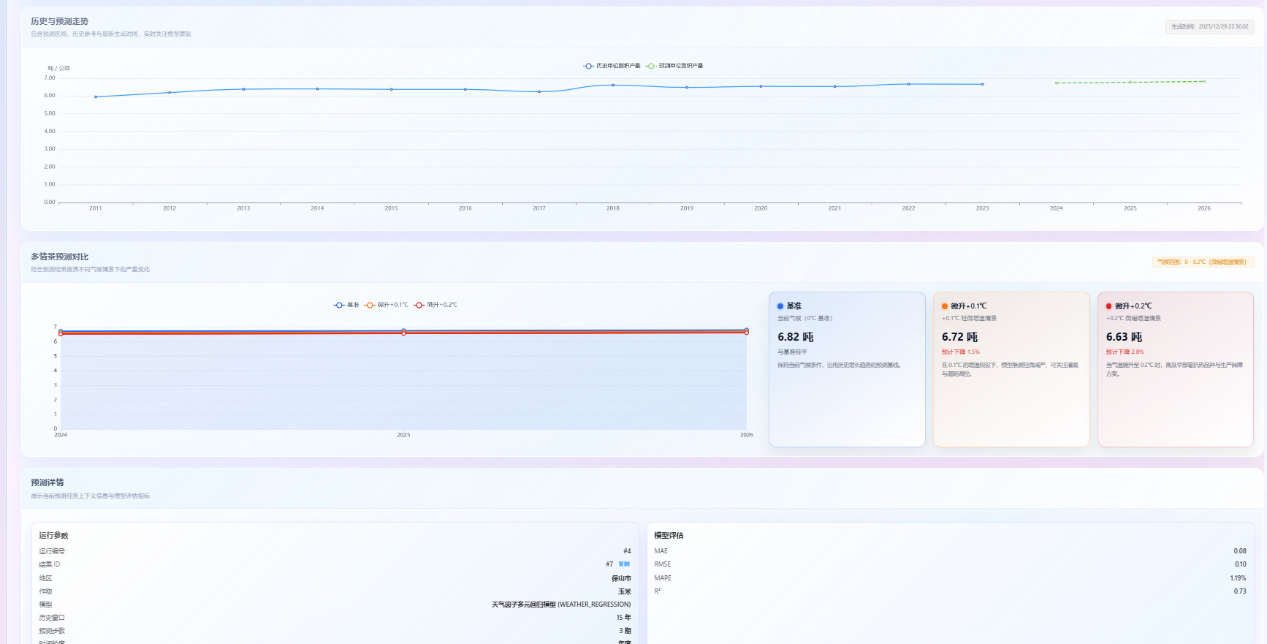


图5-11 预测结果页面

关键代码实现

（1）预测接口

@PostMapping("/forecast/predict")

public ApiResponse<ForecastExecutionResponse> predict(

@RequestBody ForecastExecutionRequest request) {

return ApiResponse.success(forecastService.runForecast(request));

}

（2）预测服务核心逻辑

public ForecastExecutionResponse runForecast(ForecastExecutionRequest req) {

List<YieldRecord> history = yieldRecordRepository

.findByCropIdAndRegionId(req.getCropId(), req.getRegionId());

double[] series = history.stream()

.mapToDouble(YieldRecord::getProduction).toArray();

double[] forecast = forecastEngine.predict(series);

return new ForecastExecutionResponse(forecast);

}

（3）前端发起预测

request.post('/api/forecast/predict', form).then(res => {

forecastResult.value = res.data

})

实现说明

预测中心将模型计算封装在服务层，前端仅负责参数配置和结果展示。

5.1.11 数据可视化页面实现

页面功能说明

数据可视化页面负责显示并分析已导入的产量数据，用户可遵照作物，区域以及时间范围实施筛选，系统会以图表形式表现产量变化趋势，区域对比等结果，这样就能方便快捷地掌握数据特点，数据可视化页面的界面见图5 - 12。



图5-12 数据可视化页面

关键代码实现

（1）前端路由配置

{

path: 'visualization',

name: 'visualization',

component: () => import('../views/VisualizationCenterView.vue'),

meta: { requiresAuth: true }

}

（2）前端图表数据加载

onMounted(() => {

loadVisualizationData()

})

（3）后端数据统计接口

@GetMapping("/analysis/summary")

public ApiResponse<AnalysisSummaryResponse> summary() {

return ApiResponse.success(analysisService.getSummary());

}

实现说明

系统通过后端接口对历史产量数据进行统计汇总，前端利用图表组件将统计结果进行可视化展示，实现数据分析结果的直观呈现。

5.1.12 天气监测页面实现

页面功能说明

气象分析页面用于对历史气象数据进行统计分析，并探索气象因素与农作物产量之间的关系。页面以图表方式展示气象要素变化趋势，为农业产量分析提供环境因素参考。气象分析页面界面如图5-13所示。

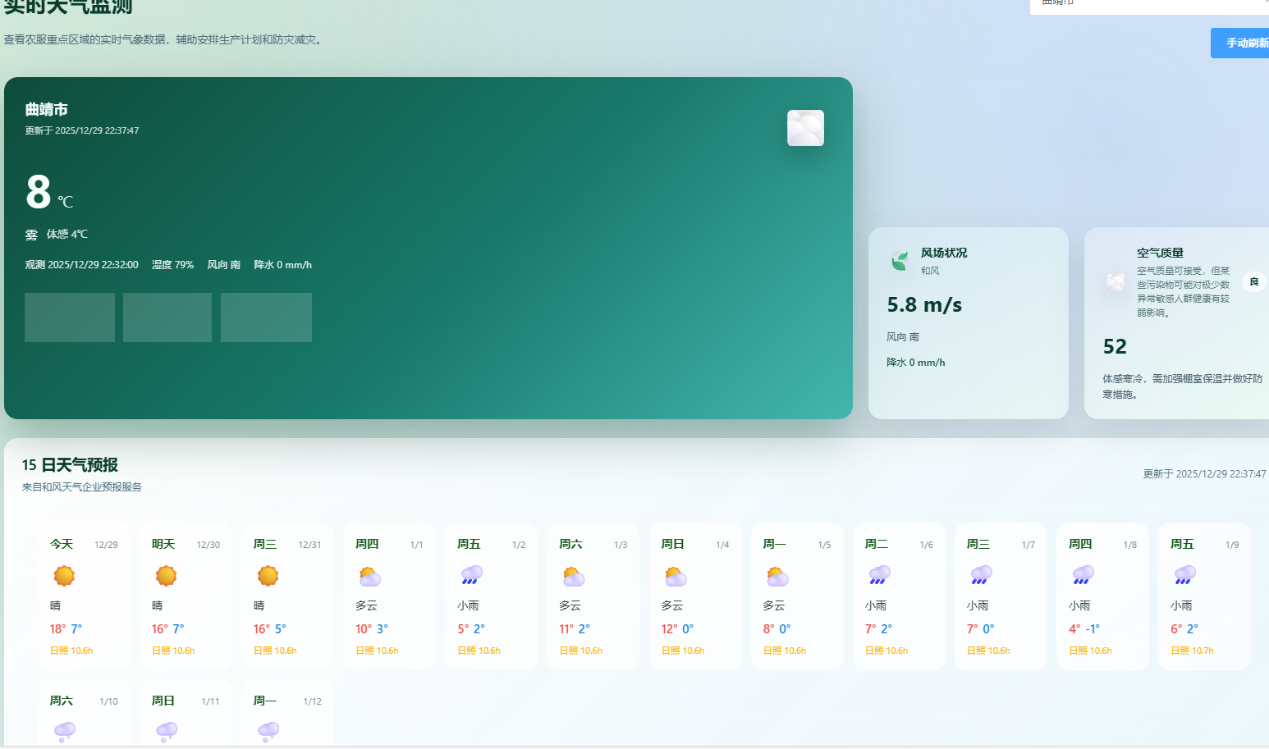


图5-13 天气监测页面

关键代码实现

（1）气象分析接口

@GetMapping("/weather/analysis")

public ApiResponse<WeatherAnalysisResponse> analyze(

@RequestParam Long regionId) {

return ApiResponse.success(weatherAnalysisService.analyze(regionId));

}

（2）前端加载气象分析结果

onMounted(() => {

loadWeatherAnalysis()

})

（3）分析结果图表渲染

watch(analysisData, () => {

renderChart(analysisData.value)

})

实现说明

系统对气象数据进行统计分析后，将分析结果以折线图等形式展示，为产量预测提供辅助参考。

5.1.13 气象分析页面实现

页面功能说明

气象分析页面用于对历史气象数据进行统计分析，并探索气象因素与农作物产量之间的关系。页面以图表方式展示气象要素变化趋势，为农业产量分析提供环境因素参考。气象分析页面界面如图5-14所示。

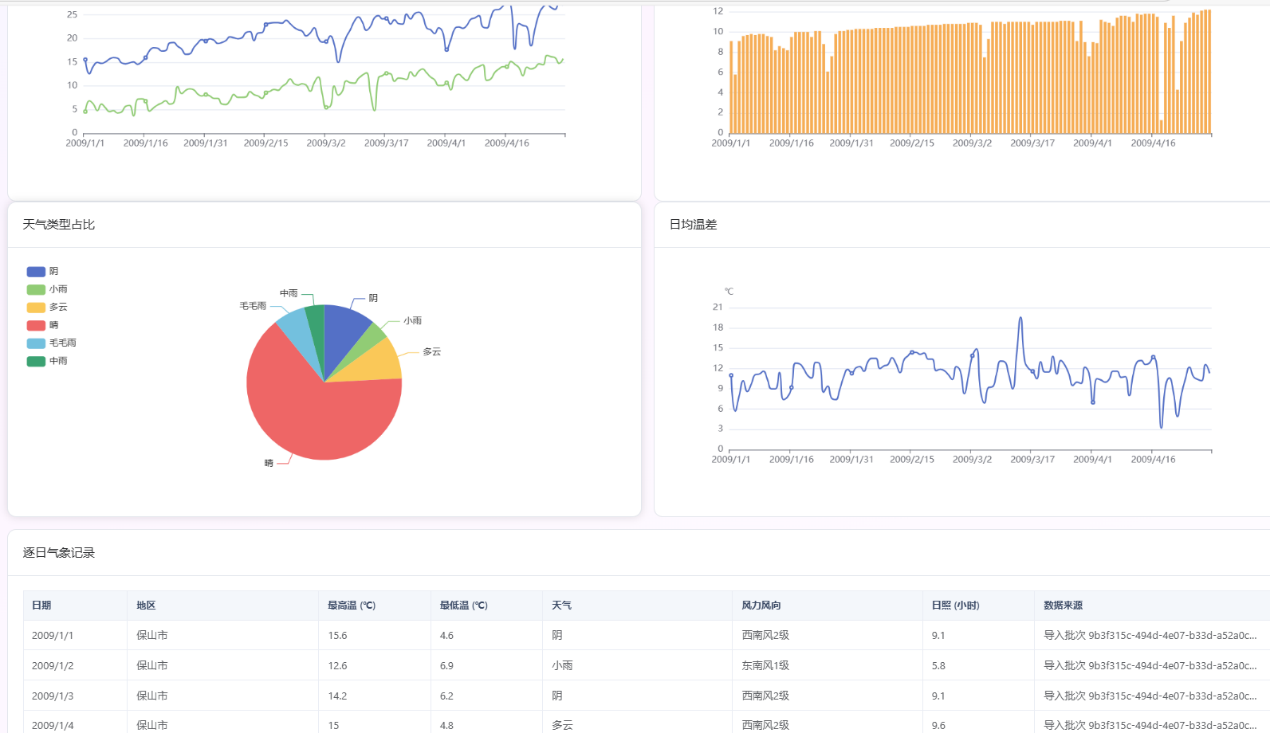


图5-14 气象分析页面

关键代码实现

（1）气象分析接口

@GetMapping("/weather/analysis")

public ApiResponse<WeatherAnalysisResponse> analyze(

@RequestParam Long regionId) {

return ApiResponse.success(weatherAnalysisService.analyze(regionId));

}

（2）前端加载气象分析结果

onMounted(() => {

loadWeatherAnalysis()

})

（3）分析结果图表渲染

watch(analysisData, () => {

renderChart(analysisData.value)

})

实现说明

系统对气象数据进行统计分析后，将分析结果以折线图等形式展示，为产量预测提供辅助参考。

5.2 系统测试

5.2.1 测试方法

1.白盒测试

白盒测试侧重检查系统内部实现是否按预期运行，重点覆盖控制层、业务层、异常处理与数据访问等关键路径的分支与边界情况。结合本项目 Spring Boot 后端，测试内容主要包括：

（1）控制层参数校验是否能被 @Valid 正确触发，并在失败时返回对应的校验异常（如 MethodArgumentNotValidException）；

（2）业务层核心分支是否覆盖到位，例如作物编码唯一性判断（存在/不存在）以及保存成功或失败等情况；

（3）全局异常处理是否能将不同类型异常映射为统一、规范的错误响应（如校验异常返回 400，业务异常按 ResultCode 返回相应状态码，响应体符合 ApiResponse.failure(...) 约定）；

（4）数据库约束和业务校验需保持一致，如此才能防止并发时产生重复数据，其考察可依循 JUnit5 结合 Mockito，MockMvc（或者 SpringBootTest）来开展，而语句覆盖和分支覆盖则属于主要参照标准。

2.黑盒测试

黑盒测试从用户使用角度出发，重点检查功能是否可用、流程是否顺畅以及异常提示是否符合预期，不涉及内部实现细节。本项目黑盒测试主要覆盖：

（1）注册、登录、作物管理等核心流程能否正常完成；

（2）前端校验与后端 @Valid 校验是否一致，提示信息是否清楚（如必填、长度限制、编码重复等）；

（3）前后端接口交互是否符合约定，包括请求参数与统一响应格式 ApiResponse；

（4）在网络异常、参数缺失、重复提交或非法输入等情况下，系统能否稳定返回可理解的错误信息；

（5）不同角色的访问控制是否生效，例如无权限用户进入作物管理页面时是否被拦截并提示。测试可通过页面实际操作配合 Postman/Apifox 调用接口完成。

5.2.1 测试用例

**用例1：用户注册测试**

**前置条件**：管理员/专员未登录系统；数据库中不存在用户名为“testuser”的账号。

**执行步骤**：

1.打开注册页面；

2.输入用户名“testuser”、密码“Pass1234”、确认密码“Pass1234”、邮箱“test@example.com”；

3.点击“注册”；

**预期输出**：

1.系统提示注册成功，自动跳转至登录页面；

2.数据库User表新增用户名为“testuser”的记录，密码已加密存储，邮箱为“test@example.com”，角色为默认普通用户角色。

**测试结果**：实际与预期一致，注册功能正常。

**用例2：登录测试**

**前置条件**：数据库中存在用户名为“admin”、密码为“Admin123”的管理员账号，且已启用。

**执行步骤**：

1.打开登录页面；

2.输入用户名“admin”、密码“Admin123”、正确的验证码；

3.点击“登录”；

**预期输出**：

1.页面跳转至系统主界面，显示管理员欢迎信息；

2.返回的JWT令牌有效，后续访问受限接口时可通过认证；

**测试结果**：实际与预期一致，登录功能正常。

**用例3：作物新增测试**

**执行步骤**：

1.点击“新增作物”；

2.填写名称“玉米”、编码“C001”、分类“粮食”、生长期“120天”、选择区域“云南省”；

3.点击“提交”；

**预期输出**：

1.系统提示新增成功，作物列表出现名称为“玉米”的记录；

2.数据库base\_crop表新增编码“C001”、名称“玉米”的作物数据，区域关联正确。

**测试结果**：实际与预期一致，新增作物功能正常。

**用例4：预测任务创建与结果展示测试**

**执行步骤：**

1.进入“预测中心”，点击“新建预测任务”；

2.选择作物、区域，设置预测周期/预测年份，并在模型下拉框中选择一个已配置模型；

3.点击“提交/开始预测”；

4.等待任务执行完成，进入预测结果页面或历史预测列表查看结果。

**预期输出：**

1.系统创建预测任务并显示任务状态（如执行中/已完成/失败）；

2.预测完成后，结果页面可正常展示预测曲线与历史对比信息；

3.历史预测列表新增一条记录，状态与结果可查询；

4.后端接口返回成功响应：POST /api/forecast/predict 能返回预测结果数据。

**测试结果：** 实际与预期一致，预测任务创建与结果展示功能正常。

**用例5：系统日志权限控制测试**

**执行步骤：**

1.使用非管理员账号登录系统；

2.尝试进入“日志管理”页面或直接访问日志查询接口；

3.退出后使用管理员账号登录；

4.再次进入“日志管理”页面查看日志列表。

**预期输出：**

1.非管理员访问日志功能被拦截，系统提示无权限或返回 403；

2.管理员可正常进入日志管理页面并看到日志列表数据；

3.日志查询接口返回数据结构正常（分页/列表信息完整）。

**测试结果：** 实际与预期一致，日志权限控制生效。

5.3 测试结果

通过对前述测试用例的实际执行，可以验证系统各主要功能均能够按照设计要求正常运行，整体功能实现效果良好，未发现影响系统使用的严重缺陷。

在功能测试过程中，首先对用户注册与登录模块进行了验证。测试结果显示，用户在完成注册操作后，系统能够正确给出成功提示并引导用户进入登录流程；管理员登录后，系统首页仪表盘能够正常加载，各功能菜单显示完整，统计数据与实际数据一致。相关测试界面截图如图5-15和图5-16所示。

针对基础数据管理，在作物新增测试中，系统能够对名称与编码进行重复校验；新增成功后，页面列表立即出现新记录，且数据库写入正确。图5-17为新增成功后的页面状态。

在预测功能测试中，用户按条件提交后系统可正常生成预测任务并在后台完成计算；任务状态随执行过程更新，完成后页面能够展示预测曲线及与历史数据的对比结果，同时在历史任务列表中可查询到该记录。相关界面如图5-18所示。

权限控制方面，我们对日志访问做了验证，管理员能够正常查看日志列表及操作记录，界面见图5 - 19，这表明基于角色的访问控制起作用，在异常与边界场景下，比如编码重复，参数缺失，格式错误等情况时，系统都会给出清晰的提示，没有发生卡顿或者崩溃现象；针对预测和数据处理等比较复杂的流程执行多次重复试验之后，系统仍然可以稳定地产生结果，总体来讲，系统在功能达成，异常处理以及权限控制上符合设计需求，能够支持产量分析和预测的实际应用，要量化预测精度就要用均方误差（MSE）作评价指标，它的计算公式如下：

(5-1)

式中:

——均方误差；

—— 预测样本数；

—— 第 个预测值；

—— 第 个实际值。



图5-15 用户注册成功



图5-16 数据统计面板



图5-17 新增记录成功





图5-18 预测功能测试

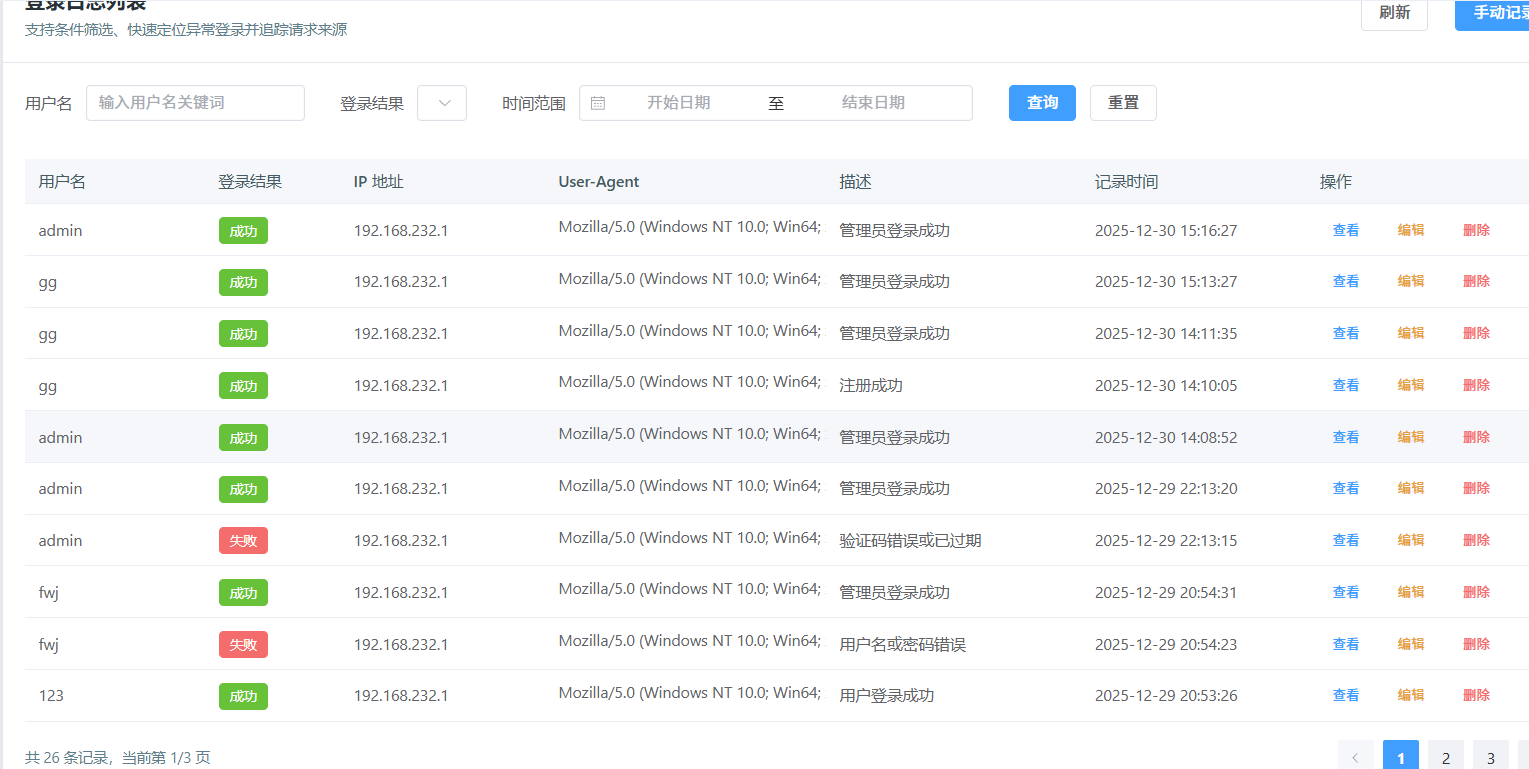


图5-19 系统日志权限控制测试

5.4 测试小结

系统测试结果反映，这个项目实现的平台整体运行比较平稳，用户管理、基础数据维护、数据带进、预测分析、报告与仪表盘这些核心模块在单项测试和集成测试里都达到了预想。对于必填项缺失、输入超限、重复提交这类常见异常场景，系统能够给出清楚提示，没有出现服务异常或页面无响应情况，说明输入校验与异常处理机制比较完善。

测试过程也显示出一些可以改进的地方：在并发场景里（比如同时新增相同作物）仍然比较依赖数据库唯一约束来挡住冲突，后面可以借助更合理的并发控制与提示机制来改善体验；还有，前后端校验规则需要进一步统一，减少“前端没拦住、后端报错”的现象。后续也能补充压力测试与更细粒度的权限验证，用来评估大数据量、多用户访问下的性能与安全表现。

整体来说，平台已经基本达到设计目标，能够支持数据带进、预测计算与可视化展示这些关键流程。上线前可以结合上面提到的改进点做进一步优化，以提升在生产环境里的稳定性与可用性。

第六章 总 结

6.1 工作总结

此阶段已达成系统核心功能的开发及验证，其中包含用户注册登录，基本数据守护，数据导入清洗，模型设置与预测执行，结果可视化以及在线咨询服务等模块，该系统以前后端分离的方式达成，其模块界限分明，接口调用也正常，经由白盒和黑盒检测可知，系统在常见和异常情况下均可稳定运行，其整体效果符合预期目的。

6.2 不足总结

农作物产量预测经营系统尚有改良之处，其预测结果的历史对比功能较为缺乏，报表导出形式也较少，在大数据导入及并发预测时，其性能仍需改善，可以采用异步队列与缓存来提升效率。前端交互及操作引导部分还可继续完善，系统中的模型类型比较单一，日后可增添更多算法以增强其适应性和效果。

6.3 展望与未来

后续工作主要包括：根据实际使用反馈迭代功能如接入实时气象数据和拓展移动端；面向更大规模部署优化架构负载均衡和容器化等；持续引入新方法，完善预测算法体系，提高系统智能化水平。

6.4 结论

这篇文章创建起针对云南主要农作物的产量分析及预测系统，该系统具有数据运作，多模型预测以及可视化表现等功能，可以给农业部门给予辅助分析手段，从整体来看，这个系统达成了设计目的，对于改善预测分析效率，支撑决策来说有一定的参考意义，也具备后续进一步完善和推广的根基。

参考文献

1. 刘钦普. 基于时空回归模型的粮食产量预测方法新探[J]. 世界农业, 2010(3): 45-50.
2. 石杰锋, 李修华，黄为. 基于多种机器学习算法预测广西蔗区甘蔗产量[J]. 智慧农业, 2023, 5(4): 88-98.
3. 赵春江, 王浩, 刘丽. 农业知识智能服务技术综述[J]. 智慧农业, 2023, 5(2): 1-12.
4. 陈光泽. 作物生长模型在农作物生产上的应用[J]. 中国农业科技导报, 2024, 26(4): 120-128.
5. 岳才军, 于树林. GM(1,1)—模糊马尔科夫链联合预测作物产量模型[J]. 系统工程理论与实践, 1992, 12(6): 55-60.
6. 李俊, 王磊. 基于遥感与GIS的农作物产量预测研究进展[J]. 农业工程学报, 2021, 37(16): 1-12.
7. 胡斯威, 王建军. 农业可持续发展研究热点与趋势——基于文献计量的可视化分析[J]. 农业经济问题, 2020(5): 112-120.
8. 刘淼, 梁正伟. 低氮高密增微肥对水稻产量及氮肥利用率的影响[J]. 土壤与作物, 2021, 10(2): 220-228.
9. 王华, 张国庆. 干旱对作物产量影响研究进展与展望[J]. 地理学报, 2021, 76(11): 2500-2515.
10. 张小丽, 李强. 基于深度学习的农作物产量预测模型综述[J]. 计算机科学, 2022, 49(8): 145-153.
11. 刘志强, 王明辉. 基于灰色预测的中国粮食产量研究[J]. 农业系统科学与综合研究, 2019, 35(2): 89-95.
12. 黄丽, 陈晓东. 基于ARIMA模型的云南水稻产量预测研究[J]. 农村经济与科技, 2020, 31(18): 201-203.
13. 王俊, 赵凯. 基于Prophet模型的玉米产量预测实证研究[J]. 统计与决策, 2021(24): 85-88.
14. 郑磊, 李志. 基于XGBoost的农业大数据产量预测模型[J]. 计算机应用, 2021, 41(9): 2607-2612.
15. 周婷, 胡建军. 机器学习方法在小麦产量预测中的应用[J]. 农业机械学报, 2022, 53(7): 112-120.
16. 李瑞, 张磊. 基于LSTM的主要农作物产量预测研究[J]. 智慧农业, 2022, 4(6): 33-42.
17. 陈飞, 杨志. 农业大数据驱动下的作物产量预测研究综述[J]. 农业信息科学, 2023, 15(3): 50-58.
18. 张伟, 王静. 基于多源数据融合的农作物产量预测方法研究[J]. 资源科学, 2023, 45(7): 1350-1362.
19. Wang Y, Zhang Q, Yu F, et al. Progress in Research on Deep Learning-Based Crop Yield Prediction[J]. Agronomy, 2024, 14(10): 2264-2279.
20. Pham H T, Le N, Nguyen T. Enhancing Crop Yield Prediction Utilizing Machine Learning and Remote Sensing[J]. Sensors, 2022, 22(3): 719-730.
21. 洪帅,王天尊,符晓艺. 中国智慧农业研究演进脉络梳理及前沿趋势分析[J].江苏农业科学,2023,51(04): 28-38.
22. 丁楠.基于时间序列蔬菜价格预测及相关性分析[D].山西农业大学,2022. 5-9
23. 王小航.基于长时间序列气象数据的小麦产量预测模型研究[D].山西农业大学,2021. 14-23

致 谢

论文写到这里，总算告一段落了。在这段学习、开发和写作的过程中，真的有很多人给了我帮助和支持，在此想认真说一声感谢。

首先要感谢我的指导老师。从选题确定、系统设计到论文撰写，老师都给了我很多耐心的指导和非常具体的建议。每次我思路不清或者写得不够严谨的时候，老师都会及时指出问题、帮我把方向拉回来，也让我在反复修改中把论文和系统一步步完善起来。同时也感谢各位任课老师在学习阶段的教学与指导，让我在专业知识和工程实践上有了足够的基础，才能顺利完成这次毕业设计。

还要感谢我的同学们在论文写作跟系统开发过程里给出的帮忙。在需求讨论、功能实现、调试排错和界面优化这些地方，大家给了我不少想法，也让我很快找到并改掉了系统里的一些问题，在这里一起说声谢谢。

最后要感谢这次课题开发用到的开源框架、工具和参考资料，它们给系统的实现带来了不少方便，让我可以更专心把核心功能和逻辑部分做得更完整。向所有帮助过我的人表达谢意。