

需求规格评审表-2020HYSE08

需求特性	描述	检查内容
清晰性	需求描述是否明确清晰，能否防止产生歧义	<ul style="list-style-type: none"> - 需求描述是否具有明确的目标和定义 - 是否存在任何可能导致混淆或误解的术语或表达方式 - 是否存在任何歧义或模糊之处
完整性	需求是否充分覆盖了所有关键方面	<ul style="list-style-type: none"> - 需求是否覆盖了所有主要的功能和业务需求 - 是否存在任何关键方面被遗漏的情况 - 需求和文档是否具有足够的详细信息，以便开发人员理解和实现
可靠性	需求是否具有足够的可靠性和健壮性	<ul style="list-style-type: none"> - 需求是否能够正确处理各种异常情况 - 需求是否能够恢复到正常状态，以便在故障发生时继续工作或给出提示 - 需求是否具有足够的容错能力
可用性	需求是否易于使用和理解	<ul style="list-style-type: none"> - 需求是否符合用户的期望和需求 - 需求是否具有足够的易用性和可操作性 - 需求和文档是否易于学习和使用
安全性	需求是否具有足够的安全性	<ul style="list-style-type: none"> - 需求是否能够保护系统的安全性和完整性 - 需求是否能够正确地验证和授权用户的访问权限 - 是否存在任何可能导致安全漏洞的情况，例如防范常见的安全攻击如注入、攻击等。
兼容性	需求是否与系统的其他部分或其他系统兼容	<ul style="list-style-type: none"> - 需求是否符合系统的整体架构和设计 - 需求是否与现有的硬件、软件和第三方工具兼容 - 是否存在任何可能导致兼容性问题的情况
可维护性	需求是否易于维护、跟踪和修改	<ul style="list-style-type: none"> - 需求是否具有足够的可扩展性和可维护性 - 需求是否易于修改和调整 - 需求是否具有足够的文档和注释，以便开发人员理解和维护
可测试性	需求是否能够被有效地测试和验证	<ul style="list-style-type: none"> - 需求是否具有可衡量和可验证的标准 - 是否存在任何无法测试或无法验证的需求 - 需求是否能够被自动化测试
性能	需求实现时响应时间和资源利用率	<ul style="list-style-type: none"> - 对需求的响应时间，例如操作响应时间快，用户等待时间短。 - 需求实现时的资源利用率，例如代码优化，资源占用率低。 - 实现时的并发性能，例如在高并发情况下系统稳定性和性能都能够保证。
一致性	需求描述中在不同的界面中行为一致	<ul style="list-style-type: none"> - 功能之间的一致性，例如相同的操作在不同的场景下有相同的结果。 - 界面之间的一致性，例如相同的元素在不同的界面中位置、样式、行为一致。 - 文档之间的一致性，例如不同的文档中用词、术语、格式一致。

说明：需求特性分类共 10 组，每组满分 10 分。