工作日志

|  |  |  |
| --- | --- | --- |
| 时间 | 工作内容 | 备注 |
| 2022年6月2日 | 实验内容：  1.通过实例，熟悉各种UML图的使用。参考教材P226，补充材料6-4皇家服务站的例子，学习分析其各种UML图的设计过程。参照上面的例子，回顾自己的项目曾设计过的UML图，分工协作，修改。  2.论述利斯科夫替换原则（里氏代换原则）、单一职责原则、开闭原则、德（迪）米特法则、依赖倒转原则、合成复用原则，结合自己的实践项目举例说明如何应用（保存到每个小组选定的协作开发平台上，以组为单位）  3.上网查询“阿里编程规范（如：阿里巴巴JAVA开发手册）；华为编程军规”等，对照自己的代码看有哪些不符合规范的地方，修改。  4.阅读下面软件测试相关资料（或查阅其它相关资料），了解软件测试的基本概念、主要技术、重要挑战等  组长陈文盼对工作进行了分配，郑晓旭负责总结UML图，我和刘诗婷同学负责说明各面向对象设计原则，赵一帆负责查看代码规范，陈文盼负责总结软件测试，并对实验报告作出总结。 | 实验内容 |
| 2022年6月3日 | 我们学习和讨论了关于UML和面向对象设计原则、代码规范、软件测试等内容。 | 学习讨论 |
| 2022年6月5日 | 总结了部分面向对象设计原则：  **L利斯科夫替换原则（里氏代换原则）**  在面向对象的程序设计中，里氏替换原则（Liskov Substitution principle）是对子类型的特别定义。  定义1：如果对每一个类型为 T1的对象 o1，都有类型为 T2 的对象o2，使得以 T1定义的所有程序 P 在所有的对象 o1 都代换成 o2 时，程序 P 的行为没有发生变化，那么类型 T2 是类型 T1 的子类型。  定义2：所有引用基类的地方必须能透明地使用其子类的对象。  里氏代换原则(Liskov Substitution Principle, LSP)：所有引用基类（父类）的地方必须能透明地使用其子类的对象。  **S单一职责原则**  就一个类而言，应该仅有一个引起它变化的原因。应该只有一个职责。每一个职责都是变化的一个轴线，如果一个类有一个以上的职责，这些职责就耦合在了一起。这会导致脆弱的设计。当一个职责发生变化时，可能会影响其它的职责。另外，多个职责耦合在一起，会影响复用性。例如：要实现逻辑和界面的分离。  **德（迪）米特法则** | 面向对象设计原则 |
| 2022年6月7日 | 作为本周工作的总结，我们建立了项目跟踪表，记录了每个人工作量、里程碑、工作进度，以便于记录各人的工作完成情况，之后的每周都将依据实际情况对其进行更新。我也完成了自己的工作日志。 | 工作总结 |