**电子科技大学**

**计算机科学与工程学院**

**标 准 实 验 报 告**

**（实验）课程名称 软件开发综合实验**

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告**

|  |  |  |
| --- | --- | --- |
| **小组： XXX** | **组长（学号）、组员（学号）、组员（学号）** | |
| **组长姓名：** | | **学号：** |
| **组员姓名：** | | **学号：** |
| **组员姓名：** | | **学号：** |
| **一、实验室名称：**主楼A2-413-1 | | |
| **二、实验项目名称：**文件备份软件 | | |
| **三、实验目的：**  本课程的教学目标是为学生提供理论与实践相结合的基础平台，以计算机科学与技术专业必修课程《程序设计（C与C++）》、《数据结构与算法》和《软件工程》三门课程所涉及的主要知识为核心，基于具体的工程项目，加深学生对理论知识的理解，使学生能够解决计算机系统的复杂工程问题，达到选择或设计恰当的逻辑结构、存储结构及相应的算法的培养目标，培养学生在实际软件生产开发过程中对其所学知识的综合运用能力，提升学生的计算思维能力、算法设计与分析能力、程序设计与编程实现能力、团队协作能力、计算机系统的认知、分析、设计和运用能力，工程实践能力。 | | |
| **四、实验内容：**  **实验内容：**   1. 设计并实现一款数据备份软件，以项目组形式推进，每组最多三人。 2. 基于软件工程方法学进行项目推进，经历从需求分析、系统设计、编码实现、软件测试的整个软件生命周期。 3. 实验最终成果包括一款基本可用的软件及其对应文档。 4. 软件应包括指明的完整功能，重点考察其正确性、易用性、健壮性。 5. 软件文档应包括：需求分析说明书、系统设计文档、软件测试报告，重点考察其规范性、一致性、可读性。 6. 采用现代化软件开发工具辅助项目开发，包括但不限于：项目管理工具，UML建模工具，集成开发环境，版本控制工具，软件测试工具。   该实验包括必选的基本要求和可任意选择的扩展要求：  **基本要求：**  各小组“独立”实现一款数据备份软件（对应基础分总分40分）：  数据备份：将目录树中的文件保存到指定位置  数据还原：将目录树中的文件恢复到指定位置  开发语言选择：C/C++/Java/Go等语言；用户界面可以采用脚本语言编写；后台逻辑若也选择脚本语言，则小组基础分记0分。  **扩展要求：**  各项目组根据自身情况自行选择扩展要求（对应扩展分总分）。  文件类型支持（10分）：支持特定文件系统的特殊文件（管道/软链接/硬链接等）  元数据支持（10分）：支持特定文件系统的文件元数据（属主/时间/权限等）  自定义备份（各5分）：允许用户筛选需备份的文件（路径/类型/名字/时间/大小）  压缩解压（10分）：通过文件压缩节省备份文件的存储空间  打包解包（10分）：将所有备份文件拼接为一个大文件保存  加密备份（10分）：由用户指定密码，将所有备份文件均加密保存  实时备份（15分）：自动感知用户文件变化，进行自动备份  定时备份（10分）：允许用户进行周期性定时备份和数据淘汰  图形界面（10分）：实现友好易用的GUI界面  网络备份（30分）：将数据备份软件从单机模式扩展为网盘模式（10分），还涉及到的功能包括：用户管理（5分）、元数据管理（5分）、传输加密（5分）、增量备份（5分）等。  其它功能：视功能难度讨论加分。 | | |
| **五、实验器材（设备、元器件）：**  PC多台，可连接互联网。 | | |
| **六、实验步骤及操作：** | | |
| **需求分析说明书（10分）**  **1. 任务概述**  任务概述是对这份软件产品需求分析报告的概览，是为了帮助阅读者了解这份文档是如何编写的，并且应该如何阅读、理解和解释这份文档。  **1.1 引言**  **1.2 综合描述**  这一部分概述了正在定义的软件产品的作用范围以及该软件产品所运行的环境、使用该软件产品的用户、对该软件产品己知的限制、有关该软件产品的假设和依赖。  **1.2.1 产品的状况**  描述了在软件产品需求分析报告中所定义的软件产品的背景和起源。说明了该软件产品是否属于下列情况：   * 是否是产品系列中的下一成员； * 是否是成熟产品所改进的下一代产品； * 是否是现有应用软件的替代品(升级产品)； * 是否是一个新型的、自主型的产品。   如果该软件产品需求分析报告定义的软件系统是：   * 大系统的一个组成部分； * 与其它系统和其它机构之间存在基本的相互关系。   那么必须说明软件产品需求分析报告定义的这部分软件是怎样与整个大系统相关联的，或者（同时）说明相互关系的存在形式，并且要定义出两者之间的全部接口。  **1.2.2 产品的功能**  因为将在需求分析报告的第5部分中详细描述软件产品的功能，所以在此只需要概略地总结。仅从业务层面陈述本软件产品所应具有的主要功能，在描述功能时应该针对每一项需求准确地描述其各项规格说明。如果存在引起误解的可能，在陈述本软件产品主要功能的作用领域时，也需要对应陈述本软件产品的非作用领域，以利读者理解本软件产品。  为了很好地组织产品功能，使每个读者都容易理解，可以采用列表的方法给出。  **1.2.3 用户类和特性**  确定有可能使用该软件产品的不同用户类，并且描述它们相关的特征。往往有一些软件需求，只与特定的用户类有关。描述时，应该将该软件产品的重要用户类与非重要用户类区分开。  用户不一定是软件产品的直接使用者，通过报表、应用程序接口、系统硬件接口得到软件产品的数据和服务的人、或者机构也有他们的需求。所以，应该将这些外部需求视为通过报表、应用程序接口、系统硬件接口附加给软件产品的附加用户类。  **1.3 运行环境**  **1.3.1 基本配置**  文件备份软件系统所需的基本配置如下：  （1）硬件平台：  （2）操作系统：  **1.3.2 其他配置**  （1）内存：  （2）硬盘：  （3）网络：  **2. 功能需求**  需要进行详细的需求记录，详细列出与该系统功能相关的详细功能需求，并且，唯一地标识每一项需求。这是必须提交给用户的软件功能，使得用户可以使用所提供的功能执行服务或者使用所指定的使用实例执行任务。描述软件产品如何响应己知的出错条件、非法输入、非法动作。  **2.1 功能划分**  根据具体的用户业务对系统进行功能划分，便于后续需求分析。  **2.2 系统用例**  **2.2.1 用例编号xxx**  **2.2.1.1 用例图**  用例一（例）    **2.2.1.2 用例描述**  用例图只是简单地用图描述了一下系统，但对于每个用例，我们还需要有详细的说明，这样就可以让别人对这个系统有一个更加详细的了解，这时我们就需要写用例描述。  对于用例描述的内容，一般没有硬性规定的格式，但一些必须或者重要的内容还是必须要写进用例描述里面的。用例描述一般包括：简要描述（说明）、前置（前提）条件、基本事件流、其他事件流（异常事件流）、后置（事后）条件等等。下面说说各个部分的意思：  简要描述：对用例的角色、目的的简要描述；  前置条件：执行用例之前系统必须要处于的状态，或者要满足的条件；  基本事件流：描述该用例的基本流程，指每个流程都“正常”运作时所发生的事情，没有任何备选流和异常流，而只有最有可能发生的事件流；  其他事件流：表示这个行为或流程是可选的或备选的，并不是总要总要执行它们；  异常事件流：表示发生了某些非正常的事情所要执行的流程；  后置条件：用例一旦执行后系统所处的状态；）  用例一用例描述（例）   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 用例标识 | | xxx | 用例名称 | Create a new Regular Blog Account | | 创建人 | | xxx | 创建日期 | 2020年11月2日 | | 语境目标 | | 一位新的或现有作者向管理员请求一个新的普通博客账户 | | | | 前置条件 | | 作者需有适当的身份证明 | | | | 成功的结束状态 | | 成功创建该作者的新普通博客账户 | | | | 失败的结束状态 | | 该作者的新普通博客账户申请被拒绝 | | | | 参与者 | | Administrator | | | | 触发器 | | 管理员要求CMS建立一个新的普通博客账户 | | | | 基础用例 | | Create a new Blog Account | | | | 包含用例 | | Check Identity | | | | 扩展用例 | | Record Application Failure | | | | 事件流 | 基本流程 | 1、管理员要求系统建立一个新的博客账户 | | | | 2、管理员选择普通博客账户类型 | | | | 3、管理员输入作者的详细数据 | | | | 4、检查该作者的详细数据 | | | | include::Check Identity | | | | 5、成功创建新的博客账户 | | | | 扩展流程 | 4.1、不允许该作者建立新的博客 | | | | 4.2、博客账户申请被拒绝 | | | | 4.3、此申请失败情况被记录在该作者的历史数据里 | | | | extend::Record Application Failure | | | | 非功能需求 | | 安全性 | | | | 补充规格说明书 | | 无 | | |   **2.2.1 用例编号xxx**  **2.2.1.1 用例图**  **2.2.1.2 用例描述**  **3. 外部接口需求**  通过本节描述可以确定，保证软件产品能和外部组件正确连接的需求。  注意：必须将附加用户类的特征与外部接口需求加以区分，附加用户类的特征描述的是通过接口取得软件产品的数据和服务的人的需求；而外部接口需求描述的是接口本身的需求。  **3.1 用户界面**  用户界面可以是命令行界面或者图形界面。  陈述需要使用在用户界面上的软件组件，描述每一个用户界面的特征。以下是可能包括的一些特征：   * 将要采用的图形用户界面(GUl)标准或者产品系列的风格； * 有关屏幕布局或者解决方案的限制； * 将要使用在每一个屏幕(图形用户界面)上的软件组件，可能包括：   + 选单；   + 标准按钮；   + 导航链接；   + 各种功能组件；   + 消息栏； * 快捷键； * 各种显示格式的规定，可能包括：   + 不同情况下文字的对齐方式；   + 不同情况下数字的表现格式与对齐方式   + 日期的表现方法与格式；   + 计时方法与时间格式；   + 等等。 * 错误信息显示标准；   对于用户界面的细节，例如：一个特定对话框的布局，应该写入具体的用户界面设计说明中，而不能写入软件需求规格说明中。  如果采用现成的、合适的用户界面设计规范(标准)，或者另文描述，可以在这里直接说明，并且将其加入参考文献。  **3.2 硬件接口**  描述待开发的软件产品与系统硬件接口的特征，若有多个硬件接口，则必须全都描述。接口特征的描述内容可能包括：   * 支持的硬件类型； * 软、硬件之间交流的数据； * 控制信息的性质； * 使用的通讯协议；   **3.3 软件接口**  描述该软件产品与其它外部组件的连接，这些外部组件必须明确它们的名称和版本号以资识别，可能的外部组件包括：   * 操作系统； * 数据库； * 工具； * 函数库； * 集成的商业组件   说明：这里所说的“集成的商业组件”，是指与系统集成的商业组件，而不是与软件产品集成的商业组件。例如：中间件、消息服务，等等。  描述并且明确软件产品与软件组件之间交换数据或者消息的目的。描述所需要的服务，以及与内部组件通讯的性质。确定软件产品将与组件之间共享的数据。如果必须使用一种特殊的方法来实现数据共享机制，例如：在多用户系统中的一个全局数据区，那么就必须把它定义为一种实现上的限制。  **4．其它非功能性需求**  **4.1 性能需求**  阐述不同应用领域对软件产品性能的需求，并且说明提出需求的原理或者依据，以帮助开发人员做出合理的设计选择。尽可能详细地描述性能需求，如果需要，可以针对每个功能需求或者特征分别陈述其性能需求。在这里确定：   * 相互合作的用户数量； * 系统支持的并发操作数量； * 响应时间； * 与实时系统的时间关系： * 容量需求   + 存储器；   + 磁盘空间；   + 数据库中表的最大行数。   **4.2 安全性需求**  详尽陈述与系统安全性、完整性问题相关的需求，或者与个人隐私问题相关的需求。这些问题将会影响到软件产品的使用，和软件产品所创建或者使用的数据的保护。定义用户身份认证，或备授权需求。明确软件产品必须满足的安全性或者保密性策略。也可以通过称为完整性的质量属性来阐述这些需求。一个典型的软件系统安全需求范例如下：“每个用户在第一次登录后，必须更改他的系统预置登录密码，系统预置的登录密码不能重用。”  **4.3 软件质量属性**  详尽陈述对客户和开发人员至关重要的在软件产品其它方面表现出来的质量功能。这些功能必须是确定的、定量的、在需要时是可以验证的。至少也应该指明不同属性的相对侧重点，例如：易用性优于易学性，或者可移植性优于有效性。  **5．项目规划**  **5.1 人员分工**  项目由三位成员分工具体安排如下（例）。   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **角 色** | **主要职责** | **负责模块** | **人员** | **备注** | | 项目经理PM | * 项目全面负责 * 项目设计 * 主要框架/模块编写 * 项目进度控制 | * 代码框架 * 代码汇总 | xxx |  | | 程序员DEV | * 模块编写 | * 打包\解包模块 * 校验模块 | xxx |  | | 程序员DEV | * 模块编写 | * 压缩\解压 | xxx |  |   **5.2 项目总体规划**  绘制甘特图并进行说明（例）。 | | |
| **系统设计文档（20分）**  **1.** **开发环境和工具**  **1.1 开发环境**  （1）硬件平台：  （2）操作系统：  （3）编译环境：  （4）IDE：  ……  **1.2 依赖库**  （1）QT：版本5.8，用于图形界面编程。  ……  **2. 总体设计**  **2.1 系统结构设计**  这部分要求提供高层系统结构的描述，使用构件图来显示主要的组件及组件间的交互。  **2.1.1 顶层系统结构**  **2.1.1.1 构件图**  系统构建图（例）  **图示  描述已自动生成**  **2.1.1.2 构件图描述**  **2.1.2 子系统一结构**  **2.1.2.1 构件图**  子系统一构建图（例）    **2.1.2.2 构件图描述**  **2.1.3 子系统xxx结构**  **2.1.3.1 构件图**  **2.1.3.2 构件图描述**  **3. 静态建模**  这部分的作用是描述系统的静态特性。一般使用类图和对象图。  **3.1 系统对象模型**  提供整个系统的类（对象）模型，即提供系统类图或者对象图。如果模型过大，按照可行的标准把它划分成小块，例如可以把客户端和服务器端的对象模型分开成两个图表述。  所有对象之间的关联必须被确定并且必须指明联系的基数（一对一、一对多还是多对多，0..1，\*，1..\*）。聚合和继承关系必须清楚地确定下来。每个图必须附有简单的说明。  可能经过多次设计的迭代之后才能得到系统的正确的对象模型。  **3.2 类（对象）描述**  在这个部分叙述每个类（对象）的细节，它的属性、它的方法。在这之前必须从逻辑上对对象进行组织。你可能需要用结构图把对象按子系统划分好。  为每个类（对象）做一个条目。在系统类（对象）模型中简要的描述它的用途、约束（如只能是单例），列出它的属性和方法。  对每个类（对象）的每个属性详细说明：名字、类型，如果属性不是很直观或者有约束（例如，每个对象的该属性必须有一个唯一的值或者值域是有限正整数等）。  对每个类（对象）的每个方法详细说明：方法名，返回类型，返回值，参数，用途以及使用的算法的简要说明（如果不是特别简单的话）。如果对变量或者返回值有什么假定的话，Pre-conditions和Post-conditions必须在此说明。列出它或者被它调用的方法需要访问或者修改的属性。最后，提供可以验证实现方法的测试案例。  **3.2.1 XXX类（对象）**  **3.2.1.1 类描述**  对类的作用、功能，对象的生存周期等信息进行描述。  **3.2.1.2属性描述**  以表格或列表形式对类或对象属性进行描述。  **3.2.1.2方法描述**  以表格或列表形式对类或对象方法进行描述。  **3.3 其它静态建模方式（可选）**  该部分可以包括其它的UML静态建模方法和图表。  **4. 动态建模**  这部分的作用是描述系统如何响应各种事件。例如，可以建立系统的行为模型。一般使用顺序图和状态图。  确定不同的场景（Scenario），不需要确定所有可能的场景，但是必须至少要覆盖典型的系统用例。  **4.1 系统场景**  对每个场景做一则条目，包括以下内容：   * 场景名：给它一个可以望文生义的名字 * 场景描述：简要叙述场景是干什么的以及发生的动作的顺序。 * 顺序图：描述各种事件及事件发生的相对时间顺序。   **4.1.1 场景：场景一**  **4.1.1.1 场景描述**  针对系统用例来确定场景，并对该场景进行描述，需要确定系统对象间和对象内的交互逻辑。  **4.1.1.2 顺序图**  场景一顺序图（例）    **4.1.1.3 流程说明**  对顺序图中的每一步进行详细说明：  1.……  2.……  **4.2 系统状态**  这部分的内容包括系统动态模型重要的部分的状态图。可能你想为每个对象画一个状态图，但事实上会导致太多不期望的细节信息，只需要确定系统中一些重要的对象并为之提供状态图即可。  **4.2.1 XXX状态切换**  **4.2.1.1 状态描述**  对该状态特征所对应的系统业务状态进行描述。  **4.2.1.2 状态转换图**  状态转换图一（例）    **4.2.1.3 状态图说明**  对状态转换图中的每一个状态和状态转换过程进行详细说明：  1.……  2.……  **5. 数据库设计（如有）**  **5.1 引言**  说明这份数据库设计说明书是为哪份软件产品编写的，开发这个软件产品意义、作用以及最终要达到的意图。通过这份数据库设计说明书详尽准确地描述了该软件产品的数据库结构。如果这份数据库设计说明书只与整个系统的某一部分有关系，那么只定义数据库设计说明书中说明的那个部分或子系统。  **5.2 数据库逻辑设计**  数据库设计人员根据《软件需求分析报告》，创建与数据库相关的实体关系图(E-R图)。如采用面对对象的分析和设计方法，则此处的实体相当于类。  在此处，应给出逻辑设计的完整的E-R图。  **5.3 数据库物理设计**  在此处应给出完整的数据库物理结构E-R图。开发者应根据逻辑设计的结果，进行数据库的物理设计，并对表结构进行规范化处理(第一范式，第二范式，第三范式)。  **5.4 基表设计**  每个基表采用一张表格进行描述，其格式如下：   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 数据库编号： | | | | | | 基表编号： | | | | | | 基表英文名称： | | | | | | 基表中文名称： | | | | | | 字段编号 | 英文字段名 | 中文字段名 | 字段类型 | 备注 | |  |  |  |  |  | | 说明： | | | | |     **5.5 视图设计**  每个视图采用一张表格进行描述，其格式如下：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 数据库编号： | | | | | | | 视图编号： | | | | | | | 视图英文名称： | | | | | | | 视图中文名称： | | | | | | | 相关基表和视图： | | | | | | | 字段编号 | 英文字段名 | 中文字段名 | 字段类型 | 字段源 | 备注 | |  |  |  |  |  |  | | 说明： | | | | | |     **5.6 索引设计**  每个数据库的所有采用一张表格进行描述，其格式如下：   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 数据库编号： | | | | | | 索引编号 | 基表名称 | 索引名称 | 字段集名称 | 备注 | |  |  |  |  |  | | | |
| **软件测试报告（10分）**  **1. 引言**  为了尽可能的找出软件的不足，提高软件的质量，促进软件的成功验收，专门制定了本大纲。其主要目的在于为所要进行的测试工作制定各种必要的准则和规范，以及在有关方面协议的基础上对测试工作进行合理组织与管理。  **2. 功能测试**  检查数据在流程中各个阶段的准确性。对系统中每一模块利用实际数据运行，将其结果与同样数据环境下应该得出的结果相比较，或与软件需求规格说明书中要求的结果进行比较，如有偏差，则功能测试不能通过。  检查软件需求规格说明书中描述的需求用例是否都得到满足；系统是否缺乏软件需求规格说明书中规定的重要功能；以及系统实际使用中不可缺少而软件需求规格说明书中没有规定的功能。  如果存在遗产数据，应该检查遗产数据转换是否正确。  **2.1 xxx模块**  测试用例如下表所示：   |  |  |  | | --- | --- | --- | | **程序版本号：**2.6.23 | | **模块名：**用户登录页面 | | **测试用例编号：**TestCase-PAGE-01 | | **用例级别：**重要 | | **用例名称：**用户登陆 | | **测试时间：**2022/3/5 | | **预置条件：**用户已经注册成功 | | | | **测试输入：**用户名和密码 | | | | **操作步骤** | 1. 直接访问系统，首次进入直接跳转到登录/注册页面： 2. 步骤2 3. 步骤3 | | | **预期结果**：登陆成功 | | | | **实际输出：**登陆成功 | | | | **测试人员：**XXX | | |   **2.2 xxx模块**  **3. 代码测试（可选）**  仅对系统关键模块的源代码进行抽查，检查模块代码编写的规范性，批注的准确性，是否存在潜在性错误，以及代码的可维护性。包括：命名规范检查、注释检查、接口检查、数据类型检查、限制性检查。  推荐使用lint工具对代码进行全面静态分析，并给出代码检查结果。针对无法修改的告警，给出对应理由。  **4. 性能测试（可选）**  根据系统设计指标，或者对被测软件提出的性能指标，测试软件的运行性能，例如：传输连接最长时限、传输错误率、计算精度、记录精度、响应时限和恢复时限等。  测试系统的能力最高实际限度，即检查软件在一些超负荷情况下，功能实现的情况。例如：要求软件进行某一行为的大量重复、输入大量的数据或大数值数据、对数据库进行大量复杂的查询等。  **5. 健壮性测试（可选）**  采用人工的干扰使应用软件、平台软件或者系统硬件出错，中断正常使用，检测系统的恢复能力，以及程序的内存、临界资源等在负载压力下的正确性。  **6. 测试结果分析**  对所测试的模块在功能、代码、性能、健壮性等方面进行的测试结果汇总。  测试结果汇总表（例）如下：   |  |  |  | | --- | --- | --- | | **测试模块** | **测试项目** | **测试结果** | | 菜品首页浏览功能 | 轮播图展示与切换 | 测试通过 | | 分类模块展示与跳转 | 测试通过 | | 菜品列表模块展示与跳转 | 测试通过 | | 购物车功能 | 购物车为空提示跳转到首页 | 测试通过 | | 将菜品加入购物车 | 测试通过 | | 在购物车编辑菜品 | 测试通过 | | 在购物车页面结算 | 测试通过 | | 菜品收藏功能 | 收藏菜品 | 测试通过 | | 在收藏列表浏览菜品 | 测试通过 | | 菜品分类功能 | 分类icon展示 | 测试通过 | | 分类对应菜品展示 | 测试通过 | | 个人中心功能 | 登录功能 | 测试通过 | | 订单管理、收货地址展示与跳转 | 测试通过 | | 意见反馈与客服 | 测试通过 | | 菜品详情浏览功能 | 图片展示与图片放大展示 | 测试通过 | | 菜品价格与名称展示 | 测试通过 | | 菜品详情介绍 | 测试通过 | | 分享菜品、加入收藏、加入购物车 | 测试通过 | | 搜索功能 | 精确搜索 | 测试通过 | | 模糊搜索 | 测试通过 | | 收货地址功能 | 新建地址 | 测试通过 | | 编辑地址 | 测试通过 | | 分享功能 | 跳转分享成功 | 测试通过 | | | |
| **七、实验结论：**  （阐述选择的项目包含哪些功能，采用了什么技术完成各个功能，最终的实现效果。） | | |
| **八、总结及心得体会：**  （对整个项目进行回顾，比如，遇到的问题及解决办法、成员之间的沟通、协调问题，团队推进的效率及原因分析，总结、心得体会等等。） | | |
| **九、对本实验过程及方法、手段的改进建议：**  （可以对老师的讲授内容、方法、过程的意见或建议，也可以对自己在做实验之后进行反思和改进。） | | |
| **报告评分：**  **指导教师签字：** | | |