|  |  |
| --- | --- |
| 得分 |  |

5、（本题满分10分）

为下列描述建立ERD

我们企业的MIS 部门想构造一个数据库来跟踪所有的硬件和软件。我们拥有工作站、网络服务器和外设，而且MIS 部门想跟踪软件包以及这些软件包的许可证。有些软件许可证是针对单机的，我们可以把这个软件安装在网络服务器上，但只能允许与许可证授权的用户数同样多的网络用户使用该软件。我们还拥有网络许可证，单个网络许可证授权了一定数量的用户。非网络许可证可以安装在工作站或服务器上。我们想跟踪软件许可证安装在哪里。某些许可证可以在某个时间未被安装在任何地方。我们还必须能够证明安装软件的合法性。每个许可证必须被跟踪到一个购买订单、赠品或者一次租借。我们也可以订购一些软件。我们订购软件包，同时收到许可证。

|  |
| --- |
| 合  法  性 |

|  |
| --- |
| \* |

|  |
| --- |
| N |

|  |
| --- |
| 硬件 |

|  |
| --- |
| 用户 |

|  |
| --- |
| 网络服务器 |

|  |
| --- |
| 工作站 |

|  |
| --- |
| 软件 |

|  |
| --- |
| 许可证 |

|  |
| --- |
| 来源 |

|  |
| --- |
| 网络许可证 |

|  |
| --- |
| 单机许可证 |

|  |
| --- |
| 其他设备 |

|  |
| --- |
| 授权使用 |

|  |
| --- |
| 1 |

|  |
| --- |
| N |

|  |
| --- |
| N |

|  |
| --- |
| \* |

|  |
| --- |
| 购买 |

|  |
| --- |
| 租借 |

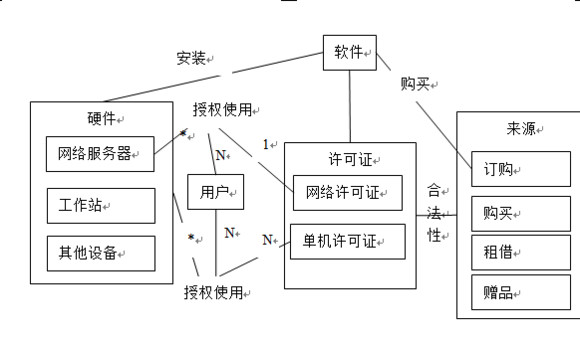
|  |
| --- |
| 赠品 |

|  |
| --- |
| 订购 |

|  |
| --- |
| 安装 |

|  |
| --- |
| 购买 |

|  |
| --- |
| 授权使用 |



|  |  |
| --- | --- |
| 得分 |  |

7、（本题满分20分）

下面是一段用例的描述，针对一个汽车保险系统中“将一辆新车加入一个已有保单中”的用例。请你为其设计：（1）领域模型（要求给出建立过程）；（2）活动图。

1) 客户打电话给保险公司，并提供他的保单号，办事员输入这个信息，系统显示基本的保单。然后办事员检查信息，以确保保险费通用及保单有效。

2) 客户给出要添加的汽车的牌子、模型、年份和车辆识别代号（VIN），办事员输入这些信息系统验证这些数据是否有效。然后客户选择期望的保额类型，以及每种类型的数量，办事员输入这些信息，系统会逐一记录并根据保单限制验证所请求的数量。输入所有的保额后，系统验证保额总和，包括保单上的其他汽车。

3) 最后，客户必须要确认所有的驾驶员，以及他们驾驶汽车的时间比例。如果有一个新驾驶员加入，则调用另一个用例“增加新驾驶员”。

4) 整个过程最后，系统更新保单，计算新的保险费，打印新的保单说明，邮寄给保单所有人。

（本页为答题页）

（1）领域模型：结果基本正确（6分）；给出了过程（4分）

（2）活动图：结果基本正确（10分）

|  |  |
| --- | --- |
| 得分 |  |

8、（本题满分5分）

用于需求获取的原型与用于需求验证的原型有何异同？

都是原型的一种用法（1分）

用于获取的原型起始于不明确的需求（1分）；侧重于可调整方案（1分）；

用于验证的原型起始于比较明确的需求（1分）；侧重于功能评价方案（1分）；

|  |  |
| --- | --- |
| 得分 |  |

9、（本题满分10分）

如何有效的处理需求的变化是很多现代软件开发技术的主题，对此现象你有什么看法？

(1) 认识到需求变化是不可避免的；1分

a) 给出需求变化不可避免原因的解释；2分

包括：问题发生了改变。环境发生了改变。需求基线存在缺陷。

(2) 明确需求的变化是需要控制的；1分

a) 能够描述需求基线的概念；1分

在响应需求变更的过程中，项目团队还要及时准确的维护需求基线，审计变更记录：要更新需求基线，保证项目涉众可以访问到最新的需求；

b) 能够描述需求变更的过程；2分

(3) 明确需求变化需要特殊处理的；1分

a) 能够大概写出特殊处理情况的；2分

包括：

l 推迟产品的交付时间。

l 要求增派人手。当然，这个做法只有在有限的情况下有效，因为很多情况下，增加人手只会使得项目更加落后。

l 要求员工加班工作。一段时期的加班会耗尽员工的储备精力，因此加班不能是长期的，一般以30天为限，否则会产生很多消极影响。因此，这个做法也只能适度的使用。

l 推迟或者去除尚未实现的优先级较低的需求。

容许产品质量的降低。当然，这个做法是最不提倡的，因为低质量的产品会伤害整个开发团队。所以，除非其他的做法都不能达到效果，否则不要使用这种做