

# PBL项目实践报告评分标准

课程名称: Web应用开发实践

项目类型: 基于项目的学习 (PBL)

总分: 100分

## 一、项目规划与需求分析 (10分)

### 1.1 项目选题与背景 (3分)

- 优秀 (3分)**: 选题具有实际应用价值, 项目背景描述清晰, 问题定义明确
- 良好 (2分)**: 选题合理, 背景描述基本清晰
- 及格 (1分)**: 选题简单, 背景描述不够完整
- 不及格 (0分)**: 选题不合理或缺少背景说明

### 1.2 需求分析与项目规划 (7分)

#### 功能需求分析 (3分)

- 详细列出系统功能需求 (注册、登录、管理员功能、用户功能等)
- 使用用例图或功能列表清晰表达需求
- 非功能需求分析 (性能、安全、可用性)

#### 项目计划与分工 (4分)

- 时间规划 (2分): 包含详细的项目时间表和里程碑
- 团队分工 (2分): 明确的任务分配和责任划分

## 二、数据库设计 (15分)

### 2.1 数据库设计文档 (5分)

#### ER图设计 (3分)

- 实体关系图完整, 实体、属性、关系表达清晰准确
- 实体定义清晰, 关系表达正确

#### 数据库规范化 (2分)

- 满足第三范式，无数据冗余，设计优雅
- 表之间关系合理，外键约束完整

## 2.2 数据表结构设计（10分）

**要求：至少设计5张表，每张表2分**

**必需的表：**

1. **用户表 (users)** - 2分
2. **管理员/角色表 (admins/roles)** - 2分
3. **业务表1** - 2分
4. **业务表2** - 2分
5. **业务表3** - 2分

**每张表的评分标准（2分）：**

- **表结构合理性（0.8分）**
  - 字段命名规范
  - 数据类型选择恰当
  - 字段长度合理
- **主键与索引设计（0.6分）**
  - 主键设计合理
  - 必要字段建立索引
- **外键与约束（0.6分）**
  - 外键关系正确
  - 约束设置合理（NOT NULL、UNIQUE、DEFAULT等）

---

## 三、用户认证与权限管理（15分）

### 3.1 用户注册功能（5分）

**前端实现（2分）**

- 注册界面美观，字段完整（1分）
- 客户端表单验证完整（1分）

**后端实现（3分）**

- 服务端数据验证（1分）
- 密码安全加密存储（bcrypt、MD5+盐等）（1.5分）
- 数据正确入库，错误处理完善（0.5分）

## 3.2 用户登录功能（5分）

### 登录验证（2.5分）

- 用户名/邮箱+密码验证正确（1.5分）
- 登录失败提示友好（1分）

### 会话管理（2.5分）

- Session/Token管理机制完善（2分）
- 会话过期处理（0.5分）

## 3.3 权限控制与安全（5分）

### 角色权限（3分）

- 管理员与普通用户权限明确分离（1.5分）
- 角色权限验证完整（1.5分）

### 访问控制（2分）

- 未登录用户重定向到登录页（1分）
- 权限不足时的处理（1分）

---

## 四、功能模块实现与测试（30分）

### 4.1 管理员模块（10分）

#### 管理员界面（3分）

- 界面布局合理，导航清晰（1.5分）
- 功能模块划分明确（1.5分）

#### 用户管理功能（3分）

- 查看用户列表（支持分页/搜索）（1分）
- 编辑用户信息（1分）
- 删除用户（含确认提示）（1分）

#### 业务数据管理（4分）

- 对业务表1的管理功能（1.5分）
- 对业务表2的管理功能（1.5分）
- 数据统计与展示（1分）

## 4.2 用户模块（10分）

### 个人中心（4分）

- 个人信息完整展示（2分）
- 个人资料修改功能（2分）

### 业务操作功能（6分）

- 数据查询与列表展示（1.5分）
- 数据新增功能（1.5分）
- 数据修改功能（1.5分）
- 数据删除功能（1.5分）

## 4.3 5张表的CRUD操作（5分）

### CRUD完整性检查（每张表1分）

- 5张表均实现完整的增删改查功能（5分）
- 表单验证完整，数据回显正确
- 删除操作有确认提示
- 列表支持分页或搜索

## 4.4 功能测试（5分）

### 测试执行（3分）

- 编写测试用例文档（1分）
- 对主要功能进行测试（1.5分）
- 提供测试报告（0.5分）

### Bug修复（2分）

- 记录测试中发现的问题（1分）
- 修复主要bug（1分）

---

## 五、开发工具、团队协作与文档（30分）

### 5.1 AI辅助编程应用（10分）

#### AI工具使用记录（3分）

- 列出使用的AI工具（GitHub Copilot、ChatGPT、Claude、Cursor等）

- 说明具体使用场景（至少3个实例）
- 提供AI辅助编程的截图证明

AI辅助效果评估（5分）

- AI生成代码的质量与优化
- 开发效率提升说明（统计节省时间或提高的生产力）
- 学习收获与反思（通过AI学到的新知识）

5.2 Git版本控制（10分）

仓库管理（3分）

- .gitignore配置合理
- Commit信息清晰有意义
- 提交频率合理，至少20次有效提交

分支管理（3分）

- 使用功能分支开发
- 分支命名规范，合并流程规范

5.3 GitHub团队协作（10分）

仓库文档（2分）

- README.md文档完整（项目介绍、安装说明、使用说明）
- 项目描述清晰

协作记录（4分）

- 多人协作记录（至少2人有提交）
- Pull Request使用（至少3个PR）（
- Code Review记录
- Issues管理（至少5个有效Issue）或Project Board使用

评分说明

总分构成

章节	评分项目	分值
一	项目规划与需求分析	10分
二	数据库设计	15分

章节	评分项目	分值
三	用户认证与权限管理	15分
四	功能模块实现与测试	30分
五	开发工具、团队协作与文档	30分
	总计	100分

## 等级划分

- 优秀 (90-100分)**：全面完成所有要求，代码质量高，文档完善，有创新点
- 良好 (80-89分)**：完成主要功能，代码质量较好，文档较完善
- 中等 (70-79分)**：完成基本功能，代码可运行，文档基本完整
- 及格 (60-69分)**：完成部分功能，存在一些问题但可以运行
- 不及格 (<60分)**：功能不完整，代码无法运行或文档严重缺失

## 额外加分项（最多+10分）

- 界面设计优秀 (+3分)**
  - 使用现代前端框架 (Vue/React)
  - UI美观，用户体验好
  - 响应式设计
- 高级功能实现 (+3分)**
  - 图片上传功能
  - 数据导出 (Excel/PDF)
  - 邮件通知
  - 第三方登录
- 性能优化 (+2分)**
  - 数据库查询优化
  - 缓存机制
  - 前端性能优化
- 安全性增强 (+2分)**
  - SQL注入防护
  - XSS防护
  - CSRF防护
  - API限流

## 扣分项

- 代码抄袭**：发现大段代码抄袭，视情节严重程度扣10-30分
- 文档造假**：虚假的截图或数据，扣5-15分
- 核心功能缺失**：每项扣3-5分
- 无法运行**：项目无法正常部署运行，扣10-20分

5. **严重安全漏洞**：密码明文存储、SQL注入漏洞等，扣5-10分

## 核心功能检查清单

### ✅ 必需功能（缺一项扣5分）

- ☐ 用户注册功能完整
- ☐ 用户登录功能完整
- ☐ 管理员与普通用户权限分离
- ☐ 至少5张数据库表
- ☐ 5张表均实现CRUD操作
- ☐ 管理员可管理用户
- ☐ 管理员可管理业务数据
- ☐ 用户可管理自己的数据
- ☐ 使用AI辅助编程（有证明）
- ☐ Git仓库管理规范（>20次提交）
- ☐ GitHub团队协作（多人、PR、Issues）

## 提交要求

### 1. 提交内容

#### 必需材料

- ☐ GitHub仓库链接（公开或邀请教师）
- ☐ 数据库SQL文件（建表+测试数据）
- ☐ 项目实践报告（PDF格式）
- ☐ README.md（项目说明）

#### 推荐材料

- ☐ 项目演示视频（5-10分钟）
- ☐ 功能截图集
- ☐ 部署文档

### 2. 提交格式

项目根目录/

├─ 项目实践报告.docx

├─ 项目实践报告.pdf

```
├─ database/
│   └─ database.sql (必需)
├─ docs/
├─ screenshots/ (功能截图)
└─ src/ (源代码)
```

### 3. 报告命名规范

学号-姓名-PBL项目实践报告.pdf

### 4. 提交截止时间

第16周周日 23:59

## 各章节评分要点速查

#### 第一章：项目规划与需求分析（10分） ⌚ 5分钟

- ✓ 选题实用性
- ✓ 需求分析完整性
- ✓ 项目计划可执行性

#### 第二章：数据库设计（15分） ⌚ 8分钟

- ✓ 5张表结构合理
- ✓ ER图准确
- ✓ 符合规范化

#### 第三章：用户认证与权限管理（15分） ⌚ 8分钟

- ✓ 注册登录流程完整
- ✓ 密码加密
- ✓ 权限隔离明确

#### 第四章：功能模块实现与测试（30分） ⌚ 15分钟

- ✓ 管理员功能完整
- ✓ 用户功能完整
- ✓ 5张表CRUD完整
- ✓ 有测试报告

#### 第五章：开发工具、团队协作与文档（30分） ⌚ 12分钟

- ✓ AI工具使用有证明



- ☒ Git提交规范 (>20次)
- ☒ GitHub协作完整 (PR+Review+Issues)
- ☒ 代码质量好
- ☒ 文档完整

总评分时间：约48分钟/份

---

## 附录：推荐项目类型

### 1. 学生信息管理系统

5张表示例：

- users (用户表)
- admins (管理员表)
- students (学生表)
- courses (课程表)
- grades (成绩表)

### 2. 图书管理系统

5张表示例：

- users (用户表)
- admins (管理员表)
- books (图书表)
- borrow\_records (借阅记录表)
- categories (分类表)

### 3. 在线选课系统

5张表示例：

- users (用户表)
- admins (管理员表)
- courses (课程表)
- course\_selection (选课记录表)
- teachers (教师表)

### 4. 博客系统

5张表示例：

- users (用户表)
- admins (管理员表)
- articles (文章表)
- comments (评论表)
- categories (分类表)

## 5. 任务管理系统

### 5张表示例:

- users (用户表)
- admins (管理员表)
- projects (项目表)
- tasks (任务表)
- teams (团队表)

## 推荐技术栈

### 后端框架

- **Python**: Flask / Django
- **Java**: Spring Boot
- **Node.js**: Express / Koa
- **PHP**: Laravel / ThinkPHP

### 前端技术

- **基础**: HTML + CSS + JavaScript
- **框架**: Vue.js / React / Angular
- **UI库**: Bootstrap / Element UI / Ant Design

### 数据库

- **关系型**: MySQL / PostgreSQL / SQLite
- **工具**: Navicat / phpMyAdmin / DBeaver

### AI辅助工具

- **GitHub Copilot** (推荐)
- **Cursor** (推荐)
- **ChatGPT** / **Claude**
- **通义灵码** / **文心快码**

## 版本控制

- **Git + GitHub**
- **GitLab** (备选)

## 评分细则补充说明

### 关于AI辅助编程（8分）

#### 必需提供的证明材料：

1. AI工具使用截图（至少3张）
2. 对话记录或代码片段
3. 使用场景说明

#### 评分重点：

- 不是简单使用AI，而是理解并优化AI生成的代码
- 能说明AI如何帮助解决具体问题
- 有真实的学习收获

### 关于Git与GitHub（12分）

#### Git提交规范：

feat：添加用户注册功能  
fix：修复登录密码验证bug  
docs：更新README文档  
style：优化代码格式  
refactor：重构用户管理模块  
test：添加单元测试

#### GitHub协作要求：

- PR标题清晰，描述详细
- Code Review要有实质性意见
- Issues要有明确的任务描述

### 关于代码质量（5分）

#### 代码规范参考：

- Python: PEP 8
- JavaScript: Airbnb Style Guide

- Java: Google Java Style Guide
- 

**制定日期：** 2024年11月

**修订版本：** V3.0 (5章版)

**说明：** 本评分标准最终解释权归课程组所有。