

《.NET 平台企业级应用开发 (Enterprise application development on NET platform) 》教学大纲

一、课程基本信息

(一) **适用专业**: 本科计算机科学与技术专业

(二) **课程代码**: 3DX1008A

(三) **学分/课内学时**: 4 学分/64 学时

(四) **课程类别**: 专业教育

(五) **课程性质**: 选修/专业课

(六) **先修课程**: 《软件工程》、《数据结构》、《计算机科学导论》、《数据库原理及应用》、《算法分析与设计》、《操作系统原理》、《计算机组成原理》、《面向对象的程序设计方法》、《编译原理》等

(七) **后续课程**: 毕业设计等

二、课程教学目标

本课程是一门综合性、应用型学科。课程内容涵盖了.NET 开发技术的组成及核心知识体系, 基于.NET 的实用开发技术及技术原理解析, 在具体情境中.NET 开发技术的应用原理及具体编程实现, 程序的调试及异常处理, 项目设计以及实现的一般步骤、框架构建及不同类型的应用实现等内容。开设本课程的目的, 是适应培养应用型人才的需要, 使学生较为深入的理解.NET 开发技术的概念、组成以及内涵; 掌握.NET 核心、实用开发技术在不同情境中的具体应用及实现之法; 掌握程序调试及纠错的一般能力; 使得学生具备良好的编程规范, 提升学生软件系统开发的分析、设计以及实现能力。为学生后续的毕业设计打下基础。

本课程的主要任务是通过课堂教学、实验教学等环节培养学生的系统设计、分析与实现能力, 提升学生从原理到应用的综合实践能力, 完整的系统开发能力以及问题解决分析以及解决能力。课程目标及能力要求具体如下:

(一) 具体目标

目标 1: 掌握.NET 开发技术的基本体系结构、核心组成及基本工作原理。

目标 2: 能熟练配置和使用 VisualStudio 集成开发环境; 了解不同项目类型的特点及不同项目类型的基本框架组成, 能在开发环境中进行常见的应用程序配置、开发、调试与异常处理; 基本的程序调试能力、问题解决能力以及良好的编程规范。

目标 3: 掌握.NET 开发项目的基本实现方式、面向对象编程的内涵及在项目开发中的具体应用及实现; 掌握.NET 开发中的常用类库、实用技术、实现方

式及其在项目开发中的具体应用之法。

目标 4：能够分析、设计并实现基于.NET 开发技术的项目或系统。了解从软件需求到系统设计、再到基于.NET 开发技术的不同类型系统的设计以及实现框架搭建过程。具有将.NET 技术应用于具体项目中的能力。

(二) 课程目标与毕业要求的对应关系

毕业要求	毕业要求指标点	课程目标	教学单元	评价方式
1.专业必须的数学、自然科学、工程基础和专业知识，能够用于解决软件开发中的复杂工程问题。	指标 1.3：计算机软件与理论、计算机系统结构、计算机应用技术的基本理论、基本知识和基本技能	目标 1 目标 3	.NET 体系；.NET 程序基本结构；C# 编程基础；封装、继承和多态；.NET 的数据访问技术	作业 大作业 期中考核
2.能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析计算机软件系统中的复杂工程问题，以获得有效结论。	指标 2.2：能认识并判断软件系统开发过程的实际工程问题有多种解决方案	目标 2	面向对象编程特性在开发中的分析以及解决问题思路；.NET 常用编程技术的应用场景分析；.NET 技术开发模型（单例/工厂/MVC/分层等）的解决方案实现思路	实验
3.能够设计针对计算机应用系统设计中的复杂工程问题的解决方案，设计满足特定需求的系统，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。	指标 3.3：能够进行计算机应用系统的需求分析，方案设计、代码实现和系统集成、部署等方面进行方案优化，体现创新意识	目标 3 目标 4	.NET 开发技术在项目中的应用（基于泛型集合的项目设计及实施/基于分层的项目设计/基于 EF 技术的系统实现方案设计/基于 Web API 的项目设计及实现）	作业 大作业 期中考核
5.能够针对计算机应用系统的复杂工程问题，开发、选择于使用恰当的技术、资源、现代工程工具和信息技术工具，实现对复杂工程问题的预测与模拟，理解其局限性。	指标 5.1：能运用信息技术、大型团队开发过程管理软件等现代工程工具和信息技术工具，对计算机系统的设计开发过程进行管理	目标 4	基于.NET 开发技术的项目分析、选型、开发模型（桌面/Web/ Web API）、设计及实现方案搭建	大作业

三、教学内容与方法

(一) 教学内容及要求

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学	支撑教学	备注
----	------	---------------	--------	------	------	------	----

					方式	目标	
1	概述	<p>(1) .NET 体系结构 .NET 技术的含义及组成; C# 与.NET 的关系; .NET 技术的含义及组成; 程序集; NET 技术开发的工作原理</p> <p>(2) 集成开发环境 版本、安装、启动、导航、配置、环境说明及使用</p> <p>(3) 使用.NET 创建不同项目类型的应用程序。</p> <p>重点: .NET 体系结构; 使用 C# 创建.NET 不同项目类型的应用程序</p> <p>难点: .NET 体系结构; 使用 C# 创建.NET 不同项目类型的应用程序; 程序运行原理。</p>	<p>掌握: .NET 体系结构的组成, 各个组成部分的定义、内容及功能;.NET 技术 的工作原理; 不同编程语言在.NET 中的应用原理; 创建.NET 不同项目类型的应用程序。</p> <p>理解: .NET 体系结构的组成, 各个组成部分的工作原理; C# 程序运行的原理。</p> <p>了解: .NET 发展的历史。</p>	2	讲授 演示 实例	目标 1、2	
2	.NET 开发入门	<p>(1) .NET 程序基本结构 (命名空间类, 对象, 方法及数据)</p> <p>(2) .NET 项目的开发流程 (解决方案, 项目类型, 编程语言, 项目文件, 代码编制, 调试, 运行, 保存)</p> <p>(3) 程序的修改与调试</p> <p>(4) 基本编程规范 (代码的基本格式, 注释, 标识符, 关键字、命名、命名空间等)</p> <p>重点: 程序的基本组成结构; 基本的编程规范; 项目开发基本流程</p> <p>难点: 不同项目类型中, 各个文件及文件夹的含义。</p>	<p>掌握: .NET 中程序的基本组织结构; .NET 框架类库的基本使用方法; 简单项目最基本的实现流程及代码编制; 控制台应用程序的基本结构, 常见的输入输出控制语句; 基本编程规范。</p> <p>理解: .NET 框架类库的内涵; 项目文件及项目文件夹的固定含义。</p> <p>了解: 面向对象编程特点及其含义。</p>	2	讲授 演示 案例	目标 1、3	

3	C# 编程基础	<p>(1) 预定义数据类型 (值类型和引用类型, 简单数据类型的显示和隐式转换, 常量与变量)</p> <p>(2) 运算符与表达式</p> <p>(3) 枚举与结构</p> <p>(4) 流程控制 (顺序, 选择, 循环, 嵌套控制语句)</p> <p>(5) 数组</p> <p>(6) 程序调试 (设置断点, 单步调试, 观察变量, 条件断点)</p> <p>重点: 数据类型; 表达式; 流程控制; 数组; 程序的调试方法</p> <p>难点: 数据类型转换; 数组的应用; 程序的调试方法</p>	<p>掌握: C#中的数据类型分类, 数据类型转换方法; 常量变量的定义及使用方法; 流程控制语句的语法及表现形式; 掌握数组的定义及使用方法; 掌握程序调试方法。</p> <p>理解: C#中的数据类型, 理解计数器控制循环和标记循环的内涵及选用标准;</p> <p>了解: C#中数据类型与其它编程语言中数据类型的异同。</p>	8	讲授 案例 演示 归纳	目标 2、3	
4	面向对象编程	<p>(1) 面向对象编程内涵</p> <p>(2) 类定义、实现及成员 (数据成员; 方法成员; 构造方法的含义、定义、初始化、作用及重载; 声明带方法的类并实例化类的对象)</p> <p>(3) 类中方法成员 (方法的定义及调用, 声明不带参数的方法, 多个参数方法的定义及使用, 方法重载, 方法中高级参数的使用)</p> <p>(4) 属性(含义, 定义, 使用)</p> <p>(5) 访问修饰符</p> <p>(6) Object 类 (含义及使用)</p> <p>(7) 对象的定义及使用</p> <p>重点: 类、类成员的定义及使用, 方法重载, 访问修饰符, 方法中高级参数的使用。属性的含义及使用。</p> <p>难点: 属性的含义及实现, 方法中高级参数的使用, 方法重载。</p>	<p>理解: 类与对象的含义, 属性的作用, 类成员分类, 构造方法的作用, 重载的含义, 访问修饰符的作用范围。</p> <p>掌握: 类的定义与实例化对象的方法; 构造方法的定义方法; 类成员的定义及访问方法; 方法重载的实现, 带不同参数个数的方法成员定义及调用方法; 方法所带参数的定义及使用方法。</p> <p>了解: Object 类, 常用类库中的类。</p>	4	讲 讲 授 授 案 案 例 例 演 演 示 示 归 归 纳 纳	目标 3	
5	封装继承	<p>(1) 封装含义及实现</p> <p>(2) 继承 (定义, 类型, 实现:</p>	<p>理解: 继承和多态的含义, 抽象类的含</p>	8	讲授 案例	目标 2、3	

	和多态	<p>虚方法, 隐藏方法, 抽象类和抽象方法, 密封类和密封方法, 派生类)</p> <p>(3) 接口</p> <p>(4) 多态的不同实现</p> <p>(5) 命名空间与程序集 (定义, 添加及引用)</p> <p>(6) 异常处理机制。</p> <p>重点: 继承和多态的实现; 接口的应用; 异常处理机制及实现, 命名空间与程序集</p> <p>难点: 继承的含义及实现, 多态的含义及实现, 异常处理机制在编程中的实现。</p>	<p>义, 接口的含义, 异常处理机制;命名空间与程序集的含义。</p> <p>掌握:继承和多态的特性,继承和多态的使用,抽象类和接口的使用,里氏转换原则,异常的处理方法及代码中如何添加异常处理。</p>		演示归纳		
6	.NET 常用类库和正则表达式	<p>(1) string 类与 stringBuilder</p> <p>(2) DateTime 类与 TimeSpan</p> <p>(3) Random 类</p> <p>(4) Math 类</p> <p>(5) 正则表达式</p> <p>重点: string 类与 stringBuilder 类的使用; DateTime 类的使用; 随机数的生成与使用; 正则表达式</p> <p>难点: 使用 string 类与 stringBuilder 类进行字符串操作; 不同范围内的随机数生成; 正则表达式。</p>	<p>掌握: String 类和 StringBuilder 类使用; 掌握 DateTime 类的基本使用方法; 掌握 Random 类的使用方法;正则表达式的应用方法。</p> <p>理解:字符串的不可变性, string 类与 stringBuilder 类区别。</p> <p>了解: TimeSpan 类使用。</p>	4	讲授案例演示归纳	目标 3	
7	.NET 常用编程技术	<p>(1) 集合的基本概念及类型; 集合的继承关系; 集合长度和集合遍历</p> <p>(2) 泛型(引入, 定义, 含义, 作用及使用) 自定义泛型 (类型形参, 泛型方法的定义及调用, 类型约束); 泛型委托</p> <p>(3) 泛型集合及在项目中的应用</p> <p>(4) 委托、匿名方法和事件</p> <p>重点: 集合, 泛型集合的定义及使用; ArrayList 集合的定义及使用; 泛型集合 List 在项目中的应用; Hashtable 在项目开发中的应用; 泛型方法的定义及调用。</p>	<p>掌握: 集合的添加、删除、修改及查询元素的编程方法;掌握泛型集合的应用技巧;泛型方法的定义及使用;泛型类型约束的使用方法。</p> <p>理解: 集合、泛型集合和非泛型集合的含义;理解集合类和接口的继承关系;泛型委托的概念。</p> <p>了解: Framework 的其它泛型类型。</p>	12	讲授案例演示归纳	目标 3、4	

		<p>难点： 泛型集合 List 在项目中的应用； 泛型委托。</p>				
8	.NET 应用程序设计与开发	<p>(1) Windows 窗体应用程序 (含义, 创建, 基本框架及组成) (2) 开发 Windows 应用程序的步骤 (3) Windows 标准控件的使用 (4) 窗体 (Form 类, 多文档界面, 不同窗体之间传参, 设置启动窗体, 窗体的常用方法和属性) (5) Windows 窗体应用程序案例分析及编程实现</p> <p>重点： Windows 窗体应用程序的特点及基本框架；设计与开发 Windows 应用程序的步骤；Windows 标准控件的使用及编程；窗体 Form 类的常用方法、属性和事件操作。</p> <p>难点： Windows 窗体高级控件的使用及编程；窗体之间的参数传递问题。</p>	<p>掌握：Windows 窗体应用程序的含义, 基框架及组成, 程序的运行过程；掌握 Windows 窗体应用程序开发时的常用控件的作用及编程方法, 能根据功能选用合适的控件；掌握窗体的设计方法及不同窗体之间的转换与传参方法。</p> <p>理解：Windows 窗体应用程序的框架及组成。理解不同窗体之间传参所依据的原理。理解不同控件的作用。</p> <p>了解：控件定制与多文档界面的设计。</p>	4	讲授案例演示归纳	目标 3、4
9	LINQ 技术	<p>(1) LINQ 基础 (概述, 查询, 查询的各个子句及其含义, Lambda 表达式的使用) (2) LINQ 查询表达式 (获取数据源, 筛选, 排序, 分组, 联接, 选择) (3) LINQ 操作数据源。</p> <p>重点： LINQ 查询表达式的编制, 对数据的查询操作；Lambda 表达式的使用。</p> <p>难点： LINQ 查询表达式的定制, 操作数据库的编程, Lambda 表达式的使用。</p>	<p>掌握：LINQ 的基础概念及基本查询操作方法；掌握 LINQ 查询表达式的定制；掌握 LINQ 操作数据库的方法；掌握 Lambda 表达式的使用方法。</p> <p>理解：LINQ 的含义及优点。</p> <p>了解：LINQ 的高级操作。</p>	2	讲授案例演示归纳	目标 3
10	设计模式	<p>(1) 设计模式概述 (2) 单例设计模式及实现 (3) 简单工厂模式及实现</p>	<p>掌握：设计模式的分类及部分设计模式在.NET 中的实现。</p>	4	讲授案例演示	目标 3

		(4) 责任链模式及实现 重点： 设计模式的含义及在项目开发中的意义；不同类型的设计模式需解决的实际问题侧重点，以及如何解决实际问题；不同类型设计模式在 NET 中的实现及应用 难点： 不同类型设计模式的特点及具体实现	理解：不同设计模式的特点及不同。 了解：各种不同设计模式在项目中的应用场景		归纳		
11	.NET 技术开发模型	(1) Razor Pages Web 应用的工作原理及项目方案搭建 (2) .NET MVC 在案例中的应用及方案搭建 (3) Web API 项目工作原理及实现方案搭建 (4) 基于 WebAPI+EF+关系型数据库的项目实现方案搭建 重点： 不同类型项目的设计及实现方案搭建	掌握：常见的 .NET 技术项目开发的类型以及基本框架搭建方法； 理解：.NET 不同应用程序的框架、设计及实现。 了解：基于 MVC \WebAPI\WebAPI+EF 等框架的项目的工作原理以及基本搭建思路	14	讲授 案例演示 归纳	目标 3、4	
总计				64			

(二) 教学方法

(1) 主要采用案例式教学。通过案例激发学生学习的兴趣，培养学生系统思考、分析问题和解决问题的能力，引导学生主动通过应用实践加深对知识的理解以及应用能力。

(2) 问题导引教学。针对知识点，提出该知识需要解决的问题，通过问题导引，发现问题，分析问题，解决问题，一步步启发学生依据科学的、规范的理论基础对问题进行思考并解决。

(3) 多媒体教学资源的灵活应用，提高课堂教学信息量，增强教学的直观性与生动性。

(4) 理论教学与工程实践相结合，引导学生综合应用所学的专业知识，采用科学的方法和手段，进行项目设计与实现。

(5) 启发式教学。设定教学思路，引导学生在指定的讨论范围内，以科学、有效的思路寻找问题的求解过程，并能选择合理的技术灵活的应用在问题求解过程中。

四、考核及成绩评定

(一) 考核内容及成绩构成

课程考核以考核学生能力培养目标的达成为主要目的,以检查学生对各知识点的掌握程度和应用能力为重要内容,包括平时作业、实验、期中考核及期末大作业四个部分,考核方式:考查。各课程目标的考核内容、成绩评定方式、目标分值建议如下:

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况计算公式
目标 1: 掌握.NET 开发技术的基本体系结构、核心组成及基本工作原理。	.NET 体系结构; 程序运行原理及工作过程; 程序基本组成结构;	期中考核	5%	17%	B ₁	$\frac{B_1}{17\%} \times 5\%$
目标 2: 能熟练配置和使用 VisualStudio 集成开发环境; 了解不同项目类型的特点及不同项目类型的基本框架组成, 能在开发环境中进行常见的应用程序配置、开发、调试与异常处理; 养成良好的编程规范, 基本的程序调试能力与问题解决能力。	VisualStudio 集成开发环境; 应用程序的创建、编制、调试; 异常的处理; 基本类库的应用; 不同项目类型的选择与应用; 基本的程序调试能力以及错误处理能力	实验	10%	100%	C ₁	$\frac{C_1}{100\%} \times 10\%$ 10%
目标 3: 掌握.NET 开发技术的基本实现方式、面向对象编程的内涵及在项目开发中的具体应用及实现; 掌握.NET 开发中的常用类库、实用技术、实现方式及其在项目开发中的具体应用之法。	面向对象内涵及实现; 数据成员的初始化; 封装、继承、多态的实现; 集合、泛型、事件的含义及实现; 常用类库应用; 异常处理等	作业	10%	100%	A ₂	$\frac{A_2}{100\%} \times 10\% + \frac{B_2}{83\%} \times 25\% + \frac{D_1}{40\%} \times 20\%$ 55%
		期中考核	25%	83%	B ₂	
		大作业	20%	40%	D ₁	

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况计算公式
目标 4: 能够设计并实现基于.NET 开发技术的项目或系统。了解由软件需求, 到系统设计、再到基于.NET 开发技术的系统实现过程。具有将.NET 技术应用于具体项目中的能力	基于.NET 开发技术的项目或系统的构建; 不同项目类型的创建; 基于框架设计的项目设计及实现; 项目的完整实现过程	大作业	30%	60%	D ₂	$\frac{D_2}{60\%} \times 30\%$ 30%
总评成绩 (100%) = 作业 (10%) + 实验 (10%) + 期中考核 (30%) + 大作业 (50%)			100%	—	—	$\frac{\text{学生总评平均分}}{100}$

(二) 平时考核成绩评定

实验: 必做实验 10 次, 8 次单元实验, 2 次综合实验。支撑目标 2, 共占总评分 10%, 对应目标的评分标准如下:

对应目标	目标 2: 能熟练配置和使用 VisualStudio 集成开发环境; 了解不同项目类型的特点及不同项目类型的基本框架组成, 能在开发环境中进行常见的应用程序配置、开发、调试与异常处理; 养成良好的编程规范, 基本的程序调试能力与问题解决能力。		
考查点	实验过程	实验代码	实验问题解决
成绩比例	40%	40%	20%
评分标准	100% 至 90%	能依据软件工程开发思想, 合理组织实验过程与步骤, 实现方案设计合理、步骤完整、能合理预估各个阶段的时间分配、按照阶段设计合理实现各个过程, 且编程能力优秀。	代码结构清晰、能科学、有效实现实践要求、能灵活应用所学技术对问题进行求及实现、代码运行无误且有有效性验证、异常处理; 有一致、规范的编程表达形式, 有完整的代码注释, 便于阅读。
			能够基于实验结果准确分析和评价现有研究成果存在的问题与不足, 针对全部问题能够提出有效的问题解决方法或者优化方法。基于问题实现, 能提出科学的创新实现。

89.9% 至 80%	能依据软件工程开发思想，能良好的组织实验过程与步骤，实现方案设计合理、能良好预估各个阶段的时间分配、按照阶段设计良好实现各个过程，编程能力良好。	代码结构良好，能良好的综合应用所学表示及实现问题所需，代码能运行，具备良好的输入有效性验证；有良好的编程规范与注释	针对实验过程中的问题，具备问题分析与解决的良好能力。具备良好的程序优化与创新能力。
79.9 至 70%	能依据软件工程开发思想，较好的组织实验过程与步骤，实现方案设计较为合理、能较好的预估各个阶段的时间分配、按照阶段设计较好的实现各个过程，编程能力较好。	代码结构组织较为规范，能较好的综合应用所学表示及实现问题所需，代码能较好的运行，具备较好的输入有效性验证；有较好的编程规范与注释	能够基于实验结果较准确分析和评价现有实验成果的问题与不足，针对部分问题能够提出问题解决办法与优化思路。
69.9% 至 60%	能依据软件工程开发思想，基本能完成实验过程与步骤的组织，实现方案设计基本合理、基本上能按照设计进行代码实现。	代码结构组织基本规范，基本上能应用所学表示及实现问题所需，代码能运行，具备简单的输入有效性验证；具备基本的编程规范	基本能够分析或发现现有实验成果的问题与不足，无法针对问题提出解决办法与优化思路。
59.9% 至 0	无法依据软件工程开发思想完成实验过程与步骤的组织，实现方案设计不合理、无法有效综合应用所学知识进行问题求解。	代码结构组织混乱，无法应用所学表示及实现问题所需，代码无法运行，且不符合编程规范	无法分析或发现现有实验成果的问题与不足，对出现的问题，无法提出有效的解决思路与办法

五、参考学习资料

(一)推荐教材:《ASP.NET 项目实战教程——从.NET Framework 到.NET Core》，徐照兴等，清华大学出版社，2024年，ISBN: 9787302657552。

(二)参考资料:《C#入门经典》，(英)马克·J.普赖斯等，清华大学出版社，2023年，ISBN: 9787302583882

(三)参考资料:《C#.NET 项目开发案例教程》，陈建国等，清华大学出版社，2023，ISBN: 9787302594307

制定人: 刘海玲

审核人: 雷亮

制定时间：2023 年 9 月