计算机学院 软件工程 课程实验报告

|  |  |  |
| --- | --- | --- |
| 实验题目：实验内容八 | | 学号：202100130028 |
| 日期：2024/4/22 | 班级： 计科2班 | 姓名：薛然然 |
| Email：1745047373@qq.com | | |
| 实验目的：  1. 深入理解UML  2. 了解计算机学科中的逻辑  3. 学习对比软件体系结构设计GB和IEEE最新SAD (Software Architecture Document)的标准  4. 研究经典软件体系结构案例  3. 完成自己项目的SRS | | |
| 实验内容：  1. 阅读“The Unified Modeling Language Reference Manual”，进一步学习UML知识，理解如何应用UML对系统进行建模  ——UML 是统一建模语言的简称，它是一种由一整套图表组成的标准化建模语言。UML用于帮助系统开发人员阐明，展示，构建和记录软件系统的产出。UML代表了一系列在大型而复杂系统建模中被证明是成功的做法，是开发面向对象软件和软件开发过程中非常重要的一部分。UML主要使用图形符号来表示软件项目的设计，使用UML可以帮助项目团队沟通、探索潜在的设计和验证软件的架构设计。  在我们的SRS文档中，应用了UML图对系统进行建模。  2. 浏览“LOGIC IN COMPUTER SCIENCE--Modelling and Reasoning about Systems”，了解常用逻辑及其在计算机学科中的应用  ——命题逻辑和谓词逻辑是两种常用的逻辑方法，它们在计算机中有着不同的应用。  （1）命题逻辑：命题逻辑是逻辑系统中最基本的一种逻辑，它处理具有明确真或假值的命题。在计算机学科中，命题逻辑的应用主要体现在以下几个方面：  a.计算机电路设计：命题逻辑的规则被用于设计和验证计算机电路的正确性。通过逻辑门（如AND、OR、NOT）和逻辑表达式，可以描述电路的行为，并进行验证和优化。  b.计算机程序构造：命题逻辑也被用于描述计算机程序的结构和行为。通过逻辑表达式，可以表示程序中的条件语句、循环语句等，从而确保程序的正确性和可靠性。  c.程序正确性证明：命题逻辑提供了一种形式化的方法来证明程序的正确性。通过构造逻辑证明，可以验证程序是否符合其设计规格和需求。  （2）谓词逻辑：谓词逻辑是命题逻辑的扩展，它研究对象之间的逻辑关系。以下是其应用的几个方面：  a.数据库：在关系数据库中，谓词逻辑提供了一种形式化的方法来描述和查询数据。通过定义谓词（即关系）和查询语言（如SQL），可以描述数据之间的逻辑关系，并进行高效的查询和推理。  b.人工智能：在人工智能领域，谓词逻辑被广泛应用于知识表示和推理。通过将人类知识转化为计算机可以理解的形式（即谓词逻辑表示），计算机可以利用这些知识进行推理和决策。例如，在专家系统中，谓词逻辑被用于描述领域知识和推理规则，以实现自动化的决策支持。  c.此外，谓词逻辑还在形式语言学和软件开发等领域有着重要的应用。在形式语言学中，谓词逻辑被用于描述和推理关于语言结构和语言规则的语言。在软件开发中，谓词逻辑被用于形式化验证，以提高程序的可靠性和安全性。  3. 分工协作，参考国标“13 - 软件(结构)设计说明(SDD)”等资料，对比参考SAD最新标准IEEE-42010.pdf，针对自己的项目设计SAD初稿。  ——我们将SAD文档内容大致定义为以下几个方向：  第一部分 引言：本部分主要概述文档内容组织结构，使读者能够对文档内容进行整体了解，并快速找到自己感兴趣的内容。同时，也向读者提供了架构交流所采用的视图信息。  第二部分 架构背景：本部分主要介绍软件架构的背景，向读者提供系统概览，建立开发的相关上下文和目标。分析架构所考虑的约束和影响，并介绍了架构中所使用的主要设计方法，包括架构评估和验证等。  第三、四部分 视图及其之间的关系：视图描述架构元素及其之间的关系，表达视图的关注点、一种或多种结构。  第五部分 需求与架构之间的映射：描述系统功能和质量属性需求与架构之间的映射关系。  第六部分 附录：提供架构元素的索引，包括术语表、缩略语表。  4. 分工协作，学习、检索研究经典软件体系结构案例。  On-the-Criteria-To-Be-Used-in-Decomposing-Systems-into-Modules.pdf  http://www.cs.cmu.edu/~ModProb/index.html  ——体系结构风格的形成是多年探索研究和工程实践的结果。一个良好和通用的体系结构风格往往是工程技术领域成熟的标志。经过多年的发展，已经总结出许多成熟的软件体系结构风格，例如：  数据流风格：批处理和管道/过滤器。  调用/返回风格：主程序/子程序、层次结构和C/S。  面向对象风格。  独立部件风格：进程通信和事件驱动。  虚拟机风格：解释器和基于规则的系统。  数据共享风格：数据库系统和黑板系统。  5. 完成软件需求规格说明SRS  ——文档已提交至仓库。 | | |
| 结论分析与体会：  1.UML 的应用与意义  ——UML（统一建模语言）是一种标准化的建模语言，用于帮助系统开发人员阐明、展示、构建和记录软件系统。它通过图形符号表示软件项目的设计，有助于团队沟通、探索设计潜力和验证软件架构的设计。  2.逻辑在计算机科学中的应用  ——命题逻辑在计算机领域中用于电路设计、程序构造和程序正确性证明。  ——谓词逻辑扩展了命题逻辑，应用于数据库、人工智能等领域，用于描述对象之间的逻辑关系、知识表示和推理。  3.软件架构设计说明（SAD）初稿  ——SAD文档按照以下结构组织：引言、架构背景、各视图及其关系、需求与架构的映射，以及附录。这些部分帮助读者理解系统架构及其设计决策的背景和逻辑。  4.经典软件体系结构案例学习  ——软件体系结构的形成经历了多年的探索与工程实践，形成了多种成熟的体系结构风格，如数据流风格、调用/返回风格、面向对象风格等。这些风格代表了工程技术成熟的标志，适用于不同类型的软件系统设计与实现。 | | |