

第一创业证券企业标准

Q/FCSC 004-2024

测试开发规范

Specification of Test Development

2024-08 发布

2024-08 实施

第一创业证券股份有限公司

目 次

前 言	II
1 范围	1
2 引用文件	1
3 定义和术语	1
4 自动化测试工具规范	2
5 自动化测试流程规范	2
6 自动化脚本编写规范	3

前 言

本文件按照 GB/T1.1-2020 《标准化工作导则 第一部分：标准化文件的结构和起草规则》的规定起草。

本文件由第一创业证券股份有限公司提出。

本文件由第一创业证券股份有限公司归口。

本文件主要起草部门：第一创业证券股份有限公司信息技术中心。

本文件主要起草人：刘晔、陈凯。

测试开发规范

1 范围

本规范适用于第一创业证券股份有限公司所有测试开发工作，包括自动化的测试工具、测试流程、脚本编写等各个环节的活动。

2 引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是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。

JR/T 0191-2020 证券期货业软件测试指南 软件安全测试。

3 定义和术语

自动化测试 (Automated Testing)

自动化测试是指利用软件测试工具和脚本来执行测试任务，它通过自动化执行测试用例、生成测试报告、比较实际结果与期望结果等方式来评估软件系统的质量。它可以帮助测试团队更快速地执行重复性的测试任务，减少人工测试的工作量，提高测试效率。

功能测试 (Functional Testing)

功能测试是一种最基本、常见的软件测试方法，旨在验证软件系统是否按照需求规格说明书或功能规格说明书中所描述的功能进行正确的实现。它主要关注软件系统的功能性是否符合用户需求和预期。在功能测试中，测试团队会根据需求规格说明书或功能规格说明书，基于不同的测试用例和测试场景，对软件系统的各项功能进行测试。测试人员会模拟用户的操作行为和输入数据，以验证系统的功能是否正常工作，包括功能的正确性、完整性、可靠性、有效性等。

回归测试 (Regression Testing)

回归测试是一种软件测试方法，用于验证在软件系统进行了更改或修复后，已测试和正常工作的功能是否仍然保持正常操作。其目的是确保新的更改没有引入新的错误或导致已有的功能发生退化。在回归测试中，测试团队会重新执行一部分或全部的已有测试用例，包括功能测试、性能测试、安全测试等，以验证系统在进行更改后是否仍然正常工作。回归测试旨在捕获由于新的更改引入的潜在问题，并确保系统的稳定性和质量。

冒烟测试 (Smoke Testing)

冒烟测试是一种软件测试方法，旨在快速检查软件系统的关键功能是否能够基本运行。它常用于软件测试过程的早期阶段，以帮助测试人员快速确定系统是否可以进行详细的正式测试。在冒烟测试中，测试团队会执行一组简单、基本的测试用例，目的是验证系统的核心功能是否存在严重的问题或错误。这些测试用例通常涵盖了系统的主要功能模块和关键路径，以及常见的典型使用情况。

4 自动化测试工具规范

4.1 选型建议

平台宜采用 C/S、B/S 架构相结合的方式进行搭建。当前可选择的类型主要包括：

- a) 金证自动化测试平台；
- b) 自建自动化测试平台；
- c) 各类自动化测试工具：如 RobotFramework、Selenium、Appium、Pytest、Jmeter、Junit 等。

建议使用金证自动化测试平台，若存在平台无法满足需求的情况，可尝试二、三所述方式。

4.2 功能模块

工具宜包括但不限于以下功能：

- a) 脚本管理：支持用户根据产品、版本、业务类型等信息自建目录分层管理；支持接口/UI/APP 等多种类型自动化脚本编写；支持文本化、表格化、图形化等设计模式；支持自定义变量、模块化、导入导出、备份还原等功能；支持脚本版本管理（包括脚本编辑记录、差异对比、按记录回退等功能；对于无法迁移到平台的脚本，建议使用公司 gitlab 进行版本管理）；
- b) 脚本执行：支持脚本调试、批量执行、定时执行、串行/并发执行、执行记录等功能；
- c) 测试报告：支持自动化报告生成、比对、导出、模板定制、邮件发送以及历史报告查询等功能；
- d) 数据管理：支持对脚本所涉测试数据、前置数据的管理；支持 mock 挡板；支持多样化的数据统计等功能；
- e) 缺陷管理：支持缺陷的创建、自动创建、查询、统计等功能；
- f) 流水线调用：支持 DevOps 平台、Jenkins 等调用脚本批量执行。

5 自动化测试流程规范

5.1 自动化测试适用范围

可应用于功能测试、回归测试、冒烟测试、持续集成测试、环境巡检等。

5.2 实施自动化测试流程

- a) 可行性评估：项目或产品是否适合进行自动化测试，可根据功能/UI 变动频率、回归测试频率、系统可测性、测试场景复杂度等因素进行综合评估；
- b) 测试需求分析：可根据业务需求确定测试范围，分解功能点，确定测试粒度等；
- c) 自动化用例设计及评审：可根据测试需求设计测试场景，按自动化用例设计模板进行用例设计，完成后提交测试负责人、产品、开发进行评审，并依据评审意见进行调整；
- d) 环境准备：准备执行所需的测试环境，可通过流水线进行自动化部署，环境的切换需支持参数化配置；
- e) 数据准备：构造执行所需的测试数据，可存储至数据源（如数据库、txt、csv 等）；数据源需支持自动备份及还原；对无法构造的测试数据，可采用 mock 挡板等方式进行模拟；
- f) 自动化脚本编写：依据评审后的自动化用例进行脚本开发、调试，详见第 6 章节自动化测试脚本编写规范；
- g) 自动化测试执行：对已完成调试的自动化用例进行批量执行，可按实际情况选择并发、串行、定时、流水线调用等执行方式；
- h) 测试结果分析：对自动化执行结果进行分析，对执行失败结果进行缺陷认定、记录，并通知相关开发人员进行缺陷分析、修复；针对已修复的缺陷进行复测；对非缺陷导致的执行失败，进行脚本调整；
- i) 自动化测试报告输出：完成测试执行、结果分析以及缺陷修复后，生成自动化测试报告，并邮件发送给项目相关人员。

6 自动化脚本编写规范

6.1 自动化脚本设计总体原则

应提高自动化脚本的可读性、稳定性、可维护性、覆盖率。

6.2 可读性要求规范

- a) 脚本命名：脚本命名只准许中英文、数字、下划线，长度建议 15 字以内；同一个版本下，脚本名称不应该重复，可通过前缀和完整描述加以区分；脚本名称应包含业务名称、接口名称、场景/功能、操作内容、预期结果等信息，方便通过脚本名称快速了解脚本内容，如：市价委托-普通账户-检查是否有开通市价委托权限-办理成功；

- b) 参数命名：参数命名只准许中英文、数字、下划线，长度建议 15 字以内；参数命名应简洁明了、表意清晰；对于变量参数应使用驼峰命名规则；版本级别全局变量参数命名建议以英文 G_开头，局部套件级别变量参数命名建议以 L_开头；常量参数应全部使用大写字母，并使用“_”作为分隔符；在参数备注中使用中文描述参数；不应该英文、拼音混用；
- c) 关键字命名：关键字命名只准许中英文、数字、下划线，长度建议 15 字以内；同一个版本下，关键字名称不应该重复，可加业务前缀加以区分；
- d) 注释：对脚本、模块、输入参数、输出参数、操作步骤、断言等应添加必要的注释，描述其功能、注意事项、版本号等信息；注释应简洁明了、表意准确。

6.3 稳定性要求规范

- a) 测试数据：应避免在脚本中固化测试数据，应灵活的从数据库、txt、csv 等可配置化的数据源或接口中获取；对于不可复用的测试数据，应封装数据构造脚本/接口动态生成数据；对于测试数据的有效性可添加必要的校验；对于可能存在干扰或临时修改的数据，应添加前置或后置数据处理进行清理或还原；对于暂时无法获取的接口数据，可采用 mock 挡板；
- b) 降低耦合性：用例或步骤宜能够独立运行，降低用例之间的耦合性，不建议依赖其他用例集，避免因所依赖的用例执行失败而影响当前用例的执行；
- c) 加强异常处理：对可能出现异常的步骤应添加异常处理、重试机制，避免脚本报错直接退出；
- d) 持续验证：应保持一定的执行频率，避免长时间未执行导致无法及时更新脚本；
- e) 合理延时：对需要添加延时处理的地方，应结合条件判断+循环的方式，递增短延时，避免盲目增大延时；
- f) 设备/环境兼容：脚本应能够在不同终端设备或环境中执行。

6.4 可维护性要求规范

- a) 参数化：对可能发生变更的内容，应参数化，如配置的 IP 地址、端口等；
- b) 模块化：对于复用率高的脚本或步骤，宜抽取为公共模块，以方便后期维护；
- c) 简化逻辑：建议简化脚本逻辑，避免复杂的逻辑判断。

6.5 覆盖率要求规范

- a) 场景覆盖：从业务测试角度出发，可通过评审确定自动化测试覆盖范围；优先覆盖关键场景、高频场景等；
- b) 流程覆盖：建议覆盖被测业务的主流程、分支流程等；

- c) 接口覆盖：建议覆盖被测系统接口，完善接口参数校验；
- d) 持续集成：可通过持续集成，不断增加对场景、流程、接口的覆盖。