

信息系统开发 (.NET) 大实验

实验简介

在现代软件应用中，前后端分离的架构已成为主流。这种模式将负责数据处理、业务逻辑和数据库交互的后端服务，与负责用户界面和交互的前端应用分离开来，通过 API 进行通信。

本实验旨在模拟一个海洋科学研究中的典型图像处理场景。海洋科考团队在水下勘探中会采集海量的图像数据，这些数据存储在数据库中。水下图像是海洋光学观测的重要信息媒介，然而，水下光学成像过程中会受到光信号折射和散射等衰减现象，导致水下图像降质，不利于后续水下研究。为了高效地进行科学分析、物种识别或成果展示，研究人员需要使用图像增强算法对水下图像进行增强，因此，需要一个便捷的工具来浏览这些图像，并从中筛选出最清晰、最有研究价值的图片。因此，实验任务要求你需要构建一个后端 Web API 服务来管理和提供这些图像数据，并开发一个独立的前端客户端软件供研究人员使用。前端应用将调用后端 API 来获取水下图像，呈现给研究人员进行筛选，并将“选优”结果通过 API 再反馈给后端服务器进行归档。

通过完成本次实验，学生将掌握在 .NET 环境下设计和实现 RESTful API、进行数据库编程、设计前端交互界面，以及理解客户端与服务器之间基于 HTTP 的数据交互模式。

实验目标

基础目标

基础目标必须全部正确实现。

对于前端部分，即客户端程序：

- 学习并实践如何使用 .NET 技术连接到后端，能够通过 HTTP 请求（GET, POST）在前端和后端之间传输数据。
- 能够设计并实现一个图形用户界面，要求用户先进行登录，然后进行以下 workflow：
 - 从后端获取图像信息，展示若干图像并允许用户选择选项中的一个。
 - 记录用户的选择，并将相关选择信息返回给后端服务器。
 - 获取新的图像，然后重复此循环。

对于后端部分，即服务端程序：

- 掌握如何使用 .NET 技术连接数据库并进行数据操作，例如从数据库表中读取图像并在客户端应用程序中正确呈现。

- 提示: 可以将图像二进制转换为 Base64 字符串 来直接处理, 也可以仅存储一个指向图像文件的 URL, 然后根据 URL 来进行文件操作。
- 学习使用 .NET 技术来设计并实现一个用户系统, 包括如何设置并保存用户的登录状态。
 - 业界的常见解决方案有: JWT 或 Session, 请查找相关资料学习如何安全地使用用户登录。
 - **永远不要在数据库内直接明文存储用户的密码!** 你应当对密码进行哈希加盐的安全处理。
- 学习使用 .NET 技术构建后端 RESTful 服务(ASP .NET Core WebAPI), 并向前端至少提供以下几个数据交互端点:
 - 一个 POST 接口, 携带一个登录的用户账号和密码时, 验证通过则使用户登录, 否则拒绝登录。
 - 一个 GET 接口, 登录的用户请求时能够提供若干图像信息, 否则应响应错误。
 - 一个 POST 接口, 登录的用户携带一个选项请求体请求时能记录此选项并追加写入数据库中, 否则应响应错误。

拓展目标

拓展目标为选做, 并且可以不限于推荐的目标。

- 使用版本管理系统(如 Git)来处理团队成员的代码协作, 规范填写 Commit Message 和使用 Pull Request。
- 使用 ORM 技术代替裸 SQL 命令来进行数据库操作。
- 提供一个注册模块, 让用户能自行注册账号。
- 实现权限管理分级(普通用户, 管理员等), 管理员能管理标注人的账户数据。
 - 更进一步地, 管理员能审核新注册的游客账号, 并将他们设置为普通用户, 游客不能参加标注工作。
- 添加自动化测试系统, 实现一个 CI/CD 工作流。

实验要求

你的小组必须实现全部的基础目标, 并且注明每个组员的工作, 建议你的小组分为前端和后端两个小队来协作, 组长应做好相应的统筹安排。

你可以使用课上教授的 WinForm + ASP .NET Core Web API 技术栈来完成, 也可以选用任何课外的基于 .NET C# 的框架来完成此实验。

重申: 可以使用 AI 工具请教问题, 以及提供思路与问题分析, **但是一旦发现项目代码滥用 AI 实现、抄袭他人代码或实验课随机抽查时针对项目代码不能给出合理的思路与实现解释, 将被大幅减分或取消此次实验的成绩。**

实验交付

完成实验后，你需要打包整个项目代码，包括前端和后端代码，前端和后端都需要提供一份简要说明文档（此部分的负责成员，如何运行和亮点等），最后在整个项目的根目录下提供实验的报告。

完成实验后，你的小组可能将进行答辩，但无需准备 PPT。届时你的小组要简单介绍项目设计与实现，各部分的负责成员需要回答一些简短的提问。