

GLMS
国联民生证券股份有限公司企业标准

Q/GLMS J001-2026

信息系统测试标准

Information System Testing Standards

2026-06-11发布

2026-06-11实施

国联民生证券股份有限公司 发布

目 次

目 次	1
前 言	3
国联民生证券信息系统测试标准	4
1 范围	4
2 规范性引用文件	4
3 术语和定义	4
3.1 重要信息系统	5
3.2 自主开发类系统	5
3.3 外购类系统	5
3.4 代维类系统	5
3.5 技术测试	6
3.6 业务验收测试	6
3.7 IT 项目经理	6
3.8 测试类型	6
3.9 需求提出方	6
3.10 DevOps 平台	6
4 测试架构和职责	7
4.1 技术测试	7
4.1.1 按测试级别	7
4.1.2 按测试类型	7
4.2 业务验收测试	8
4.3 角色职责	8
5 测试准入标准	9
5.1 自研类系统	9
5.2 外购类系统	9
6 测试过程管理	10
6.1 自研类系统	10
6.1.1 测试设计	10
6.1.2 测试执行	12
6.1.3 缺陷管理	13
6.2 外购类系统	15
6.2.1 测试设计	15
6.2.2 测试执行	17
6.2.3 测试用例规范	17
6.2.4 缺陷管理	17
6.2.5 缺陷提交规范	17
7 测试准出标准	20

附 录 A 测试类型	23
A.1 功能测试	23
A.2 性能测试	24
A.3 可靠性测试	25
A.4 安全测试	26
A.4.1 目的	26
A.4.2 测试内容/关注点	26
A.4.3 测试技术	27
A.5 可移植性测试	28
A.5.1 目的	28
A.5.2 测试内容/关注点	28
A.5.3 测试技术	29
A.6 可维护性测试	30
A.7 应急演练	31
A.8 联网测试	31
A.9 全市场测试	31
A.10 选型测试	32
附 录 B 软件测试模板	34
B.1 测试工作量评估表	34
B.2 测试计划	35
B.3 测试案例模板	35
B.4 测试缺陷列表	37
B.5 测试报告模板	38
附 录 C 测试类型选择原则	40

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规范》的规定起草。

本文件由国联民生证券股份有限公司提出。

本文件由国联民生证券股份有限公司归口。

本文件起草部门：国联民生证券股份有限公司信息技术总部。

本文件主要起草人：徐璨、周强强。

国联民生证券信息系统测试标准

1 范围

本文件规定了信息系统测试管理工作职责和工作流程，测试准入/准出标准，测试类型选择的原则，以及对项目所需测试资源的评估标准，避免测试不足或过度测试。

本文件适用于公司新建、升级、变更、换代信息系统中包含技术测试或业务验收测试的项目，涉及人员包括项目经理、测试人员、开发人员、安全管理员、运维人员、系统管理员等。本文件对于重要信息系统以及其他直接对客提供交易、服务类系统，必须严格遵守，其他系统建议参照执行。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

JR/T 0175-2019 《证券期货业软件测试规范》

GB/T 25000 《系统与软件工程 系统与软件质量要求和评价 (SQuaRE)》

《证券期货业信息系统压力测试指南》

《国联证券股份有限公司信息技术项目管理办法》

3 术语和定义

下列术语和定义适用于本文件。

3.1 重要信息系统

重要信息系统是指支持公司关键业务功能，如出现异常将对证券期货市场和投资者产生重大影响的信息系统。包括集中交易系统、投资交易系统、金融产品销售系统、估值核算系统、投资监督系统、份额登记系统、第三方存管系统、融资融券业务系统、网上交易系统、电话委托系统、移动终端交易系统、法人清算系统、具备开户交易或者客户资料修改功能的门户网站、承载投资咨询业务的系统、存放承销保荐业务工作底稿相关数据的系统、专业即时通信软件以及与上述信息系统具备类似功能的信息系统。

3.2 自主开发类系统

自主开发类系统是指完全由公司信息技术总部自有人力完成系统建设，或者以信息技术总部自有人力为主，承担设计、研发等主要关键工作，部分开发、测试则以外包作为补充的系统（以下简称为“自研类”）。

3.3 外购类系统

外购类系统是指公司信息技术总部与专业厂商就某个领域进行合作开发，使用原厂的开发人员，或者基于某个外购产品进行定制化（定制化模块采购），最终满足公司要求的系统。外购类系统（标准化模块采购），一般为行业标准产品的采购（以下简称为“外购类”）。

3.4 代维类系统

代维类系统是指归属于业务使用部门或子公司管理，由其承担系统建设和主要运维职责（包括但不限于测试、监控、变更、应急等），并且部分基础运维职能委托信息技术总部代为管理的系统。

3.5 技术测试

技术测试是指运用测试技术，对信息系统的功能、性能、安全等方面进行检验的测试活动，通常包括单元测试、集成测试、系统测试、系统集成测试四个测试级别。

3.6 业务验收测试

项目组技术测试完成后，需求单位组织业务人员进行业务测试与验收，确认项目成果符合项目需求，并形成业务验收测试报告，作为系统上线依据。

3.7 IT 项目经理

IT项目经理在信息系统建设前，由团队自行指定，整体负责项目建设。

3.8 测试类型

测试类型是指使用不同的测试技术，对功能、性能、安全、兼容、可靠性、可维护性、可移植性等对被测软件进行测试验证的类型。

3.9 需求提出方

需求提出方包括外部需求提出方和内部需求提出方两类。外部需求提出方主要包括交易所、登记结算机构、银行、基金公司、合作公司等；内部需求提出方是公司相关业务和管理部门。

3.10 DevOps 平台

Devops平台是公司面向研发、测试、运维、运营团队的一站式DevOps平台，构建包含敏捷迭代、持续集成、制品管理、测试管理、度量分析等模块的管理工具，本文主要针对测试管理模块进行描述。

4 测试架构和职责

信息技术总部负责系统的技术测试，需求提出方（需求单位及相关业务和管理部门）负责系统的业务验收测试。

IT项目经理（以下简称“项目经理”）组织项目的相关方确定实施系统的技术测试和业务验收测试，确定技术测试经理和业务测试经理，测试经理负责规划测试工作，编制测试计划和方案，组织实施测试工作。

4.1 技术测试

从测试级别和测试类型的角度，描述在相关测试活动中的责任主体和参与者，如表1、表2所示。

4.1.1 按测试级别

系统类别	测试级别	责任主体
自研类重要信息系统	单元测试	开发团队
	集成测试	
	系统测试	测试团队
	系统集成测试	
	验收测试	
外购类重要信息系统	验收测试	
自研类一般信息系统	全部	项目团队自测。
外购类一般信息系统		测试团队提供技术支持。
代维类系统		

表 1 测试级别及责任主体

4.1.2 按测试类型

测试类型	责任主体	参与者
功能测试	需求提出方、开发团队、测试团队	需求提出方、开发团队、测试团队、运维团队及供应商等干系人。
性能测试		
可靠性测试		
可移植性测试		
可维护性测试		
安全性测试	需求提出方、安全团队	需求提出方、开发团队、测试团队、运维团队、安全团队、第三方安全服务机构及供应商等干系人。
应急演练	需求提出方、运维团队	测试发起方或组织方应根据实际的业务需求和测试需求选择必要的关联方参与测试。关联方主要包括但不限于市场核心机构、市场经营机构、市场服务机构等市场各参与方。
联网测试		
全市场测试		
选型测试	需求提出方、运维团队	需求提出方、开发团队、测试团队、运维团队及供应商等干系人。

表 2 测试类型及参与者

4.2 业务验收测试

由项目经理组织和协调相关责任主体，进行业务验收测试，按需求类型描述责任主体和反馈方式，如表3所示。

需求类型	责任主体	反馈方式
内部需求	需求提出方	产品经理提交业务验收测试报告。
所司改造	业务人员或业务管理部门	业务人员提交业务验收测试报告。

表 3 需求类型及反馈方式

4.3 角色职责

参与测试的团队中，各角色的职责如表4所示。

团队	角色	职责
需求提出方	客户	提出业务需求，对系统需求规格说明书进行确认，对出现分歧的业务需求进行确认，负责业务验收测试。
开发团队	项目经理	组织项目的相关方确定实施系统的路径，制定项目整体日程安排，协助测试环境准备。
	产品经理	对业务需求进行分析，并完成可测的系统需求规格说明书。该角色职责亦可由项目经理、需求经理承担。

	开发人员	根据业务需求文件及系统设计文档进行编码，对代码引起的缺陷进行分析及修复等。
测试团队	测试经理	组建和管理测试团队，把控项目测试整体质量、进度、资源/成本、风险，协调过程中各种问题等。
	测试人员	参与业务需求调研，对系统需求规格说明书进行静态测试；测试需求的编写和维护；测试用例编写、执行、记录测试结果、对发现的缺陷进行提交、验证等；负责测试环境部署与维护；参与测试框架设计，分析测试工具需求，完成测试工具开发和维护。
安全团队	安全管理员	通过安全测试手段和方法，验证软件的安全特性实现与预期一致、检验软件产品对各种攻击情况的防范能力，从而保障产品的安全质量。
运维团队	运维工程师	协调和安排整个项目的运维工作； 负责项目非功能需求的提出和验收等； 负责项目上线计划编写及与相关方沟通上线相关工作。

表 4 各角色的职责

5 测试准入标准

5.1 自研类系统

对于自研类系统，采用敏捷测试流程，测试团队应在产品经理组织需求分析时介入，在项目经理完成项目计划的制定后，即满足技术测试准入条件，测试团队开展的技术测试工作和开发团队开展的开发工作同步进行，持续进行问题反馈。

5.2 外购类系统

外购类的重要信息系统，测试团队执行测试之前，应达到准入要求后，才可展开测试执行。测试准入要求如下：

1. 测试设计安排审核通过。

2. 验收测试用例评审通过，验收测试用例可以从厂商提供的系统测试用例中抽取部分用例，也可以基于验收测试目的，重新编写。

3. 冒烟测试通过，无阻碍进一步测试的缺陷存在。

4. 测试环境准备完成。

5. 验收测试前，厂商应提供完善的系统文档，由验收测试小组负责对厂商提供的系统文档进行审核验收。参考下表5：

文档	是否必须	说明
改造说明书	是	对已上线系统的改造，须提供改造说明书，用于概要说明改造范围、影响范围、涉及模块等事项。
需求规格说明书	否	若定制化需求，须厂商提供，通过该说明书可评估厂商需求理解是否有偏差或遗漏。
测试用例	是	须提供完善的系统测试用例
测试报告	是	由厂商提供产品出厂测试报告，视项目要求，测试报告应包括功能测试报告、性能测试报告等。
系统安全设计实施方案	是	针对新上线系统或产生重大架构变更的系统，须由厂商提供包括但不限于系统安全设计实施方案等安全性证明材料。
安全测试报告	是	针对新上线系统或产生重大架构变更的系统，由厂商提供安全测试报告。
产品说明书	是	由厂商提供产品说明书。
安装手册	否	若首次安装，须提供。若是新增功能模块，不涉及安装，可不提供。
升级配置说明	是	必须提供。
运维手册	是	新增软件系统或已上线系统，新增运维操作、硬件设备的，须由厂商提供说明手册。

表 5 测试准入文档提交情况

6 测试过程管理

使用DevOps平台进行信息系统测试过程管理，全部技术测试活动均应在平台中记录、流转。

6.1 自研类系统

自研类系统测试过程管理包括测试设计流程、测试执行流程和缺陷管理流程。

6.1.1 测试设计

测试计划流程，是在需求文档输出后，测试即开始介入，参与需

求评审、研发程序设计，最终完成测试用例评审的整个过程中，持续进行的测试计划和用例设计的相关活动，如图1所示。

测试计划流程，分为测试计划、测试设计阶段、测试评审阶段，流程说明如下：

一、测试计划、测试设计阶段

1. 测试经理根据需求文档和程序设计文档，撰写测试计划初稿（文档标准规范见附录B.2）。

2. 测试人员根据测试计划初稿，编写思维导入和测试用例（文档标准规范见附录B.2）。

3. 测试设计过程中对需求和程序设计文档产生分歧时，由测试经理发起分歧沟通。

4. 若需要对需求/程序设计文档进行修改，则修改后更新测试计划，并根据最新需求/程序设计文档进行测试设计。

5. 输出测试思维导图、测试用例待评审稿。

二、测试评审阶段

1. 测试经理组织对测试思维导图、测试用例待评审稿评审会议。

2. 输出测试思维导图、测试用例已评审稿。

3. 测试经理开始准备测试环境和测试数据，内容包括基线版本、上下游依赖环境、脱敏后的测试数据。

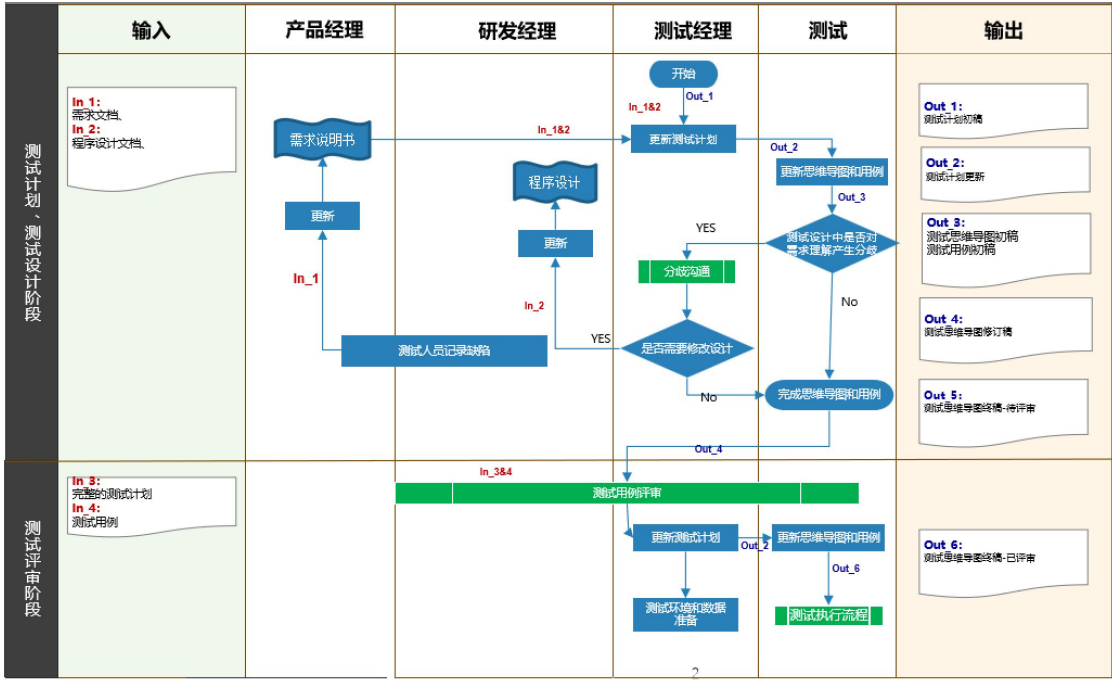


图 1 自研类系统测试设计流程管理

6.1.2 测试执行

测试执行流程，是指从开发正式提测开始，到业务验收完成的整个过程中相关活动，如图2所示。

测试执行流程分为SIT测试阶段（也称系统集成测试阶段）、UAT测试阶段（也称用户验收测试阶段）：

一、SIT测试阶段

- 1.测试人员收到正式提测的制品后，首先应该进行冒烟测试，验证制品的可测性。
- 2.若冒烟测试通过，则进入正式测试阶段；若冒烟测试不通过，需要打回，由开发人员修改后重新提测。
- 3.正式测试执行后，若用例执行失败，提交缺陷（参见6.2.5缺陷提交规范）。

4.测试执行完成后，更新测试计划进度。

二、UAT测试阶段

- 1.UAT测试阶段开始前，不应存在非关闭状态的缺陷，若存在，测试经理应依照缺陷管理流程，对非关闭状态缺陷进行处理。
- 2.研发经理发布经过版本复核的正式版本，测试人员进行UAT环境部署以及升级操作。
- 3.执行UAT测试，对当前版本变更内容进行第二轮测试验证。
- 4.执行回归测试，对当前版本进行回归测试。
- 5.测试完成后，输出测试报告（参见附录B.5测试报告模板）。
- 6.测试经理发起测试报告评审，并协助需求提出方进行业务验收测试。

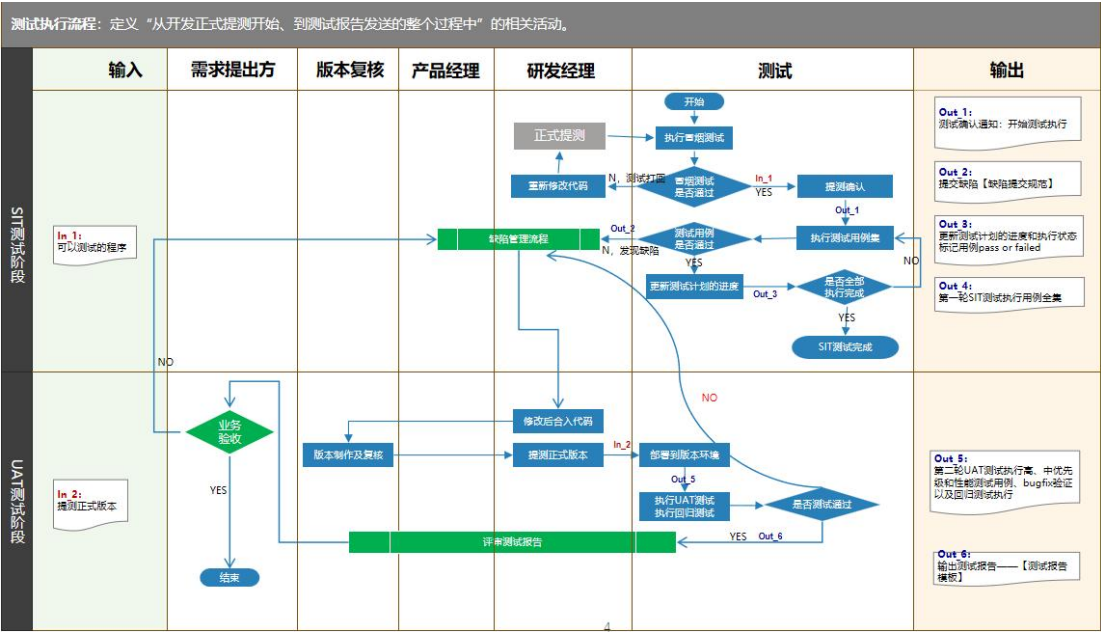
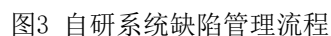


图 2 自研类系统测试执行流程管理

6.1.3 缺陷管理

- 1.测试人员创建并提交缺陷给研发人员，缺陷状态为新。



6.2 外购类系统

外购类系统的技术测试，以验收测试为主，在测试执行之前，测试经理应根据项目相关资料，组织测试需求分析评审，明确测试计划、测试范围、资源投入，组织厂商进行系统变更说明介绍，明确变更范围，如图4所示。

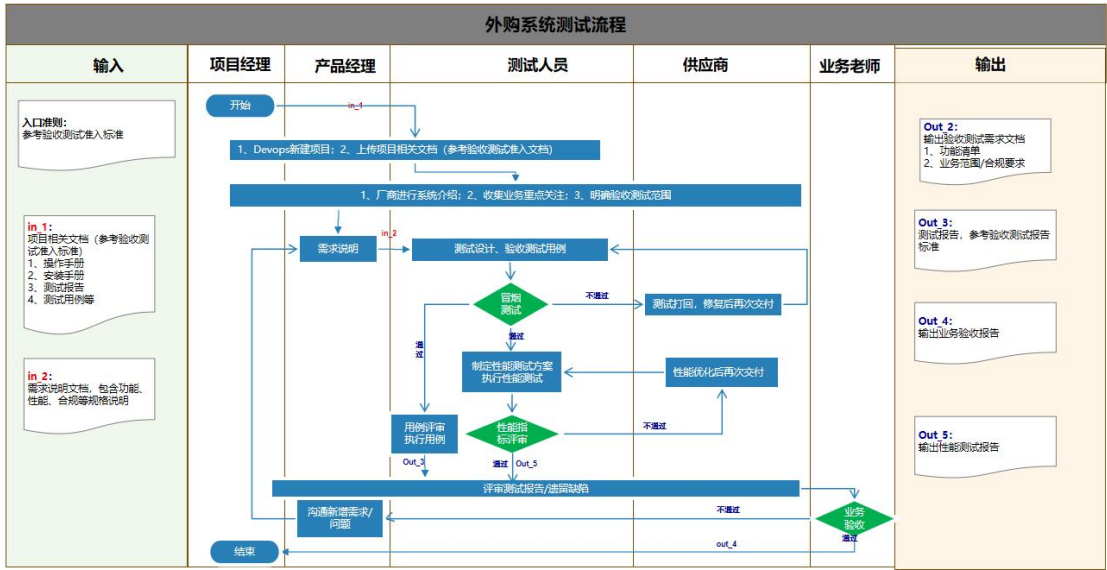


图4 外购系统测试流程

6.2.1 测试设计

测试设计应合理安排测试范围、资源投入、里程碑时间点，避免测试不足或过度测试(见图5)。

通常需要从以下方面考虑，测试类型选择原则，详细选择原则可参考附录C内容。

1.新系统上线及重大版本上线测试，应对系统所有业务功能进行全面测试，同时进行性能、可靠性等非功能测试。

2.补丁版本上线测试，应对修改和新增业务功能重点测试，同时对关联变化功能以及系统核心功能进行回归测试。

3. 运行在不同操作系统或硬件设备上的系统,应考虑可移植性测试,重点关注系统兼容性。

4. 有大量终端客户使用的系统,应考虑易用性测试、安全性测试,重点关注用户体验性以及系统应对各种攻击情况的防御能力。

5. 实际上线功能有超出需求提出方原始需求的,应对超出部分关联变化功能以及系统核心功能进行回归测试。

测试思维导图是对测试设计文档的图形化体现,是帮助测试人员梳理测试场景、测试环境依赖、测试资源、测试计划等避免遗漏的有效工具,也是测试评审时直观、高效的展示形式,便于参与评审人员评审测试计划,参考如下:



图5 测试设计思维导图

6.2.2 测试执行

测试用例和测试执行，应在DevOps平台中记录，标记测试轮次、测试用例执行结果，未执行用例应写明原因。对于核心系统及核心功能应对测试执行过程以截图或保留日志的方式留痕。

6.2.3 测试用例规范

- 1.必填字段：测试用例创建时，必填字段包括标题、分组、等级。
- 2.测试用例等级规范：分为P0、P1、P2、P3，其中P0代表核心主流程测试用例，P1代表高频核心功能测试用例，P2代表异常场景或低频测试用例，P3代表边缘功能或用户体验测试用例。
- 3.用例标题规范：需求名称-测试用例名称，测试用例名称应清晰简述测试内容，如被测模块名称、业务场景。
- 4.前置步骤：描述测试用例执行的前置条件，包括测试数据、前置操作步骤、测试环境条件等。
- 5.测试步骤：描述测试执行时所需的操作步骤、测试输入。
- 6.预期结果：描述测试步骤执行后的被测系统预期结果，可以与测试步骤一一对应，也可描述最终结果，不可出现系统正常、测试正常等笼统描述，需要描述被测系统具体的预期反馈结果。

6.2.4 缺陷管理

测试执行过程中，发现的缺陷，应记录到DevOps平台中，定期更新缺陷状态，测试结束时，应无新建\未确认的缺陷。

6.2.5 缺陷提交规范

缺陷标题。不能含有测试流程步骤和模块数据，不能有错别字，描述节点和失败结果时需要与系统中描述一致。同时不可有主观语气，整体以描述事实为基准。不可模糊描述，标题总字数控制在20字内。

优先级。紧急：阻塞核心功能或引发系统崩溃、数据丢失，修复优先级最高。高：影响主要流程或者用户体验（比如高频闪退），需要高优先级处理。中：次要功能异常或边界场景问题，可延期修复。低：界面优化或非核心瑕疵，不影响发布。

类型。缺陷的一级分类，下拉列表为 Bug-缺陷、PROD Bug-生产事故、ENHANCE-优化

缺陷类型。对缺陷的二级分类，字段类型为单选下拉列表，非必填；下拉列表选项如下：**【代码错误】【设计缺陷】【界面优化】【安装部署】【配置相关】【性能问题】【安全相关】【新增需求】【设计变更】【兼容性】【易用性】【其他】**。

解决方案。描述缺陷处理的最终结果，字段类型为文本，非必填；下拉列表选项如下：

【设计如此】若缺陷所述内容与产品或设计图一致，则研发人员在将缺陷置为已解决状态时，可选择：设计如此，但建议在备注内进行说明；

【重复缺陷】若缺陷为重复缺陷，即已经存在与此相同的缺陷，则研发人员在将缺陷置为已解决状态时，可选择：重复缺陷，并填写重复缺陷的ID，若有特殊说明可在备注内进行说明；

【外部原因】若缺陷的出现原因为外部原因（例如硬件、第三方软件等导致的问题），则研发人员在将缺陷置为已解决时，可选择：外部原因，并在备注内进行说明；

【已解决】若缺陷中描述的问题已解决，则研发人员在将缺陷置为已解决状态时选择：已解决；

【无法重现】若缺陷为无法重现的缺陷，则研发人员在将缺陷置为已解决状态时，可选择：无法重现，并在备注内进行说明，建议研发人员遇到此类问题联系测试人员进行复现；

【延期处理】若研发人员考虑到时间等原因，觉得缺陷需要延期进行处理，则在将缺陷置为已解决时，选择：延期处理，并填写计划在哪个版本进行修复，在备注内进行原因说明；

根本原因。由开发人员填写，描述缺陷原因，字段类型为文本，非必填。

根因归类。字段类型为单选下拉列表，非必填；下拉列表选项如下：【需求不明确或需求变更】【变更未告知】【测试用例覆盖不全】【测试执行】【测试环境或数据受限】【特殊场景或数据】【其他】。

改进方案。由开发人员填写，用于描述缺陷的解决方案，字段类型为文本，非必填；在页面模板，缺陷模板里加入上面添加的五个字段；生产问题，在缺陷页面添加自定义标签“生产问题”。

缺陷内容。描述缺陷发生环境，前置条件、步骤、结果和期望结果。

【环境】：包括操作系统、被测系统版本、被测环境（UAT\SIT\PROD）等其他重要信息；

前置条件：前置条件需要明确地指出提交的缺陷是在何种情况下发生的。如4G网络正常；已进入XX页面；

【重现步骤】：重现步骤需要清晰地分步来描述缺陷问题，每步要用数字序号。对于特定数据产生的问题，要提供具体的测试数据。步骤之间要用组合连接符->，步骤描述中的模块不用带引号。同时步骤描述不能过长，语言精炼。步骤中不能含有结果。

【测试结果】：即实际结果，实际结果要精确描述按当前步骤产生的结果，不能有歧义，不要有模糊不确定的描述，如：有误、不正常、不准确等。直接描述结果值。另外，预期结果要与实际结果相对应。

【期望结果】：按照复现测试步骤及需求文档的要求准确描述预期结果。同时预期结果必须是无歧义的，可以判定的。描述结果时不要含有步骤。

优先级&严重程度。根据当前缺陷由测试人员来界定，也需要参考当前的环境。严重程度级别与优先级别不一定总是成正比，需要结合实际情况来全面考虑。

7 测试准出标准

测试准出标准包含三个方面：

1. 用例执行情况：计划实施的测试用例 100% 执行完成，无法执行的用例备注原因，且项目组评审通过。

2. 缺陷解决情况：所有缺陷均被记录，遗留缺陷验收小组评审通过，且流转至下一迭代，预发布迭代中缺陷均已终结。

3. 满足各测试类型准出标准，参考表 6 所示。

测试类型	准出要求
功能测试	1. 所有功能点均通过测试验证，与需求文档严格一致； 2. 设计的测试用例 100% 执行完成，未执行的用例已标注未执行原因； 3. 优先级为“紧急”或“高”的缺陷完成 100% 修复。优先级为“中”或“低”的遗留缺陷需有明确解决方案与排期，且经风险评估确认不影响上线。
性能测试	1. 在预设负载下，响应时间、吞吐量、并发用户数、资源利用率（CPU/内存/磁盘 I/O/网络带宽）等关键指标必须满足需求文档或 SLA（服务等级协议）中的明确要求。 2. 系统在持续负载（如 8 小时以上压力测试）中无崩溃、内存泄漏或性能衰减现象。 3. 所有优先级为“紧急”和“高”的性能缺陷（如超时错误、资源耗尽）必须修复并通过回归验证。优先级为“中”和“低”的遗留缺陷需有明确解决方案与排期，且经风险评估确认不影响上线。 4. 已出具包含测试目标、场景设计、结果数据、瓶颈分析及优化建议的正式性能测试报告。 5. 测试计划、脚本、监控日志等原始数据完整存档备查。
可靠性测试	1. 所有功能点均通过测试验证，与需求文档严格一致；在负载、并发等场景下，响应时间及吞吐量需保持稳定，无崩溃或性能衰减现象。 2. 优先级为“紧急”和“高”的缺陷完成 100% 修复。优先级为“中”和“低”的遗留缺陷需有明确解决方案与排期，且经风险评估确认不影响上线。 3. 系统及组件的容错、恢复能力和成熟性指标满足需求； 4. 系统灾备处理能力满足需求。
安全性测试	1. 安全性测试整体要求：应完成代码级安全漏洞扫描、系统级安全漏洞扫描，可选完成拒绝服务式攻击测试； 2. 代码级安全漏洞扫描：至少覆盖 OWASP Top 10、CWE、SANS 等安全静态标准，并给出不合规代码的修复建议； 3. 系统级安全漏洞扫描：至少覆盖了 CVE 已发现的安全漏洞以及隐患漏洞，对于关键性的严重级别漏洞应通过渗透测试验证； 4. 拒绝服务式攻击测试：至少完成资源比拼型（ICMP Flood、ACK Flood、Connection Flood）、软件缺陷利用型（SYN Flood、ARP Spoof、Ping of Death）等类型的拒绝服务式攻击测试，并针对被测系统表现情况给出改善建议。
可移植性测试	1. 所有功能点均通过测试验证，与需求文档严格一致。 2. 优先级为“紧急”和“高”的缺陷完成 100% 修复。优先级为“中”和“低”

	的遗留缺陷需有明确解决方案与排期，且经风险评估确认不影响上线。 3. 软件的安装/升级程序、安装/升级文档齐全，按照安装文档的指导能正确安装/升级程序； 4. 软件能在其支持的所有硬件平台、操作系统、网络环境中安装/升级、使用。
可维护性测试	1. 软件系统的需求、设计等过程文档清晰规范，不存在引起歧义的描述；相应的专家评审意见均需确认且不存在遗留未修改的问题； 2. 软件的代码实现规范，注释描述清晰，注释率不低于 20%，函数、类、全局变量以及逻辑复杂的代码段、算法等应有相应注释； 3. 每个函数最大代码行不超过 1000 行（非空、非注释行）；圈复杂度不超过 10，如有必要超过的情况应经相关专家评审通过。
应急演练	1. 应急方案演练成功； 2. 未发现重大缺陷或发现缺陷已经处理完成。
联网测试	1. 接口联网测试成功； 2. 未发现重大缺陷或发现缺陷已经处理完成。
全市场测试	1. 全市场联网测试成功； 2. 未发现重大缺陷或发现缺陷已经处理完成。
选型测试	1. 测试执行情况：所有测试场景全部执行完毕，每个测试场景在不同备选竞品上均执行完毕且不存在测试场景只运行在某个或某几个备选竞品的情况，未执行场景均有相关说明并评审通过； 2. 测试报告情况：竞品评定指标数据均采集完成，未填写项均有相关说明并评审通过；清晰地阐述竞品指标执行步骤和指标结果，对相同指标项应明确写出不同备选竞品的测试数据差异。

表 6 各测试类型准出标准

附录 A

测试类型

A.1 功能测试

功能测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
功能测试的核心目的在于通过系统化验证确保软件行为符合预期功能需求，同时为产品质量和用户体验提供保障。	测试内容/关注点具体如下： 1. 需求符合性验证。对照需求规格说明书，确认软件所有功能模块的实现准确性和完整性，通过输入数据与预期输出的匹配度检测功能逻辑是否正确。 2. 业务流程完整性保障。验证用户操作路径的连贯性及分支流程的覆盖度，确保核心业务场景无遗漏。	以黑盒测试技术为主，灰盒和白盒测试技术为辅。具体如下： 1. 黑盒测试。 场景法：模拟真实用户操作流程，验证端到端业务流程的完整性。 错误推测法：基于经验预测易错点，模拟异常场景。 边界值分析法：针对输入范围的边界点设计测试用例。 2. 灰盒测试。 数据流测试：基于系统内部数据流转路径设计用例，验证数据在各处理阶段（输入→处理→存储→输出）的完整性与一致性。 接口测试：聚焦模块间交互协议（如API、中间件），检测参数传递、数据格式、异常处理等逻辑是否符合设计规范。 状态迁移测试：针对状态机驱动系统，覆盖状态转换路径及边界条件。 3. 白盒测试。 静态结构分析法：是一种不运行程序，通过解析源代码内部结构（控制流、数据依赖、调用关系等）来定位缺陷的技术。 逻辑覆盖法：逻辑覆盖法是一种基于程序内部逻辑结构设计测试用例的测试技术，通过覆盖代码中的条件分支组合确保测试充分性。其核心

		方法按覆盖强度递增分为以下六类：语句覆盖、判定覆盖、条件覆盖、判定-条件覆盖、条件组合覆盖和路径覆盖。 基本路径测试法：通过分析程序控制流结构导出独立路径集合，设计测试用例覆盖所有逻辑分支。其实施聚焦于代码内部逻辑，确保高覆盖率与缺陷发现效率。
--	--	--

A.2 性能测试

重要信息系统性能测试工作的具体实施要求参照《证券期货业信息系统压力测试指南》。性能测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
通过模拟真实负载场景，评估系统在效率、稳定性及资源使用等方面的表现，确保其满足业务需求和技术指标。	测试内容/关注点具体如下： 1. 基准测试。通过标准化方法和工具对系统性能进行量化评估的过程，旨在建立可对比的基线，为性能优化、技术选型及版本迭代提供客观依据。 2. 负载测试。通过模拟特定负载条件（如并发用户量、数据量或运行时间）验证系统在预期压力下的表现，以识别瓶颈、确保稳定性并预测容量极限。 3. 压力测试。通过超越常规负载的极端施压（如模拟突发流量），验证系统崩溃临界点及故障恢复能力，定位性能瓶颈。 4. 容量测试。通过逐步加压至临界点，精确测量系统关键指标的极限值（如最大并发用户数、数据库记录数、事务处理量），确保系统在极限负载下仍能维持核心功能稳定运行。。 5. 业务响应时间测试。被测系统在稳定压力持续运行过程中或达到一定容量后进行业务操	具体如下： 1. 性能需求分析。包括明确性能指标阈值和识别业务模型两个阶段。 2. 性能场景设计。 混合场景：按照实际业务比例组合操作。 异常场景：模拟故障注入。 3. 工具链选型。 负载模拟：推荐JMeter模拟各类API并发测试。 资源监控：推荐Prometheus + Grafana工具组合。 4. 瓶颈分析与性能优化。 瓶颈分析：硬件层通过操作系统命令检测CPU超负载或内存泄漏迹象，定位资源耗尽问题。网络层分析网络延迟或带宽饱和现象，判断路由器或负载均衡瓶颈。应用层利用JVM命令抓取高负载线程堆栈，识别死循环、GC频繁或代码阻塞。数据库层通过检查SQL执行计划，定位慢查询或索引命中问题。 性能优化：指通过各种技术手段和策略提升软件应用、网站或系统的

	作，记录系统处理所花费的时间，以此检测业务响应时间符合性能需求。	运行效率，减少资源消耗，改善用户体验的过程。性能优化可以在多个层面进行，包括但不限于代码优化、数据库优化、硬件资源利用、网络优化等。优化整改后，通过测试验证效果，确保没有引入新的问题。上线后持续监控性能，根据反馈进行调整。
--	----------------------------------	---

A.3 可靠性测试

可靠性测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
可靠性测试是通过模拟系统可能出现的各种异常或故障等场景，来验证系统在异常处理、故障容错、数据恢复及容灾等方面的能力，并对系统的可靠性进行评估，目的是确保系统在特定的环境部署条件下满足高可用性要求，通常包括成熟性测试、容错性测试、稳定性测试和可恢复性测试。	测试内容/关注点具体如下： 1. 成熟性测试。当软件系统中出现现代码判断错误、集成中的接口错误、通讯报文错误、系统中业务逻辑错误以及其他在设计、开发等环节中造成的错误时，验证系统能消除错误造成的影响并仍可正常工作； 2. 容错性测试。当引入外部错误，如违规操作、删除数据、强行终止等严重的行为使系统(包括硬件、软件及附属程序)发生异常时，验证系统在异常情况下应具有防护性措施（处理异常的方法包括：系统自动处理和人工干预处理）； 3. 稳定性测试。通过持续性测试的方法验证系统在一定压力下长时间运行的稳定性。 4. 可恢复性测试。系统在失效状态下，恢复至正常状态的能力。失效状态应是系统崩溃或者宕机停止所有功能，而不是带故障的运行状态；正常状态应是至少保证系统的主要功能可全部运行。获取系统恢复时间（RTO）、数据恢复时间（RPO）。	具体如下： 1. 成熟性测试、容错性测试的测试方法。通过人工模拟故障场景，如进程崩溃及挂起、网络断开及延迟、服务器宕机等场景，或通过工具或手工制造异常操作或异常数据输入等，验证系统或组件在出现故障时进行有效处理的能力。 2. 稳定性测试的测试方法。7*24 小时压力测试，即仿照生产的业务配比关系，根据实际生产需要，保持不同倍数下单压力，每天运行被测系统 24 小时，连续运行 7 天。在测试期间可通过查看系统进程状态、执行功能操作、查看运行日志信息的方式来验证系统运行状态的正确性。 3. 可恢复性测试的测试方法。是模拟在出现故障或灾难导致系统失效时，通过分析系统日志或流水计算得到系统恢复时间（RTO）和数据恢复时间（RPO）指标。

A.4 安全测试

确认被测试系统符合系统安全性要求。安全测试在系统新上线和系统产生重大架构变更的场景下必须执行。

安全测试的主要目的、测试内容及测试技术如下：

A.4.1 目的

通过安全测试手段和方法，验证软件的安全特性实现与预期一致、检验软件产品对各种攻击情况的防范能力，从而保障产品的安全质量。

A.4.2 测试内容/关注点

具体如下：

1.配置管理。按照信息安全要求对操作系统、应用程序、数据库、网络设备等进行安全配置。

2.资源利用。应用程序内存使用和资源竞合问题，如变量未初始化、数组越界、内存溢出、资源泄露、进程线程异常等。

3.身份鉴别。对登录操作系统、数据库系统和应用系统的用户进行了身份标识和鉴别，如应使用强密码策略、限定连续登录尝试次数、使用有效验证码等。

4.访问控制。用户应根据自己的权限大小来访问系统资源（如服务器、目录、文件等），不得越权访问。

5.数据保护。应用程序存储敏感信息的情况，存储的敏感信息应进行加密，包含但不限于密码、密钥等敏感信息应加密写入缓存、文件系统、数据库中等。

6. 通信安全。应用程序使用安全通信协议（如 SSL、TLS）的情况，对敏感信息的传输应进行加密等。

7. 会话管理。本地及服务器端对会话超时的设置，会话结束后应用程序使用过的敏感信息应从内存中删除，合法用户的会话不被劫持等。

8. 数据验证。应用程序在使用前正确验证来自客户端或外界输入数据的情况，避免发生如跨站脚本攻击、命令注入、文件注入、SQL 注入等漏洞。

9. 安全审计。对网络系统中的网络设备运行状况、网络流量、用户行为等进行日志记录。审计记录应包括事件的日期和时间、用户、事件类型、事件成功及其他审计相关的信息等。

A.4.3 测试技术

常见的安全性测试技术有安全功能检查、代码安全测试、漏洞扫描、渗透测试和模糊测试。

具体如下：

1. 安全功能检查。是指测试人员通过对相关系统管理人员进行访谈和对测试对象（如相关系统、各类设备、基线标准、开发设计文档）进行观察、验证、分析以确认测试对象符合相关国家法律法规、标准或监管规定的信息安全要求以及安全功能需求的过程。

2. 代码安全测试。分为静态检测和动态检测。静态检测是使用代码检测软件进行代码静态扫描，并对扫描后结果进行人工分析来发现定位漏洞的过程；动态检测即在运行时进行错误检查，主要通过将已

经写好的程序转换成一个新程序（其功能与原程序是等价的），新程序包含有一些额外的代码，它们的作用是在程序执行期间检查错误。

3. 漏洞扫描。是指利用专业的漏洞扫描工具，基于公共漏洞和暴露（CVE）漏洞数据库或特征库，通过扫描、探测等方法对目标系统的安全情况进行检测，发现可利用的漏洞。

4. 渗透测试。是通过模拟恶意黑客的攻击方法对目标系统进行模拟入侵，以找出系统中的脆弱点、技术缺陷和漏洞。

5. 模糊测试。是一种通过提供非预期的输入并监视异常结果来发现软件故障的方法，是一个自动的或半自动的过程，通常包括识别目标、识别输入、生成模糊测试数据、执行模糊测试、监视异常、确定可利用性等阶段。

A.5 可移植性测试

确认被测试系统符合客户业务需求规定的可移植指标。

可移植性测试的主要目的、测试内容及测试技术如下：

A.5.1 目的

通过模拟部署在不同的运行平台来验证系统运行的适应性，目的是确保系统在运行环境发生变化时运行及处理可满足预期要求，通常包括可安装性测试、兼容性测试、适应性测试和可替换性测试。

A.5.2 测试内容/关注点

测试内容/关注点包含可安装性测试、兼容性测试、适应性测试和可替换性测试，具体如下：

1. 可安装性测试。在目标环境对软件安装程序所进行的测试，包

括操作系统安装软件或在客户个人电脑上安装软件产品的安装向导软件。主要关注：安装文档应齐全、软件安装的程序文件应齐全、文件格式应与安装指导中要求的文件格式相同；所有预置的数据应齐全；软硬件环境的配置应合理；所有文件应正确产生并确有所需要的内容等。

2. 兼容性测试。如果不存在相互依赖关系的计算机系统可在同一环境（例如：同一个硬件平台）中运行，而不影响彼此的行为（如资源冲突），则称之为兼容，包括应用软件与硬件平台、应用软件与操作系统、操作系统与硬件平台间的兼容性，例如：当新的或升级之后的软件被大量装入已经安装了应用程序的环境（例如：服务器）时，应执行兼容性测试。

3. 适应性测试。软件系统能在所有特定的目标环境（硬件、软件、中间件、操作系统等）中正确运行，包括软件系统在不同处理器配置下的适应性、在台式机与专业服务器下的适应性、在不同操作系统下的适应性、在不同网络环境下的适应性。

4. 可替换性测试。软件系统中组件能被替换的能力以及软件升级引起的新旧版本的可替换性。

A.5.3 测试技术

可安装性测试、兼容性测试、适用性测试和可替换性测试的测试方法如下：

1. 可安装性测试的测试方法。模拟软件系统在目标环境中进行安装、升级及卸载等操作，验证其正常运行能力。

2. 兼容性测试的测试方法。模拟软件系统在目标环境中与其他应用程序、升级后操作系统等软硬件共存的场景，验证其正常运行能力。

3. 适应性测试的测试方法。模拟软件系统在不同的特定目标环境下运行的场景，验证其正常运行能力。

4. 可替换性测试的测试方法。模拟系统在不同软件组件或者版本上运行的场景，验证其正常运行能力，尤其对于以第三方商业软件为特定组件的软件系统。

A.6 可维护性测试

确认被测试系统符合客户业务需求规定的可维护性指标。

可维护性测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
通过评审系统设计结构、开发语言、系统文档及用户文档来评估系统后续易于修改、优化及能满足新需求的开发，目的是确保系统在交付后直至到期被淘汰的整个时期内易于理解和改进。	测试内容/关注点具体如下： 1. 软件系统具有错误的易分析性。内部：检验设计、文档、代码的书写规格，代码注释和错误标注规范性和充分性等；外部：检验错误发生时系统的响应结果，便于发现和准确定位问题。 2. 软件系统具有使指定的修改可被实现的能力，包括编码、设计和文档的更改，实现参数文件、数据库、表、操作方式、个性化配置。 3. 修改后的软件系统仍然能达到所需功能要求和运行指标。	常见的维护性测试技术有：专家评审法、技术测试法和用户调查法。 1. 专家评审法。是一种主观的评审方法。评审时，应根据被评审对象和评审目的，设计评审项目表，列出栏目、分值、权重和打分规则。专家根据自身的经验与认知，进行判断打分，然后根据专家的权重和统计规则计算最终评分。 2. 技术测试法。是一种客观的评分方法。技术测试时，可依据被测对象和测试目的，选择适用的自动化测试工具，也可人工进行手动测试。技术测试获得的结果是一种量化的测量结果。 3. 用户调查法。是一种面向用户群的问卷征询方法。用户调查时，应根据调查的目的和特定的用户群，设计调查表，让被调查对象填写并反馈，调查表回收数应达到一定的数量，并不低于发出的适当比

		例。然后对回收的调查表进行汇总计算，其计算值作为用户调查的结果。
--	--	----------------------------------

A.7 应急演练

确认被测试系统符合客户业务需求规定的应急演练要求。

应急演练的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
充分分析可能出现的各种突发事件并对其制定相应应急预案，通过评审及演练，检验预案的实用性、可用性、可靠性，相关人员应急行动实施的有效性以及避免事故、防止事故、抵抗事故的能力，目的是确保在上线过程或上线后突发重大事故、设备故障等情况下，相关人员可及时反应、妥善处理，从而使事故造成的影响和损失降到最低。	具体如下： 1. 应急预案具有完整性、正确性、一致性及易理解性，符合应急要求。 2. 突发事件场景考虑全面，应急过程所用工具、脚本可用，按照应急预案中描述的操作步骤可将系统恢复至正常运行状态。 3. 系统恢复时间（RTO）、数据恢复时间（RPO）符合应急要求。与可靠性测试的区别在于测试环境不同，且测试场景更贴近于生产已经发生过的或者可能发生的紧急情况。	以黑盒测试为主，通过应急预案静态检查、应急预案模拟演练、分析系统日志或流水等方法，保证应急预案的有效性和正确性。

A.8 联网测试

确认被测试系统符合客户业务需求规定的联网测试要求。

联网测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
市场核心机构组织部分或全部市场参与方，在测试环境或准生产环境开展的接口处理及业务流程测试，目的是验证交易所和各参与方的接口处理及业务流程的正确性，以及网络的连通性。此测试类型可根据实际情况进行裁剪。	具体如下： 1. 验证机构间接口含义理解及处理的一致性、正确性。 2. 验证机构间业务及数据交互流程的正确性。	以黑盒测试为主，参测各方通过场景设计对测试关注点进行验证。

A.9 全市场测试

通过模拟真实的市场行为来验证交易所和各参与方的系统功能、性能、可靠性以及整体业务流程等功能测试和非功能测试与需求规格

说明书的一致性，确认系统具备上线条件。

全市场测试的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
在软件竞品选择的过程中，使用测试理论和方法对软件竞品进行多种技术指标的评测，通过多个维度对软件竞品进行评测对比，为软件竞品选择提供参考指标数据。	具体如下： 1. 各参与者系统和具体业务的功能及流程应正确，性能、可靠性等非功能测试应满足需求规格说明书； 2. 系统上线所需线路等技术条件应完备并可正常使用； 3. 参与市场业务各方上线条件应已完备； 4. 新程序或系统上线后，相关系统能正常适应； 5. 可根据实际情况需要，增加通关测试。	以黑盒测试为主，参测各方通过场景设计对测试关注点进行验证。

A.10 选型测试

确认软件竞品符合评测要求，为软件竞品选择提供参考依据。

选型的主要目的、测试内容及测试技术如下：

目的	测试内容/关注点	测试技术
在软件竞品选择的过程中，使用测试理论和方法对软件竞品进行多种技术指标的评测，通过多个维度对软件竞品进行评测对比，为软件竞品选择提供参考指标数据。	具体如下： 1. 软件可靠性选型：使用相同的可靠性测试方法针对不同软件竞品进行可靠性测试，检测不同竞品的可靠性指标差异； 2. 软件性能选型：使用相同硬件环境和相同的性能测试方法针对不同的软件竞品进行性能测试，检测不同竞品的性能指标差异； 3. 安全选型：使用相同的安全测试方法针对不	根据选型目的选择一种或多种测试方法进行选型测试，具体测试方法见 A.1、A.2、A.3 和 A.4。

	同的软件竞品进行安全测试，检测不同竞品的安全指标差异； 4. 软件功能选型：根据产品需求对软件功能进行功能测试，检测不同竞品在满足相同需求条件下功能实现的差异性。	
--	---	--

附 录 B

软件测试模板

B.1 测试工作量评估表

活动名称	测试阶段	责任人	评审	计 划 开 始日期	计 划 结 束日期	工 作 量 (人/天)	资 源 (人)
案例编写	测试方案		是				
1 轮测试	执行		否				
2 轮测试	执行		否				

B.2 测试计划

1	简介
1.1	项目概述
1.2	基本假设
1.3	参考资料
2	测试目标
3	测试范围
4	测试策略
4.1	测试级别
4.2	业务功能
4.3	非功能
4.4	测试依据版本
4.5	风险评估
4.6	测试方法
5	测试计划
5.1	角色及职责
5.2	测试工作里程碑
5.3	测试组人员需求
6	测试环境
6.1	测试系统
6.2	测试数据
6.3	测试工具
7	测试管理与报告程序
7.1	测试管理
7.1.1	问题追踪管理流程
7.1.2	变更管理流程
7.1.3	进度追踪流程
7.2	测试报告

B.3 测试案例模板

用例名称	用例等级	分组	被测需求编号	前置步骤	测试步骤	预期结果	标签	评审人	评审结果

B. 4 测试缺陷列表

标题	优先级	状态	父工作项	描述	创建人	创建时间	经办人	版本	迭代	缺陷类型	提交人	解决人	来源需求

B.5 测试报告模板

基本信息

起止时间	YYYY-MM-DD - YYYY-MM-DD
负责人	
参与人员	
项目目标	
概况	本次测试共涉及 XX 个需求。 共测试了 XX 个版本。共编写了 XX 个用例，共执行个 XX 用例，产生了 XX 个结果，阻塞 X 条，失败的用例 XX 个。 共发现 XX 个缺陷，XX 个为“紧急”优先级缺陷，XX 个为“高“优先级缺陷；遗留 XX 个缺陷，其中有 XX 个”紧急”和”高”优先级缺陷；缺陷修复率为 XX%。

测试环境

系统名称	环境信息
XX 系统	中间件/数据库/模拟报盘/模拟平台

系统名称	当前测试版本	升级至系统版本
XX 系统		

测试计划

ID	测试轮次	责任人	测试周期
1	系统测试		YYYY-MM-DD - YYYY-MM-DD
2	UAT 测试、回归测试		YYYY-MM-DD - YYYY-MM-DD
3	生产环境验证		YYYY-MM-DD - YYYY-MM-DD

测试内容

需求/缺陷 ID	需求/缺陷名称	测试人员	状态

测试案例

案例 ID	案例名称	测试人员	执行状态

缺陷清单

ID	优先级	缺陷标题	创建者	解决者	解决日期	状态

遗留的缺陷

ID	优先级	缺陷标题	创建者	解决者	解决日期	状态

总结

本次测试是否通过。

【签名图片】

附 录 C

测试类型选择原则

根据软件产品可能涉及的测试内容需选择适合的测试类型，具体选择原则如下：

类别	内容	测试类型									
		功能测试	性能测试	可靠性测试	安全性测试	可移植性测试	可维护性测试	应急演练	联网测试	全市场测试	选型测试
基础架构	通信线路	◎	●	●	●					●	●
	网络结构	●	●	●	●						●
	主机硬件	◎	●	●	●	●	●				●
	操作系统及编译环境	●	●	●	●	●	●				●
	存储设备	◎	●	●	●	●	●				●
	灾备架构	●	●	●			●	●			●
应用软件	数据交换接口	●	●	●	●				●	●	●
	数据存储方式	●	●	●	●						●
	应用程序	●	●	●	●	●	●				●
	数据库等第三方应用	●	●	●	●	●	●				●
业务规则	根据实际业务情况填写，如交易规则、清算业务	●								●	
上线运维	上线步骤						●	●		●	
	应急方案							●			
	数据迁移	●					●	●		●	
	运维管理						●	●		●	
◎可选择 ○必选											

注：根据被测软件产品的测试内容，结合本表格进行摘选。