

北京化工大学

软件工程课程设计个人总结

班级：计科2203

学号：2022040055

姓名：于豪杰

1. 任务概述

1.1 项目背景

本次课程设计项目是"掌上博物馆"，它是海外藏中国文物知识管理与服务平台中的一个子系统。整个平台包含了知识图谱构建、Web服务、知识问答、掌上博物馆和后台管理等五个子系统。掌上博物馆是一个基于HarmonyOS开发的移动端应用，专注于展示从波士顿美术馆、纳尔逊-阿特金斯艺术博物馆和明尼阿波利斯艺术博物馆爬取的中国文物数据，为用户提供便捷的文物浏览、搜索和交流功能。

作为掌上博物馆子系统的开发组长，我负责团队的整体协调和技术指导，制定了基于Git Flow的团队协作方案，并带领团队成员高效完成了项目开发工作。我还确保了掌上博物馆与其他子系统的有效对接，特别是与后台管理系统和知识图谱子系统的数据交互。

1.2 个人职责

作为掌上博物馆子系统开发组长，我主要负责以下工作：

- 团队管理与协调**：组织团队会议，分配任务，制定开发计划和进度，协调解决团队内部的技术难题和资源问题
- 技术方案制定**：设计系统架构，制定技术路线和开发规范，包括Git Flow工作流、代码审查规范和提交规范
- 核心模块开发**：带领团队开发文物展示模块和图像搜索功能，这是掌上博物馆的关键功能
- 质量把控**：审查团队成员的代码，确保代码质量和一致性，组织团队进行测试和问题修复
- 跨团队协作**：作为团队代表与其他子系统的负责人沟通，确保系统间接口的一致性和数据交互的顺畅
- 技术文档管理**：督导技术文档的编写和维护，包括测试文档、开发指南和用户手册

2. 团队协作管理

2.1 协作流程与分支管理

作为团队组长，我设计并推行了基于GitHub的Git Flow workflow进行团队协作，主要包括以下步骤：

- 1. **制定分支策略**：明确划分main、develop、feature等不同用途的分支，并制定了详细的分支命名规范
- 2. **规范开发流程**：要求团队成员从develop分支创建功能分支，完成后提交PR，经过代码审查后合并到develop分支
- 3. **组织代码审查**：亲自审查关键代码，并指导高级团队成员参与代码审查，确保代码质量
- 4. **冲突管理**：指导团队成员正确处理代码合并冲突，必要时亲自参与解决复杂冲突

我的分支管理策略有效防止了代码冲突和质量问题，使团队开发工作有条不紊地进行：



2.2 团队建设与技术指导

作为组长，我在团队建设和技术指导方面做了以下工作：

- 1. **编码规范制定**：制定了详细的编码规范和提交信息规范，确保团队代码风格统一，如：

```
feat(artifact): 实现文物列表瀑布流布局
fix(search): 修复图像搜索结果排序问题
docs(api): 更新文物API文档
```

- 2. **技术难题攻关**：带领团队攻克HMS ML Kit集成和图像搜索算法优化等技术难题
- 3. **知识共享机制**：建立团队知识库和问题解决文档，促进经验分享

2.3 沟通协调与问题解决

作为组长，我建立了高效的沟通机制和问题解决流程：

- 1. **定期会议**：组织周例会，掌握项目进度，及时发现和解决问题
- 2. **技术评审会**：在关键功能开发前组织技术评审，确保方案的可行性和合理性
- 3. **跨团队沟通**：代表团队与其他子系统团队的组长定期沟通，协调接口和数据规范
- 4. **危机管理**：遇到突发技术问题时，及时组织相关人员分析问题根源，制定解决方案
- 5. **文档评审**：定期检查和更新项目文档，确保文档与代码的一致性

3. 技术领导与贡献

3.1 系统架构设计

作为技术负责人，我主导了掌上博物馆子系统的架构设计：

- 1. **前后端分离架构**：设计了基于ArkTS的前端和Node.js+Express的后端分离架构

2. **模块化设计**：将系统划分为用户模块、首页模块、文物展示模块、文物详情模块、搜索模块和收藏评论模块
3. **数据流规范**：制定了前端数据流转规范，采用MVVM架构模式提高代码可维护性
4. **接口设计**：设计了与后台管理系统和知识图谱系统交互的API接口规范

3.2 核心功能实现与指导

我不仅参与了核心功能的开发，还对团队成员进行了技术指导：

1. 文物展示模块：

- 设计了瀑布流和网格两种布局方式的切换机制
- 指导团队实现文物分类筛选功能
- 亲自编写虚拟列表和图片懒加载的核心代码，解决性能瓶颈
- 审核并优化了团队成员开发的列表UI交互设计

2. 图像搜索功能：

- 主导HMS ML Kit的集成方案设计
- 研究并指导团队优化图像特征提取算法
- 设计了搜索结果的相似度排序算法
- 带领团队实现搜索性能优化，减少响应时间

3.3 质量保障体系建设

为确保项目质量，我建立了完善的质量保障体系：

1. **代码审查制度**：制定代码审查规范，指导团队成员参与代码审查
2. **测试策略**：设计了单元测试、集成测试和UI测试策略
3. **性能监控**：引入性能监控工具，及时发现和解决性能问题
4. **持续集成**：尝试引入简单的CI流程，自动化构建和测试

4. 总结与反思

4.1 领导力与个人成长

通过担任掌上博物馆子系统的开发组长，我在以下方面获得了显著成长：

1. **团队领导力**：学会了如何有效管理和激励团队，分配任务，协调资源，处理团队冲突，带领团队高效完成项目目标。
2. **技术决策能力**：提升了技术选型、架构设计和技术路线规划的能力，能够在技术方案的可行性、先进性和团队适应性之间做出平衡。
3. **沟通协调能力**：锻炼了与团队成员、其他子系统负责人和项目管理者之间的沟通能力，学会了有效表达技术观点和解决分歧。
4. **危机处理能力**：面对技术难题和进度压力，培养了冷静分析问题、快速制定解决方案的能力。
5. **技术能力提升**：通过带领团队攻克技术难题，自身的HarmonyOS应用开发、ArkTS语言和HMS服务等技术能力也得到了深化。

4.2 项目管理经验

作为组长，我积累了宝贵的项目管理经验：

1. **计划与控制**：学会了如何制定合理的开发计划，分解任务，控制项目进度，及时调整资源分配。
2. **风险管理**：提前识别潜在风险，如技术难题、人员变动等，并制定应对策略，减轻风险影响。
3. **质量管理**：建立质量标准和保障机制，确保产品质量符合预期要求。
4. **团队建设**：营造积极的团队氛围，平衡团队成员的工作负荷，促进团队协作和知识共享。
5. **变更管理**：灵活应对需求变更和技术调整，减少对项目的负面影响。

4.3 遇到的挑战与解决策略

作为组长，我面临了一些挑战并采取了有效的解决策略：

1. **团队技术水平差异**：团队成员对HarmonyOS和ArkTS的熟悉程度不一，我安排了培训和配对编程，帮助团队快速上手。
2. **跨团队协作障碍**：与其他子系统的接口对接初期存在理解偏差，我主动与各子系统组长建立了定期沟通机制，统一了接口规范。
3. **技术方案争议**：团队对某些技术实现方案存在分歧，我组织了技术讨论会，通过原型验证和数据支持，帮助团队达成一致。
4. **进度压力**：中期出现了进度滞后的风险，我重新评估了任务优先级，调整了人力分配，并引入了阶段性里程碑来加强进度监控。
5. **代码质量问题**：初期代码审查不严格导致了一些质量问题，我强化了代码审查流程，并建立了核心代码的集体评审机制。

4.4 对未来工作的启示

担任组长的经历给我带来了以下启示：

1. **技术领导力与管理能力并重**：一个好的技术组长不仅需要扎实的技术功底，还需要有效的团队管理和协调能力。
2. **流程和规范的重要性**：明确的开发流程和规范对于保证项目质量和进度至关重要。
3. **沟通是解决问题的关键**：及时、有效的沟通可以预防和解决大部分问题。
4. **持续学习的必要性**：技术发展迅速，作为组长需要不断学习新技术，引导团队成长。
5. **以身作则的影响力**：组长的工作态度和技术水平对整个团队有示范作用。

4.5 结语

作为掌上博物馆子系统的开发组长，我不仅带领团队成功完成了项目开发任务，还在团队管理、技术决策和跨团队协作方面积累了宝贵经验。通过制定Git Flow工作流、建立代码审查机制和推动技术创新，我为项目质量和团队能力提升做出了贡献。

这次担任组长的经历极大地提升了我的综合能力，使我更全面地理解了软件工程的各个环节和团队协作的重要性。我将把这些宝贵经验应用到未来的工作中，不断提升自己的技术水平和领导能力，创造更大的价值。