



南开大学
Nankai University

南 开 大 学

计 算 机 学 院

软件工程实验报告

智慧海洋牧场可视化系统开发文档

第 15 组

年级：2022 级

专业：计算机科学与技术、信息安全

指导教师：刘健

2025 年 6 月 29 日

目录

一、 项目概述 1

 (一) 开发背景 1

 (二) 项目目标 1

 1. 数据处理与分析目标 1

 2. 可视化展示目标 1

 3. 报警与通知目标 1

 4. 用户信息管理目标 2

 5. 系统整体目标 2

 (三) 开发环境 2

 (四) 可行性分析 2

 1. 技术可行性 2

 2. 经济可行性 3

 3. 操作可行性 3

 4. 时间可行性 3

 (五) 项目计划 4

 (六) 项目甘特图任务拆解 4

二、 需求分析与系统设计 6

 (一) 需求分析 6

 (二) 系统设计 6

 (三) 详细设计 7

 1. 用户管理模块 7

 2. 数据可视化模块 7

 3. 报警与通知模块 8

 4. 鱼类信息管理模块 9

 5. 水质检测模块 9

 6. 智能问答模块 10

 (四) 数据库设计 11

 (五) ER 图描述 13

 (六) UI 设计 14

三、 系统测试 15

 (一) 测试环境 15

 (二) 功能测试 15

 (三) 性能测试 16

四、 项目管理 16

 (一) 参与人员及分工 16

 (二) 项目进展记录 16

 (三) 项目管理工具 17

五、 用户手册 17

一、项目概述

（一）开发背景

中国在全球渔业养殖领域占据着举足轻重的地位。不过，当前我国水产养殖系统面临着诸多挑战。国外通过运用先进的现代水质传感器技术以及在线监控系统，能够全方位地对养殖环境进行实时监控，并借助自动化养殖设备来精准调控养殖环境，不仅提升了养殖效率，还对生态环境产生了积极影响。为了有效应对我国水产养殖系统所存在的问题，在“十四五”发展计划中，我国规划打造“物联网 + 海洋牧场”模式，利用物联网、大数据、云计算等一系列现代信息技术，构建起一套全面、实时且智能的养殖监控系统，从而实现对养殖环境的精准控制，提高养殖效率，降低病害发生的概率。

（二）项目目标

1. 数据处理与分析目标

- 实现对分布式管理系统中存储的海洋牧场相关数据的高效、准确处理与分析，涵盖海洋环境数据（如水温、盐度、溶解氧等）、养殖生物数据（如生物量、生长速率等）以及设备运行数据等多类型数据。
- 提供多种数据处理与分析功能，包括数据清洗、数据转换、数据聚合以及复杂的数据挖掘和统计分析，以满足不同用户对数据深度分析的需求。
- 根据用户设定的分析需求和规则，自动生成可视化所需的数据，确保数据的准确性和及时性，为后续的可视化展示提供可靠的数据支持。

2. 可视化展示目标

- 构建直观、易用且功能强大的可视化展示界面，通过图形化的方式将处理后的数据清晰地呈现给用户，使用户能够快速、直观地了解海洋牧场的运行状况和趋势。
- 提供丰富多样的可视化元素，包括但不限于各种图表（柱状图、折线图、饼图等）、地图（地理信息展示、养殖区域分布等）、仪表盘（关键指标实时显示）等，以满足不同场景和用户需求的展示要求。
- 优化用户交互体验，确保用户可以通过简单的操作（如点击、拖拽、缩放等）轻松获取所需信息，自由切换和组合不同的可视化元素，实现个性化的数据展示和分析。

3. 报警与通知目标

- 建立实时、准确的报警与通知机制，确保在监测数据出现异常情况时能够及时向用户发送报警信息，帮助用户快速发现潜在问题。
- 提供灵活的报警阈值设置功能，允许用户根据自身海洋牧场的实际情况和管理需求，自定义各类监测数据的报警阈值，以实现精准的异常检测。
- 支持多种报警方式，如短信、邮件、系统消息等，用户可以根据自己的偏好和使用习惯选择合适的报警方式，确保报警信息能够及时、有效地传达给用户，以使用户能够迅速采取相应的处理措施。

4. 用户信息管理目标

- 实现用户通过手机号或账号密码进行注册登录，确保用户身份的唯一性和安全性，为用户提供便捷的系统访问入口。
- 提供用户信息管理功能，包括用户资料的完善、修改以及权限设置等，方便用户对自身账户进行管理和维护。
- 增强系统的安全性和可靠性，通过加入验证码等安全验证机制，有效防止非法登录和恶意攻击，保护用户数据和隐私安全。

5. 系统整体目标

- 打造一个集成数据处理与分析、可视化展示、报警与通知以及用户信息管理等功能于一体的智慧海洋牧场可视化系统，为海洋牧场的管理者和相关人员提供全面、高效、便捷的决策支持和管理工具。
- 确保系统具有良好的扩展性和兼容性，能够适应海洋牧场规模的不断扩大和技术的持续更新，同时支持与其他相关系统（如养殖设备控制系统、环境监测系统等）的无缝对接和数据共享。
- 通过系统的应用，提高海洋牧场的管理效率和科学决策水平，降低运营成本和风险，促进海洋牧场的可持续发展，为海洋渔业的现代化建设提供有力的技术支撑。

（三） 开发环境

本系统采用前后端分离架构进行开发，前端基于原生 HTML + CSS + JavaScript 技术实现可视化界面搭建与交互逻辑编写。HTML 用于构建页面基础结构，CSS 进行样式设计与美化，JavaScript 实现数据动态展示、用户交互响应等功能，并引入 Chart.js 图表库实现数据可视化，确保用户操作流畅。

后端选用 Python 作为开发语言，利用 Flask 轻量级 Web 框架搭建核心服务，实现数据处理、业务逻辑处理以及 API 接口开发。Flask 框架简洁灵活，能够快速响应系统需求变化。数据库采用 MySQL 8.0，用于结构化存储海洋环境数据、养殖生物数据、用户信息等；

操作系统选用 Windows 11 作为本地开发环境，借助其友好的图形化界面与强大的多任务处理能力，方便开发人员进行代码编写、调试与项目管理。集成开发环境使用 Visual Studio Code (VSCode)，凭借其丰富的插件生态，如 Python 插件、HTML/CSS/JavaScript 语法高亮与智能提示插件等，有效提升开发效率；同时，VSCode 支持 Git 版本控制，便于团队协作开发与代码版本管理。在服务器部署方面，采用腾讯云轻量应用服务器，配置为 4 核 8G 内存，操作系统为 CentOS 8。

（四） 可行性分析

1. 技术可行性

从前端技术来看，HTML、CSS 和 JavaScript 作为 Web 开发的基础技术，拥有广泛的开发者社区与丰富的技术文档。Chart.js 图表库能够轻松实现柱状图、折线图等多种可视化图表的绘制，满足系统数据可视化展示需求。通过 JavaScript 对 DOM 的灵活操作和事件监听机制，可以高效实现用户交互功能，且浏览器兼容性良好，能够确保在不同终端上稳定运行。

后端采用 Python 语言搭配 Flask 框架, Python 拥有强大的数据处理与分析库, 如 Pandas、NumPy 等, 能够高效处理海洋环境数据、养殖生物数据等多类型数据。Flask 框架简洁轻量, 路由系统灵活, 便于快速开发 API 接口, 实现数据处理、业务逻辑处理等功能。同时, Flask 生态中存在众多扩展插件, 可轻松实现数据库连接、用户认证等功能, 为系统开发提供便利。

数据库方面, MySQL 8.0 具备强大的数据存储与管理能力, 能够稳定存储结构化数据, 其 ACID 特性保证了数据的完整性与一致性; Redis 作为高性能缓存数据库, 在处理高频访问数据时表现出色, 可有效降低 MySQL 数据库的查询压力, 两者结合能满足系统对数据存储与读取的性能要求。

在系统架构层面, 前后端分离架构使得前后端开发可并行进行, 提高开发效率, 且职责清晰, 便于后期维护与扩展。服务器采用腾讯云轻量应用服务器, 结合 Nginx 的反向代理与负载均衡技术, 能够应对高并发访问场景, 保障系统的稳定性与可靠性。此外, VSCode 开发环境搭配 Git 版本控制, 有利于团队协作开发与代码管理。综合来看, 当前选用的技术栈成熟可靠, 能够满足智慧海洋牧场可视化系统的功能需求, 技术上具备充分可行性。

2. 经济可行性

我国渔业养殖规模庞大, 传统养殖模式普遍存在成本高、效率低、环境调控依赖经验的弊端, 制约了渔业的可持续发展。智慧海洋牧场可视化系统的应用, 能够通过传感器与物联网技术实现对养殖环境参数的实时监测, 并基于数据分析进行精准调控, 从而降低饲料浪费、减少病害风险, 显著提升养殖效率与经济效益。在系统开发过程中, 选用阿里云等成熟的云服务平台, 可按需付费, 避免一次性购置服务器等硬件设备的高昂投入, 显著降低硬件采购与运维成本。同时, 系统架构采用模块化设计, 便于后期功能扩展与维护, 维护成本相对较低。从长期来看, 系统投入成本主要集中在研发与部署阶段, 而运行期间可持续带来养殖产量提升、病害损失降低等直接经济收益, 间接促进渔业企业的数字化转型升级, 经济收益远超开发与运营成本, 具备较强的经济可行性。

3. 操作可行性

系统设计遵循简洁易用的设计原则, 用户界面采用可视化图表与多维度数据展示相结合的方式, 能够直观呈现水温、盐度、溶解氧等环境参数与养殖生产指标的实时状态与历史趋势, 便于用户理解与决策。操作流程充分考虑用户操作习惯, 界面布局清晰, 功能模块分类明确, 用户通过简单的培训即可掌握系统的基本操作方法。同时, 系统提供完善的用户手册与在线帮助文档, 涵盖系统安装部署、功能操作、常见问题解答等内容, 降低用户的学习成本。此外, 系统配备多渠道技术支持, 包括在线客服、电话支持及远程协助, 确保用户在使用过程中遇到问题能够及时得到解决, 保障系统运行稳定可靠。系统的用户权限管理功能能够满足不同层级人员的操作需求与权限分配, 实现按需授权与安全管理, 进一步提升操作的灵活性与可控性, 操作可行性较高。

4. 时间可行性

项目团队成员具备丰富的软件系统开发、物联网集成与项目管理经验, 曾参与多个大型信息化项目的研发, 具备较强的任务执行能力与风险应对能力。在本系统开发过程中, 团队将结合以往项目经验, 采用科学合理的任务拆解与进度安排方法, 对开发任务进行细化分解, 明确各阶段目标与关键路径, 确保项目在预定时间内完成。同时, 项目开发采用敏捷开发模式, 按照迭代周期进行需求分析、系统设计、功能开发与测试验证, 能够根据实际开发情况灵活调整开发计划, 及时应对开发过程中出现的需求变更、技术难题或其他不可预见问题, 保证项目按计划推进, 提

升整体开发效率。通过阶段性成果评审与测试验证，可持续跟踪项目进度与质量，确保最终项目成果符合预期目标与用户需求，具备较强的时间可行性。

（五）项目计划

项目开发周期预计为 3 个月，具体安排如下：

1. **需求分析与设计阶段（第 1 周至第 3 周）**：在项目初期，通过深入调研用户需求，明确系统的功能需求与性能要求，完成系统架构设计、数据库设计与接口设计，输出《需求规格说明书》和《系统设计文档》，为后续开发奠定基础。
2. **系统开发阶段（第 4 周至第 8 周）**：根据设计文档，开展系统各功能模块的开发工作，包括前端页面开发、后端服务接口开发、数据库实现与优化，重点实现数据采集与处理、可视化展示、报警通知、用户信息管理等核心功能。同时，开展单元测试与集成测试，确保代码质量与功能正确性。
3. **系统测试与优化阶段（第 9 周至第 10 周）**：对系统进行全面测试，包括功能测试、性能测试与安全测试，结合测试结果对系统进行漏洞修复与性能优化，完善各项功能，确保系统的稳定性与可靠性。
4. **部署与上线阶段（第 11 周至第 12 周）**：完成系统在生产环境中的部署配置，进行上线前的最终检查与调试，开展用户培训与使用指导，确保系统顺利交付并稳定运行。

项目实施过程中，每周召开项目例会，跟进项目进度，及时解决开发过程中遇到的问题。同时，项目团队通过甘特图对各阶段任务进度进行可视化管理，保障项目按计划高质量推进，确保最终目标顺利实现。

（六）项目甘特图任务拆解

为确保项目按计划顺利实施，项目任务拆解及时间安排如下表所示：

阶段	时间	主要任务
需求分析与设计阶段	第 1 周 - 第 3 周	<ul style="list-style-type: none">• 需求调研与分析• 编写需求规格说明书• 系统架构设计• 数据库设计• 编写系统设计文档
系统开发阶段	第 4 周 - 第 8 周	<ul style="list-style-type: none">• 前端功能开发• 后端接口开发• 数据库实现与优化• 数据处理与分析模块开发• 可视化展示模块开发• 报警与通知功能开发• 用户信息管理模块开发• 单元测试与集成测试
系统测试与优化阶段	第 9 周 - 第 10 周	<ul style="list-style-type: none">• 功能测试• 性能测试• 安全测试• Bug 修复与功能优化
部署与上线阶段	第 11 周 - 第 12 周	<ul style="list-style-type: none">• 生产环境部署• 上线前系统调试• 用户培训与使用指导• 系统最终交付

表 1: 项目甘特图任务拆解表

二、需求分析与系统设计

本节将介绍在充分调研和分析客户需求的基础上，针对系统的功能需求、性能需求和用户体验要求，进行系统架构与模块的设计，并对各模块的功能实现进行详细设计，确保系统具备可用性、可维护性与可扩展性。

（一）需求分析

智慧海洋牧场可视化系统旨在为养殖管理人员提供对养殖环境的实时监测、可视化分析与决策支持，主要需求分析如下：

1. 功能需求

- **用户管理模块：**实现用户注册、登录、权限管理，支持管理员与普通用户不同操作权限。
- **数据采集与处理模块：**对各传感器采集的环境数据进行存储、处理与管理。
- **数据可视化模块：**将水温、盐度、溶解氧等关键环境参数以图表形式实时展示，便于用户直观了解养殖环境状态。
- **报警与通知模块：**当环境参数异常时，及时向用户发送报警信息，实现对环境异常的快速响应。
- **鱼类信息管理模块：**管理养殖鱼类的个体信息，包括种类、体重、体长等，支持数据录入与查询。
- **鱼类体长预测模块：**根据往期鱼类体长数据训练预测模块，对现有的鱼类体长进行预测分析。
- **智能助手模块：**通过 KIMI 的 api 接口，嵌入模式训练一个具有丰富渔业知识的 agent，方便用户获得养殖建议等。

2. 非功能需求

- **易用性：**界面简洁直观，用户经过简单培训即可熟练操作系统。
- **可靠性：**系统应具有稳定的数据采集与处理能力，保证关键功能持续可用。
- **安全性：**通过用户权限管理保障数据安全，防止未授权访问与操作。
- **可扩展性：**系统架构设计应支持后续模块与功能的拓展，如接入更多类型传感器。

（二）系统设计

根据上述需求分析，系统整体采用 **B/S 架构**，前端采用 javascript 实现动态交互界面，后端使用 Flask 提供 RESTful API，数据库使用 MySQL 存储各类结构化数据，系统部署于阿里云服务器，架构设计如下：

- **前端：**基于 HTML + CSS + JAVASCRIPT 框架实现数据可视化界面，通过 Axios 调用后端 API，绘制环境参数趋势图和实时监测图。
- **后端：**采用 Flask 框架构建 API 服务，负责处理用户请求、业务逻辑、数据库操作及与前端的数据交互。

- **数据库**：使用 MySQL 存储用户信息、传感器信息、传感器数据、鱼类信息和水质监测数据，表结构设计遵循规范化要求，保证数据一致性。

系统模块结构包括：

1. 用户管理模块
2. 数据可视化模块
3. 报警与通知模块
4. 鱼类信息管理模块
5. 水质监测模块
6. 智能问答模块

(三) 详细设计

1. 用户管理模块

用户管理模块是系统的核心管理功能之一，旨在提供用户账户的创建、修改和删除等操作。该模块界面简洁明了，列出了所有用户的用户名、角色和操作选项。管理员可以通过此模块轻松地添加新用户、更改用户权限或删除不再需要的账户。每个用户条目旁边都提供了“改变权限”和“删除”按钮，方便管理员进行快速操作。此外，该模块还支持角色管理，确保不同用户根据其职责拥有适当的访问权限，从而提高系统的安全性和操作效率。



图 1: 用户管理界面

2. 数据可视化模块

数据可视化模块是系统的重要组成部分，旨在通过图表和图形化的方式展示复杂的数据。该模块包括多种图表类型，如柱状图、饼图和仪表盘等，帮助用户直观地理解数据趋势和分布。用户可以通过选择不同的参数和种类，动态生成图表，从而快速获取所需的信息。例如，用户可以

查看特定鱼类的重量分布、水质监测数据的仪表盘展示等。数据可视化模块不仅提高了数据的可读性，还增强了用户对数据的理解和分析能力。



图 2: 可视化界面示例

3. 报警与通知模块

报警与通知模块是系统的关键功能之一，旨在及时向用户报告异常情况和重要信息。该模块通过颜色编码和图标提示，突出显示水质监测中的异常数据，如 pH 值异常和浊度偏高等。用户可以快速识别需要关注的参数，并采取相应措施。此外，该模块还支持自定义报警设置，用户可以根据实际需求配置报警阈值和通知方式，确保在关键时刻收到及时的提醒。

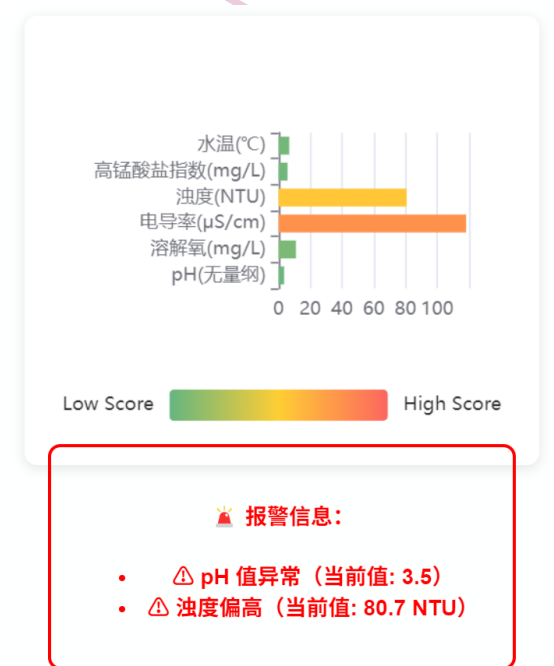


图 3: 报警通知界面

4. 鱼类信息管理模块

鱼类信息管理模块是系统的核心功能之一，旨在管理和维护鱼类个体信息。该模块包括鱼类种类、体重、体长、高度和宽度等参数的记录和管理。用户可以通过该模块添加、修改和删除鱼类信息，确保数据的准确性和完整性。此外，该模块还支持鱼类生长预测功能，用户可以基于输入的体长数据预测鱼类的未来生长情况。鱼类信息管理模块不仅提高了养殖管理的效率，还为生产统计和决策支持提供了有力支持。



图 4: 鱼类信息界面

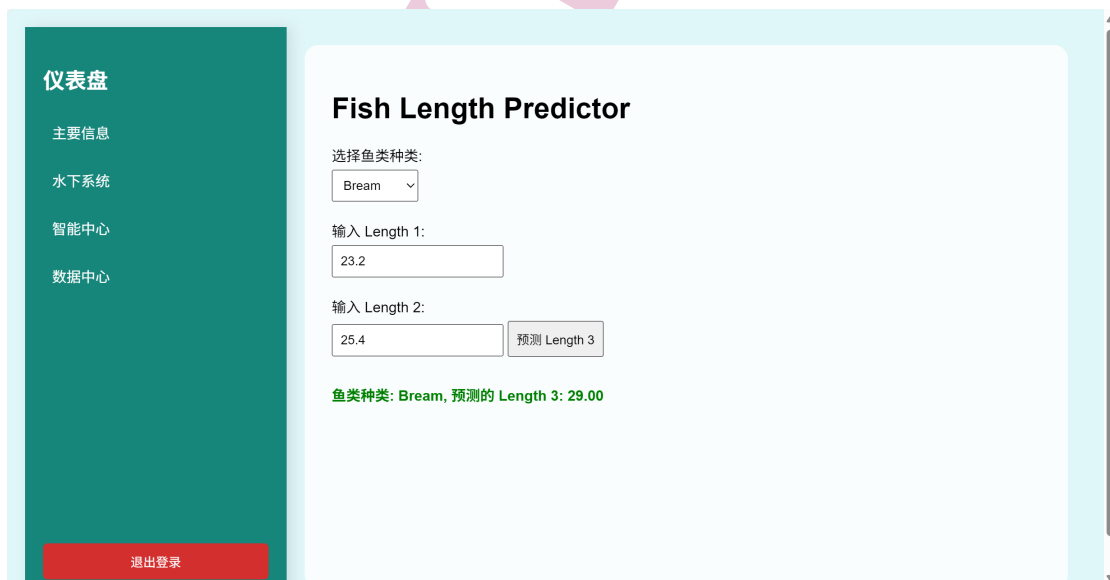


图 5: 体长预测界面

5. 水质检测模块

水质检测模块是系统的重要组成部分，旨在实时监测和记录水质参数。该模块包括水温、高锰酸盐指数、浊度、电导率、溶解氧和 pH 值等多个水质指标的监测。用户可以通过该模块查看

实时水质数据，并生成相应的图表和报告。此外，该模块还支持水质参数的异常检测和报警功能，确保用户能够及时发现和处理水质问题。水质检测模块不仅提高了水质监测的准确性，还为养殖环境的优化提供了科学依据。

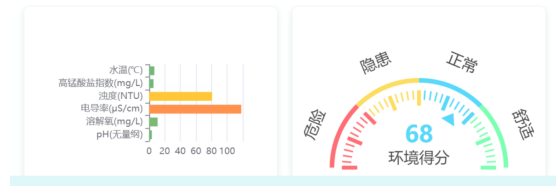


图 6: 水质检测界面

6. 智能问答模块

智能问答模块是系统的创新功能之一，旨在通过自然语言处理技术提供智能化的问答服务。该模块支持用户上传图片并提出问题，系统会自动识别图片中的鱼类种类，并结合问题内容给出详细的回答。例如，用户可以上传一张鱼类图片并询问“这是什么鱼？”系统会识别鱼类种类并提供相关信息。此外，该模块还支持简单的对话功能，用户可以通过文本输入与系统进行互动，获取所需的信息和建议。智能问答模块不仅提高了用户体验，还增强了系统的智能化水平。

简单对话

你: 养殖Bream鱼的环境中最佳pH值范围?

智能助手: Bream鱼（鲢鱼）是一种对水质要求相对较高的鱼类，它们通常在pH值接近中性的水中生长得最好。对于大多数淡水鱼类，包括Bream鱼，最佳的pH值范围通常是6.5到8.5。然而，这个范围可能会根据具体的Bream鱼种类和它们自然栖息地的水质条件有所不同。在养殖环境中，保持pH值的稳定是非常重要的，因为pH值的剧烈变化可能会对鱼类造成压力，影响它们的健康和生长。因此，定期监测和调整pH值是养殖管理中的一个重要环节。如果你需要更具体的信息，可以提供Bream鱼的具体种类，因为不同种类的Bream鱼可能对pH值的偏好有所不同。

图 7: 智能问答界面 (文字)

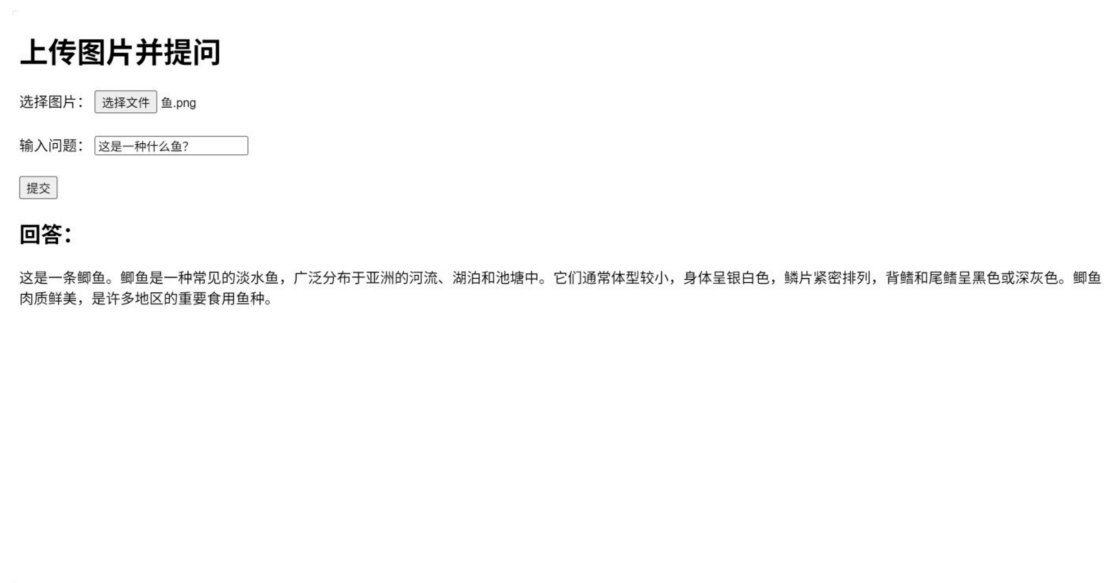


图 8: 图片识别问答界面

(四) 数据库设计

本系统数据库使用 MySQL 进行设计与管理，数据库名称为 `smart_ocean_ranch`，采用 `utf8mb4` 字符集和 `utf8mb4_unicode_ci` 排序规则，以支持多语言和特殊符号输入。

数据库主要包含以下表结构：

1. **users 表**：用于存储用户信息，包括普通用户与管理员，表结构如下：
 - **id**：主键，自增
 - **username**：用户名，唯一
 - **email**：邮箱，唯一
 - **password**：用户密码（当前存储明文密码，仅用于开发测试，实际应使用哈希加密）
 - **role**：用户角色（user/admin）
 - **created_at**：创建时间
 - **updated_at**：更新时间
2. **sensors 表**：用于存储传感器信息，包括传感器名称、类型、位置、状态等，表结构如下：
 - **id**：主键，自增
 - **name**：传感器名称
 - **type**：传感器类型（如温度、盐度等）
 - **location**：传感器位置描述
 - **status**：状态，默认为 active
 - **created_at**：创建时间
 - **updated_at**：更新时间
3. **sensor_data 表**：用于存储传感器采集的数据，表结构如下：

- **id**: 主键, 自增
- **sensor_id**: 外键, 关联 `sensors` 表
- **value**: 传感器数值
- **recorded_at**: 记录时间, 默认为当前时间

4. **fish** 表: 用于存储鱼类个体信息, 表结构如下:

- **id**: 主键, 自增
- **species**: 鱼类种类
- **weight**: 体重 (g)
- **length1**、**length2**、**length3**: 三种体长 (cm)
- **height**: 高度 (cm)
- **width**: 宽度 (cm)
- **created_at**: 创建时间
- **updated_at**: 更新时间

5. **water_data** 表: 用于存储水质监测数据, 表结构如下:

- **id**: 主键, 自增
- **province**: 省份
- **river_basin**: 流域
- **section_name**: 断面名称
- **monitoring_time**: 监测时间
- **water_quality_category**: 水质类别
- **temperature**: 水温 (°C)
- **ph**: pH
- **dissolved_oxygen**: 溶解氧 (mg/L)
- **conductivity**: 电导率 (S/cm)
- **turbidity**: 浊度 (NTU)
- **permanganate_index**: 高锰酸盐指数 (mg/L)
- **ammonia_nitrogen**: 氨氮 (mg/L)
- **total_phosphorus**: 总磷 (mg/L)
- **total_nitrogen**: 总氮 (mg/L)
- **chlorophyll_a**: 叶绿素 (mg/L)
- **algae_density**: 藻密度 (cells/L)
- **station_status**: 站点情况
- **created_at**: 创建时间
- **updated_at**: 更新时间

此外，数据库初始化插入了管理员账户（admin）和普通用户账户（user1），并添加了示例传感器信息及对应的初始数据，便于系统开发与测试。

整体上，数据库设计遵循 **规范化、可扩展性与维护便利** 原则，能够满足系统对用户管理、传感器管理、鱼类信息管理以及水质监测数据管理的综合需求，为智慧海洋牧场可视化系统提供了坚实的数据支撑。

（五） ER 图描述

本系统数据库的实体关系（ER）模型主要包括以下五个核心实体：用户（Users）、传感器（Sensors）、传感器数据（Sensor_Data）、鱼类信息（Fish）、水质信息（Water_Data）。各实体之间的关系描述如下：

1. **Users（用户表）** 用户表用于存储系统用户的基本信息，包括用户名、邮箱、密码以及用户角色（普通用户或管理员）。用户表为独立实体，与其他表暂无直接外键关联，但在系统业务逻辑中，用户操作与数据采集、数据查看存在关联关系。
2. **Sensors（传感器表）** 传感器表用于存储部署于智慧海洋牧场各监测点的传感器信息，包括传感器名称、类型、位置与状态，是系统采集环境数据的核心实体。
3. **Sensor_Data（传感器数据表）** 传感器数据表用于存储各传感器在不同时间采集到的监测数据。该表通过 `sensor_id` 外键与 `Sensors` 表建立一对多关系：
 - **Sensors (1) → Sensor_Data (N)**：每个传感器可以对应多条传感器数据记录。
4. **Fish（鱼类信息表）** 鱼类信息表用于记录养殖鱼类的个体数据，包括种类、体重、各部位体长、宽度和高度等参数，便于养殖管理与生产统计。该表当前为独立实体，与其他表暂无直接外键关系，但在实际业务中，可根据传感器监测水环境与鱼类状态进行综合分析。
5. **Water_Data（水质信息表）** 水质信息表用于存储不同省份、流域及断面在特定时间的水质监测数据，包括温度、pH、溶解氧、总氮总磷、叶绿素、藻密度等多项指标。该表同样为独立实体，与 `Sensors` 表或 `Fish` 表无外键关联，但在系统中可与传感器数据联合分析水质对养殖环境的影响。

整体来看，ER 模型中最主要的实体关系为：

- **Sensors 与 Sensor_Data** 之间的一对多关系。

其他表（Users、Fish、Water_Data）作为独立实体，分别管理用户信息、鱼类个体信息及水质监测数据，为系统的数据采集、存储、管理与分析提供完整支撑。

ER 图设计特点：

- 实体关系清晰，结构规范，便于维护与扩展；
- 数据表之间外键约束合理，保障数据一致性；
- 可根据后续业务需要，将用户与传感器或鱼类管理功能进一步关联，实现更丰富的系统功能模块。

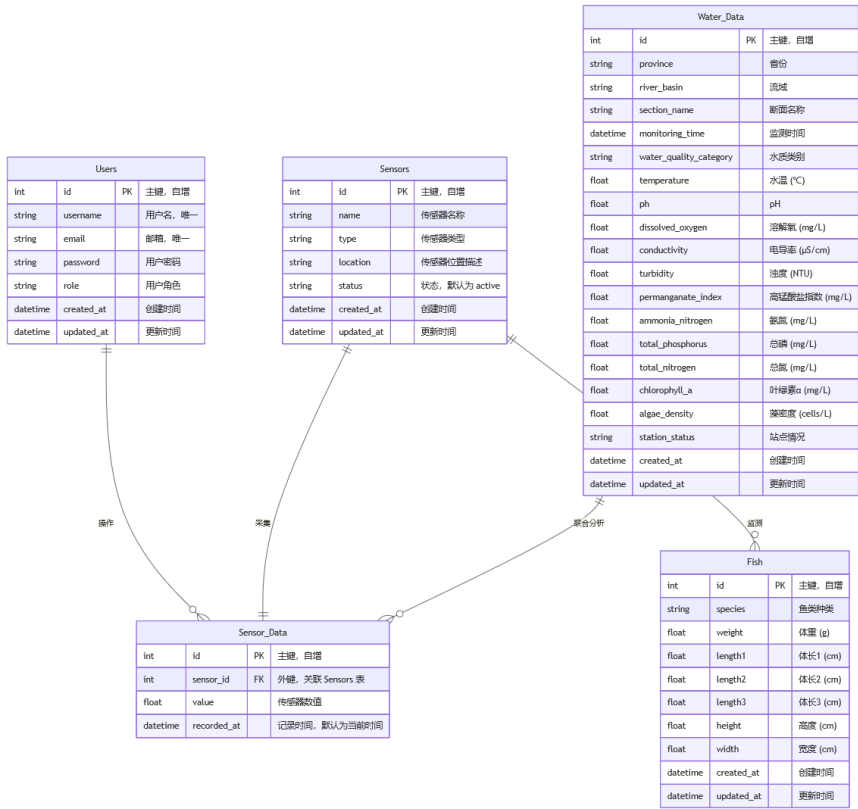


图 9: 数据库设计 ER 图

(六) UI 设计

本系统的 UI 设计注重用户体验，通过简洁明了的布局、直观的图表展示、模块化设计和用户友好的交互方式，使用户能够方便地使用系统的各项功能。UI 设计特点如下：

- **简洁明了的布局**：系统界面布局清晰，功能模块划分明确，用户可以快速找到所需功能。左侧导航栏固定，方便用户在不同模块间切换，增强了用户体验的连贯性。
- **一致的配色方案**：系统采用了统一的配色方案，主要以绿色和白色为主，给人以清新、自然的感觉。绿色通常与环保、自然相关，符合智慧海洋牧场的主题，增强了用户的视觉舒适度。
- **直观的图表展示**：系统中大量使用图表（如柱状图、饼图、仪表盘等）来展示数据，直观易懂。图表设计简洁，数据标签清晰，用户可以快速理解数据的含义，提高了信息传达的效率。
- **模块化设计**：各个功能模块（如主要信息、水下系统、智能中心、数据中心等）独立设计，用户可以专注于当前任务。每个模块都有明确的标题和功能按钮，便于用户操作，增强了系统的可用性。
- **用户友好的交互设计**：提供了多种交互方式，如按钮、下拉菜单、输入框等，用户可以根据需要选择不同的操作。按钮设计简洁，功能明确，如“下载温差图”、“导出温差数据 (CSV)”等，提升了用户的交互体验。
- **响应式设计**：界面设计适应不同设备的屏幕尺寸，确保在各种设备上都能良好显示。导航栏和内容区域可以根据屏幕大小自动调整布局，增强了系统的适应性。

- **信息提示和警告：**对于重要信息和警告，使用醒目的颜色（如红色）和图标进行提示，确保用户能够及时注意到。例如，水质监测中的 pH 值异常和浊度偏高会用红色框和警告图标标出，提高了系统的安全性。
- **多语言支持：**界面中的文字简洁明了，易于理解和翻译，支持多语言用户。这种设计考虑了不同语言背景的用户需求，增强了系统的包容性。
- **数据可视化：**通过图表和图形化的方式展示复杂的数据，帮助用户更直观地理解数据趋势和分布。例如，鱼类种类分布的饼图和环境得分的仪表盘等，增强了数据的可读性。
- **用户权限管理：**提供用户权限管理功能，管理员可以方便地管理用户角色和权限。用户管理界面简洁明了，操作按钮（如“改变权限”、“删除”）功能明确，增强了系统的管理效率。

三、 系统测试

（一） 测试环境

1. 硬件配置：

- **处理器：**AMD Ryzen 7 5800H with Radeon Graphics, 主频 3.20 GHz
- **内存：**16.0 GB RAM
- **网络：**稳定上传链路，满足访问与数据同步需求

2. 软件配置：

- **系统：**64 位 windows 11
- **数据库：**MySQL 8.0.36
- **后端：**Python 3.12.2、Flask 3.1.1、Werkzeug 3.1.3
- **前端：**HTML5、CSS3、ECharts 5.2.2、Chart.js

（二） 功能测试

功能测试旨在验证系统的各项功能是否按预期工作。以下是具体的测试内容和测试结果

1. 用户管理模块

- 管理员将用户删除后，该用户从用户列表中消失，数据库也一并更新。
- 管理员切换用户权限后，该用户权限更改，并且更新数据库记录。

2. 数据可视化模块

- 图表正确展示数据，无错误或遗漏。
- 数据导出的文件应包含所有相关数据，格式正确。
- 图表下载的文件为预期格式，内容完整。

3. 报警与通知模块

- 系统根据数据库实时更新水质参数。
- 当水质参数异常时，系统即时发出报警通知。

4. 鱼类信息管理模块

- 鱼类信息正确记录在系统中。
- 系统根据数据库实时更新鱼类数据。
- 用户上传鱼类数据功能正常。

5. 水质监测模块

- 系统准确监测水质参数。
- 监测数据完整记录在数据库中。

6. 智能问答模块

- 文本交互功能能够提供准确、有用的回答
- 系统能够准确识别图片，并提供相关信息，回答用户问题。

(三) 性能测试

性能测试旨在评估系统在预期负载下的性能表现。以下是具体的测试内容和测试结果

- **响应时间**：测量系统对用户请求的响应时间在 2 秒内
- **吞吐量**：系统能够处理至少 100 个并发用户请求。
- **数据库性能**：数据库操作在毫秒级别完成。
- **资源利用率**：CPU 使用率不超过 80%，内存使用率不超过 90%，磁盘 I/O 操作正常。

四、 项目管理

(一) 参与人员及分工

- **孟启轩**：组长，前端页面设计，移动端显示、报警与通知模块等功能实现
- **赵熙**：前端页面设计，数据处理与分析模块、可视化展示模块完善
- **赵淮佳**：数据库设计，功能测试，智能化问答等模块实现，软件开发文档编写
- **梁锦洲**：数据库设计，可视化展示等模块完善，软件开发文档编写
- **赵乐**：数据库设计，前端页面完善，可视化模块等功能实现

(二) 项目进展记录

1. **2025 年 5 月 14 日**：初步搭建前端页面，调通前后端
2. **2025 年 5 月 15 日**：智能化问答模块实现，前端页面完善，增加鱼类信息数据库
3. **2025 年 5 月 16 日**：增加鱼类信息可视化内容，初步实现可视化展示模块
4. **2025 年 5 月 18 日**：增加水质信息数据库，连接高德天气 API，完善前端页面与数据库连接

5. 2025 年 6 月 18 日：移动端显示、养殖趋势及建议、报警与通知模块完成
6. 2025 年 6 月 25 日：修改数据库配置方式，实现数据导出和图表下载功能
7. 2025 年 6 月 27 日：完善图标下载、数据上传和数据导出功能

(三) 项目管理工具

项目采用 Git 作为版本控制工具，通过 GitHub 远程仓库进行代码托管与协作管理，具体项目链接为<https://github.com/ZhaoXi2210917/Software-Engineering.git>，具体管理方法包括：

- **分支管理策略**：采用 main 分支与 feature 分支相结合的管理模式，其中 main 分支用于存放稳定可部署版本，feature 分支用于各功能模块的开发，开发完成并通过测试后合并至 main 分支。
- **提交规范**：在提交代码时，遵循统一的 Commit Message 格式，简洁明了地描述本次提交的功能、修改内容或修复问题，方便后期维护与查阅。
- **代码审查**：对主要功能模块提交前进行代码审查（Code Review），确保代码质量、风格统一以及功能实现的正确性。

五、 用户手册

1. 注册与登录

进入系统时，若已注册账号，则在左方输入用户名、密码登录；若未注册账号，则在右方输入用户名、密码、邮箱，角色选择“普通用户”进行注册，注册完成后在左方输入用户名、密码进行登录。



智慧海洋牧场 - 注册与登录

该界面包含两个主要部分：登录和注册。

登录部分：位于左侧，包含“用户名”输入框（提示“请输入用户名”）、“密码”输入框（提示“请输入密码”）以及“登录”按钮。

注册部分：位于右侧，包含“用户名”输入框（提示“请输入用户名”）、“密码”输入框（提示“请输入密码”）、“邮箱”输入框（提示“请输入邮箱”）以及“角色”选择框（当前显示“普通用户”）。注册部分还有一个“注册”按钮。

图 10: 注册与登录界面

2. 功能使用

登录后，系统主页如下图所示。此时在左侧选择要使用的功能，其中各部分介绍如下：

- **主要信息**：包含鱼类种群统计、天气预报、水质信息与环境得分
- **水下系统**：包含水下环境介绍视频
- **智能中心**：包含养殖趋势预测、风险评估、决策支持、鱼类体长预测、文本对话与图片问答等功能
- **数据中心**：包含信息可视化与数据上传、数据导出、图表下载等功能。



图 11: 系统主页