

移动应用开发课程教学标准

一、课程基本信息

1. **课程名称:** 移动应用开发
2. **课程代码:**
3. **适用专业:** 计算机应用技术、移动应用开发
4. **学年学期:**
5. **总学时:** 72学时 (理论教学36学时, 实践教学36学时)
6. **考核方式:** 考试
7. **适用层次:** 全日制专科

二、课程的地位、性质和任务

本课程是计算机应用技术专业的核心课程,具有较强的实践性和综合性。旨在培养学生掌握移动应用开发的核心技术,能够运用uni-app框架进行跨平台移动应用的开发。通过理论知识学习和大量实践操作,使学生熟悉移动应用开发流程,具备独立完成移动应用项目从需求分析、设计、编码到测试部署的能力,为从事移动应用开发相关职业奠定坚实基础。同时,注重培养学生的创新思维、团队协作精神和解决实际问题的能力,提升学生的职业素养,使其能够适应行业发展的需求。

三、教学基本要求

1. 以项目实践为导向,围绕实际移动应用开发项目组织教学内容,让学生在项目实践中掌握移动应用开发技术。
2. 理论教学与实践教学紧密结合,合理安排理论与实践学时比例,确保学生在理解理论知识的基础上,通过实践操作熟练掌握开发技能。
3. 关注行业发展动态,及时更新教学内容,引入最新的移动应用开发技术和工具,使学生所学知识与行业需求保持同步。
4. 注重培养学生的自主学习能力和创新能力,鼓励学生在实践中探索新技术、新方法,提高学生解决实际问题的能力。

四、教学内容及教学层次

(一) 初识uni-app (4学时)

1. **知识点:** uni-app的发展历程、开发工具(HBuilderX)的特点及优势、项目搭建流程、项目目录结构、项目运行和打包方式。
2. **重点:** HBuilderX的使用、项目搭建及运行打包操作、项目目录结构理解。
3. **难点:** 理解uni-app跨平台开发原理、项目打包复杂参数配置。
4. **基本要求:** 学生能阐述uni-app发展历程,熟练使用HBuilderX创建、运行和打包项目,清楚项目目录结构各部分功能。
5. **实践与练习:** 创建uni-app项目,进行项目运行和打包操作,分析项目目录结构。

(二) uni-app基础 (6学时)

1. **知识点:** pages.json文件配置、资源引用与页面样式、尺寸单位与基础组件、flex布局、案例实践与拓展。

2. **重点**: pages.json文件关键配置、资源引用和样式设置、基础组件使用、flex布局应用。
3. **难点**: 理解subPackages分包加载原理、掌握属性绑定和事件绑定机制、实现flex布局复杂效果。
4. **基本要求**: 熟练配置pages.json文件, 正确进行资源引用和样式设置, 灵活运用基础组件和flex布局开发页面, 完成模拟项目开发。
5. **实践与练习**: 根据给定设计稿完成页面开发, 实现模拟影院购票系统和志愿者服务小程序“志愿者活动页”。

(三) uni-app基础内容扩展 (6学时)

1. **知识点**: uni-app生命周期、条件编译、扩展组件 (uni-ui、uview等) 的使用、新闻列表案例实现、实训与拓展项目。
2. **重点**: 生命周期函数应用、条件编译语法及应用、uni-ui扩展组件使用、新闻列表案例关键代码实现。
3. **难点**: 理解组件生命周期函数差异、编写复杂条件编译逻辑、运用扩展组件实现复杂功能。
4. **基本要求**: 能在项目中合理运用生命周期函数和条件编译, 熟练使用uni-ui扩展组件, 完成实训和拓展项目开发。
5. **实践与练习**: 开发包含新闻列表功能的应用, 完成实训项目和拓展实训项目。

(四) uni-app组件 (12学时)

1. **知识点**: 各类常用组件 (容器组件、基础组件、表单组件等) 的属性、事件和应用场景、组件间交互与数据传递、组件优化与性能提升、综合项目实践。
2. **重点**: 常用组件核心属性和事件应用、表单组件数据提交处理、组件间交互与数据传递方式、综合项目实践。
3. **难点**: 复杂组件深入理解与灵活运用、实现组件间复杂交互效果、根据项目需求选择和优化组件。
4. **基本要求**: 熟练运用常用组件开发复杂页面, 掌握组件间交互和数据传递方法, 能对组件进行优化, 完成综合项目开发。
5. **实践与练习**: 开发公司首页、仿学习强国首页等综合项目, 进行组件优化和性能提升实践。

(五) uni-app常用API(1) (10学时)

1. **知识点**: uni-app常用API (计时器、交互反馈、网络、数据缓存、路由等) 的功能、参数和使用场景、综合案例 (智云翻译项目) 开发。
2. **重点**: 常用API使用方法和参数配置、交互反馈API应用、智云翻译项目核心功能实现。
3. **难点**: 理解API跨平台差异和兼容性、掌握数据缓存策略和管理、实现智云翻译项目复杂业务逻辑。
4. **基本要求**: 熟练使用常用API完成项目功能开发, 掌握智云翻译项目开发流程和技术要点。
5. **实践与练习**: 完成智云翻译项目开发, 对项目进行测试和优化。

(六) uni-app常用API - 2 (12学时)

1. **知识点**: 媒体控制、文件操作、设备操作、第三方服务等API的功能、参数和使用场景、自定义组件创建和使用、“仿网易云”音乐播放器项目开发。
2. **重点**: 各类API实际应用、自定义组件创建和使用、“仿网易云”音乐播放器项目核心功能实现。
3. **难点**: 处理API跨平台差异和兼容性、实现第三方服务API数据交互和认证机制、“仿网易云”音乐播放器项目复杂业务逻辑和交互效果。
4. **基本要求**: 熟练运用各类API和自定义组件开发项目, 完成“仿网易云”音乐播放器项目开发。
5. **实践与练习**: 开发“仿网易云”音乐播放器项目, 实现音频播放、录音、图片处理等功能。

(七) 智慧环保项目 (12学时)

1. **知识点**：uni-app开发流程、组件、API和扩展组件uView的综合运用、智慧环保项目业务逻辑和功能模块、json-server的安装、配置及使用。
2. **重点**：智慧环保项目各页面功能实现、uView组件使用、项目数据请求和处理。
3. **难点**：理解和实现项目业务逻辑、解决跨页面数据传递和共享问题、处理项目兼容性问题。
4. **基本要求**：能够运用所学技术独立完成智慧环保项目开发，解决开发过程中的问题。
5. **实践与练习**：完成智慧环保项目各页面开发和功能实现，对项目进行优化和测试。

(八) 天气预报项目 (6学时)

1. **知识点**：天气预报项目开发涉及的uni-app组件、API运用、天气预报数据来源和格式、天气图标和城市数据管理。
2. **重点**：天气预报项目页面布局和功能实现、数据请求与处理、天气图标展示和城市数据管理。
3. **难点**：处理不同来源天气数据格式、实现城市切换实时数据更新和页面动态刷新、优化数据请求性能。
4. **基本要求**：掌握天气预报项目开发技术，完成项目开发和优化。
5. **实践与练习**：开发天气预报项目，实现天气信息展示、城市管理等功能。

(九) 课程设计与考核验收 (4学时)

1. **知识点**：课程设计的目标、要求和流程、课程考核验收的标准和方法、项目成果展示与评价。
2. **重点**：课程设计任务拆解和项目规划、考核验收标准制定和实施。
3. **难点**：引导学生在课程设计中创新、确保考核验收公平客观。
4. **基本要求**：学生能根据课程设计的要求完成移动应用项目开发，掌握项目成果评价方法。
5. **实践与练习**：完成课程设计项目开发，参与项目成果展示和评价。

五、学时分配

章次	授课内容	理论学时	实践学时
第一章	初识uni-app	2	2
第二章	uni-app基础	3	3
第三章	uni-app基础内容扩展	3	3
第四章	uni-app组件	6	6
第五章	uni-app常用API(1)	5	5
第六章	uni-app常用API - 2	6	6
第七章	智慧环保项目	6	6
第八章	天气预报项目	3	3
第九章	课程设计与考核验收	2	2
合计			72

六、考查模式与成绩评定方法

1. **考核方式**：本课程采用多元化考核方式，注重过程性评价与终结性评价相结合。
2. **成绩评定**
 - **出勤 (10%)**：根据学生课堂出勤情况进行考核，无故缺勤将按比例扣除相应分数。

- **平时作业 (30%)**：包括实践作业、案例分析、项目阶段性成果等，考查学生对知识的掌握和应用能力。
- **期末项目 (60%)**：学生需完成一个综合性移动应用项目，涵盖课程所学知识点，根据项目功能实现、代码质量、界面设计、项目文档等方面进行评定。

七、教学建议

1. **实践与理论相结合**：在教学过程中，通过实际项目案例引入理论知识，让学生在实践中理解和掌握移动应用开发技术。理论讲解注重系统性，实践操作注重项目的完整性和综合性，提高学生解决实际问题的能力。
2. **注重创新思维培养**：鼓励学生在项目开发中尝试新的技术和设计理念，培养学生的创新思维和自主学习能力。引导学生关注行业前沿技术，激发学生的学习兴趣和创新热情。
3. **关注行业前沿**：定期更新教学内容，引入行业最新的移动应用开发技术、工具和项目案例。鼓励学生参与开源项目和实际项目开发，了解行业发展趋势，提高学生的就业竞争力。
4. **团队协作与交流**：组织学生进行小组项目开发，培养学生的团队协作精神和沟通能力。通过小组讨论、代码审查等方式，促进学生之间的知识共享和经验交流。
5. **思政元素融入**：在教学过程中融入思政元素，如通过介绍国产开发框架uni-app的发展，培养学生的民族自豪感和创新精神；在项目开发中强调代码规范和知识产权保护，培养学生的职业道德和法律意识。

八、建议使用教材及主要参考书

建议使用教材：《uni-app移动应用开发（微课版）》

九、教学资源

1. **教学平台**：建议使用在线教学平台（如学习通、腾讯课堂等），提供课程视频、教学课件、案例资料、在线测试等资源，方便学生自主学习和教师教学管理。
2. **实践工具**：推荐使用HBuilderX作为开发工具，同时配备微信开发者工具、Chrome浏览器等，方便学生进行项目开发、调试和预览。
3. **案例库**：建立丰富的移动应用开发案例库，涵盖不同类型和难度的项目案例，包括课程中涉及的智云翻译项目、仿网易云音乐播放器项目、智慧环保项目、天气预报项目等，供学生学习和参考。
4. **实验室资源**：配备计算机机房，安装所需软件工具，提供稳定网络环境，支持学生进行项目实践和测试。同时，准备一定数量的真机设备，方便学生进行移动端测试。

十、教学反思与改进

1. **教学反思**：教师在每章教学结束后，通过学生作业、课堂表现、项目实践成果等方面收集学生反馈，分析学生对知识的掌握情况和存在的问题。同时，教师自我反思教学方法、教学内容的合理性和有效性，总结教学过程中的经验和不足。
2. **教学改进**：根据教学反思结果，及时调整教学策略。针对学生理解困难的知识点，增加案例分析和实践练习；优化教学方法，提高学生课堂参与度；更新教学内容，确保与行业发展同步。同时，加强对学生自主学习能力的培养，引导学生学会查阅资料、解决问题。
3. **持续更新**：定期关注移动应用开发行业的发展动态，及时将新技术、新工具、新案例融入教学内容。鼓励教师参加行业培训和学术交流活动，提升自身专业水平，保证教学内容的前沿性和实用性。