系统应有的质量属性

1. **互操作性**

场景：系统分散、数据交互困难、数据库类型复杂、用户重复填写个人信息

源：一个系统请求与另一个系统交换信息

刺激：交换信息的请求

制品：系统交换信息的系统

环境：希望互操作的系统在运行时发现或在运行时之前已知

响应：交换请求成功；交换请求被拒绝

响应度量：正确处理的信息交换百分比

正确拒绝信息交换的百分比

策略：

1、统一的数据处理接口

2、发现服务

3、编排

4、裁剪接口

5、电子函件和消息传递系统

1. **安全性**

源：黑客

刺激：试图显示数据、试图修改系统可用性

制品：系统中的数据

环境：连接有防火墙

响应：进行身份验证进行，阻止对数据的访问

响应度量：短时间内审核其身份，拒绝其访问，并限制系统可用性

策略：

抵抗攻击：①对用户身份验证

②对用户进行授权

③维护数据的机密性

④维护完整性

⑤限制暴露的信息

⑥限制访问

检测攻击

从攻击中恢复。

1. **易用性**

场景：用户修改个人信息

刺激源：用户

刺激：用户修改个人信息

制品：信息登记系统

环境：系统正常运行，用户修改个人信息，获取服务

响应：用户修改信息成功，系统更新资料，恢复到以前页面供用户选择后续服务。

响应度量：用户修改信息成功响应速度，系统数据更新的正确率

策略：

运行时战术：一旦系统执行，就可以通过为用户提供关于系统正在做什么的反馈，以及用于提供发出基于易用性命令的能力来增强易用性。例如，在纠错或更高效的操作中，“取消”、“撤销”、“聚合”和“显示多个视图”均为用户提供支持。

1. **可修改性分析**

场景：政府要增加新的线上事务办理功能（如签证预约）

刺激源：系统开发人员

刺激：系统要增加新的功能页面

制品：新的功能办理页面

环境：系统正常运行

响应：针对页面查找构架中需要修改的位置，进行修改添加并且不影响其他功能，对修改进行测试，部署所做修改

响应度量：针对页面查找构架中需要修改的位置，进行修改添加并且不影响其他功能，对修改进 行测试，部署所做修改

策略：

维持语义的一致性、预期期望的变更、泛化该模块、限制可能的选择。

1. **可测试性**

软件的可测试性是指软件发现故障并隔离、定位其故障的能力特性，以及在一定的时间和成本前提下，进行测试设计、测试执行的能力。

场景：单元测试人员测试信息查询模块

刺激源：单元测试人员

刺激：测试人员输入信息关键词，进行信息查询

制品：搜索模块的代码

环境：在开发时进行

响应：通过商品关键词查询，所检索出的商品信息呈列表显示

响应度量：在较短的时间内完成对信息的检索

策略：

1、输入/输出

首先记录回放。再将接口与实现分离。最后特化访问路线/接口。

2、内部监视

1. **可用性**

场景：某一时期用户猛增

刺激源：用户

刺激：登录人数过多，导致系统无法响应

制品：系统的处理器、通信通道、存储器、进程

环境：用户的正常浏览操作

响应：页面呈现“网络出现故障，重新刷新”等的提示信息，提示用户下一步操作

响应度量：系统降级模式下继续运行，用户刷新页面或者重新登录之后可继续正常使用。

策略：

错误检测：命令/响应、心跳和异常

错误恢复：准备恢复、修复系统。

错误预防：从服务中删除、事务、进程监视器。

1. **性能**

场景：政府负责人希望，平台能够自动生成相关数据报表与数据分析结果供其查阅，最好能给出一些智能化决策建议。

刺激源：政府工作人员

刺激：查阅数据

制品：系统生成数据表与数据分析结果

环境：系统正常运行

响应：系统生成数据表与数据分析结果，并给出智能化决策建议

响应度量：数据报表与数据分析结果生成的速度，智能化决策的可靠性。

策略：

采用适合的调度策略处理事务，提升生成报表的速度。

通过机器学习生成智能化决策。