

自动化测试平台 解决方案

上海泽众软件科技有限公司

2025.2

咨询电话：400-035-7887

目录

CONTENTS

01

Part 01

公司介绍

02

Part 02

平台介绍

03

Part 03

平台功能介绍

04

Part 04

客户案例



PART ONE

公司介绍

上海泽众软件科技有限公司

泽众业务范围

测试工具与服务

一站式自动化测试平台、一站式全链路性能测试平台、自动化测试、移动自动化测试、性能测试工具、性能监控软件、测试管理等等。

服务包括接口自动化测试、性能测试、第三方验收测试、功能测试、安全测试等等服务

RPA机器人

泽众RPA机器人是一款能够模拟业务人员来执行重复性业务的自动化平台。已经在电力、银行、保险、证券、电商、物流、财务、人事、IT运维等方面具有广泛的应用。



业务范围

云真机与兼容性测试

泽众云真机通过网页操作接入云端的真实手机，覆盖市场海量机型，远程操控快速流畅，用户随时随地进行测试，调试应用，快速定位问题。

兼容性测试服务包括，移动应用标准兼容性测试和深度兼容性测试，覆盖了智能硬件、家居、医疗、交通等行业的适配测试。

研发测试管理

主要包括IPD，有项目管理、需求管理、产品管理、生命周期、知识库、资源库和Devops（代码库）。

人员及分支机构



在**上海**设**泽众**总部，在国内多个城市设有分支机构，拥有**680+**员工。

全国分支机构：

研发中心：苏州、西安、深圳

办事处：北京、南京、成都、福州等

项目组：杭州、宁波、太原、重庆等

在**苏州**设有云真机实验室

自主研发产品介绍



金融

建设银行，交通银行，兴业银行，平安银行，南京银行，江苏银行，广州银行，河北银行，上海农商行，哈尔滨银行，浙江农信，吉林农信，稠州农商，张家港农商，吴江农商，国泰君安，中信建投，银河证券，中泰证券，华泰证券，国联证券，九州证券，山西证券，光大证券，恒泰证券，东方证券等90多家客户

企业

中国移动，中国电信，大庆油田，长庆油田，国家电网，上汽集团，上海电力，如新科技，海能达，浪潮科技，通广集团，天地玛珂，西安鼎云电子，北京利云，搜车网，久远银海，沈阳沈飞电子，北京中彩在线，四川云从科技，TCL，易佰信科技，普陀教育局，鹏微科技，白云机场，三甲医疗等600多家客户；

科研

中国标准化研究院，中国工程物理研究院，中国铁道科学研究院，中国科学院深圳先进技术研究院，北京软件产品质量检测检验中心，内蒙古质检院，湖北质检院，陕西质检院，贵州检验检测中心，国家电子计算机质量监督检验中心，浙江标准化研究院，中国科学信息工程研究院，沈阳医疗检测等70多家客户；

其它

解放军63892部队，中国兵器，716研究所，西安二炮，国防科技大学，火箭工程大学，华中科技大学，上海二工大，郑州大学，广州大学华软学院，上海电子信息，健雄职业，宁波大红鹰，周口师范，洛阳师范，番禺职业，肇庆学院，上海版专学校，武昌学院，武汉大学，内蒙工业大学等270多家客户

资质证书



高新技术企业证书



科技小巨人企业



软件企业



互认证明&认证证书



CMMI3



质量管理体系认证证书



ISO20000认证证书



ISO27001认证证书



软件产品证书



高新技术成果转化项目



优胜产品&优胜企业



产品著作权证书



PART
TWO
平台介绍

上海泽众软件科技有限公司

为什么需要测试自动化平台？

随着软件开发的复杂性的提升，传统的手动测试已经无法满足快速交付、高质量和缩短测试周期的要求。测试自动化平台通过自动化测试过程，可以大大提高测试效率，减少错误，节省时间和资源。

01

提高测试效率

测试自动化平台可以自动执行测试用例，并根据预先定义的预期结果进行验证。它可以快速地检测出程序中的错误和缺陷，节省了大量的人力和时间。

02

加速测试周期

测试自动化平台可以在较短的时间内执行大量的测试用例，避免了手动测试中的重复劳动。它可以在短时间内完成大规模的测试，提高了测试的覆盖率。

03

标准化自动化测试体系

通过自动化平台建立自动化测试体系。

泽众一体化自动测试平台介绍



质量就是生命，效率决定未来

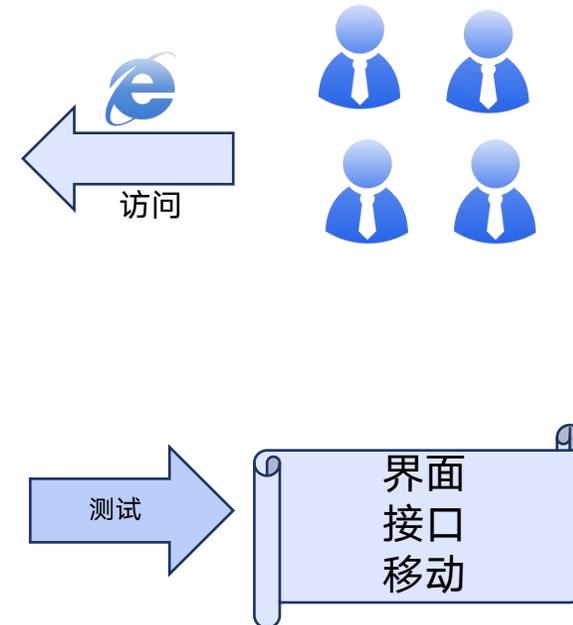
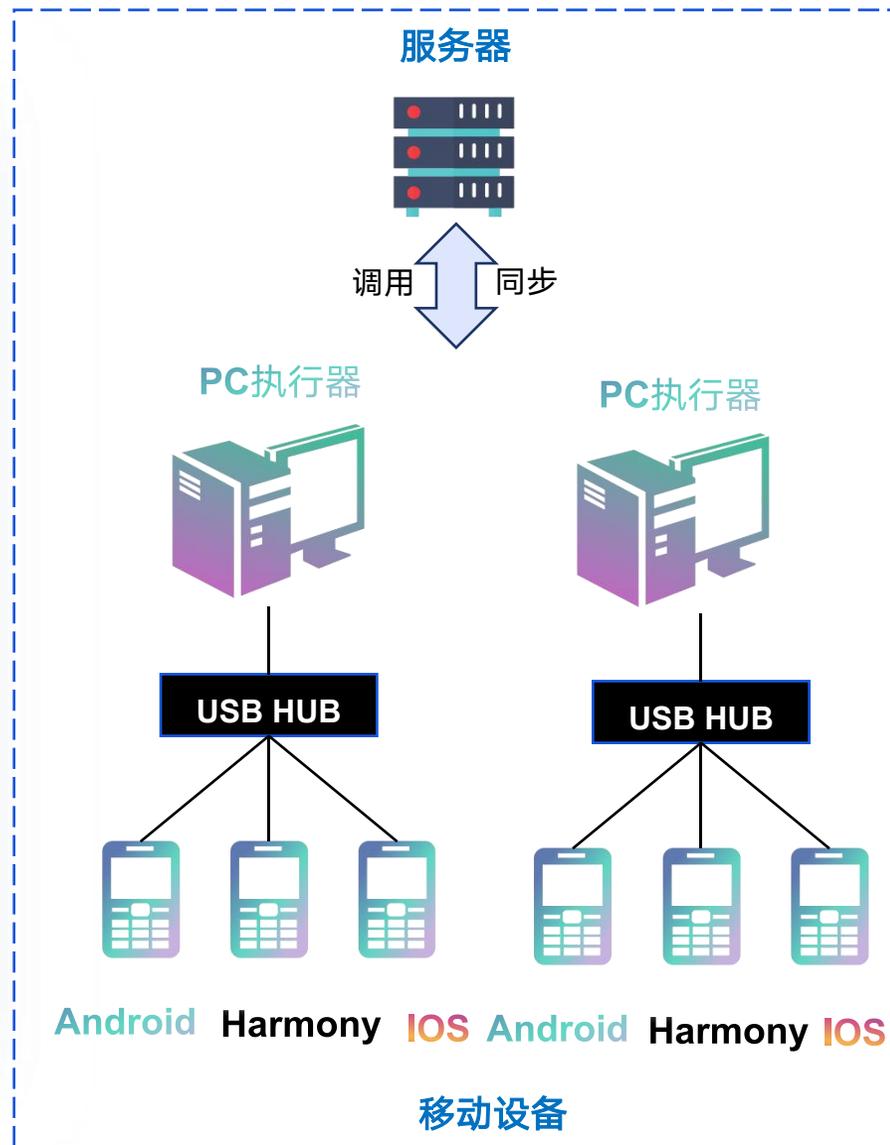
TestOne是上海泽众软件科技有限公司自主研发的一体化测试系统。基于 B/S 体系的微服务架构，集系统管理、项目管理、测试管理、缺陷管理等功能于一体，覆盖了

界面自动化测试

接口自动化测试

移动自动化测试

等测试类型，完整覆盖自动化测试项目的全过程，可快速将自动化测试管理体系建立，提高测试效率与质量。

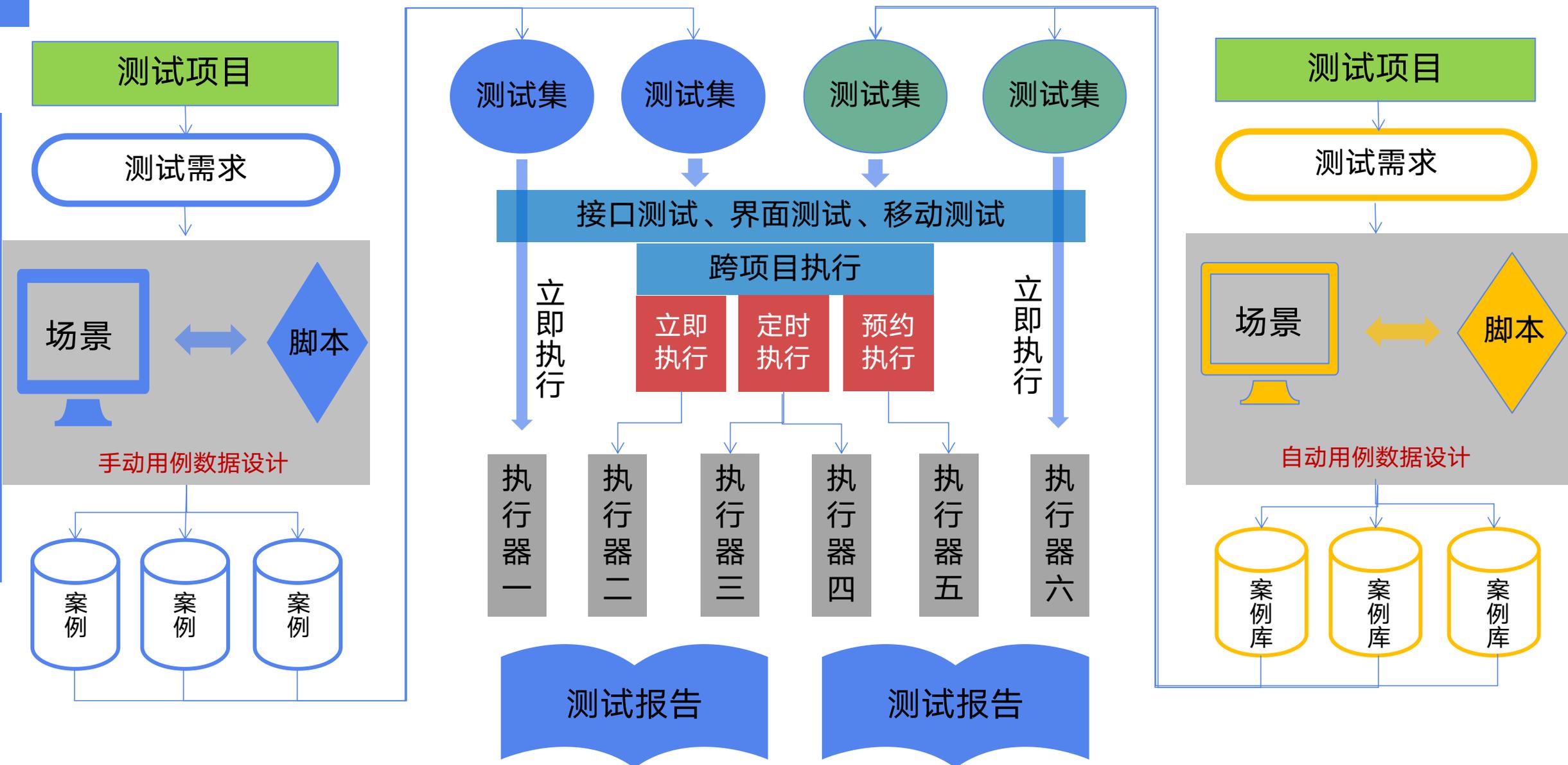


服务：
Nginx、Redis、ES、Minio
前端：
Vue
数据库：
Mysql、Oracle、达梦、OceanBase

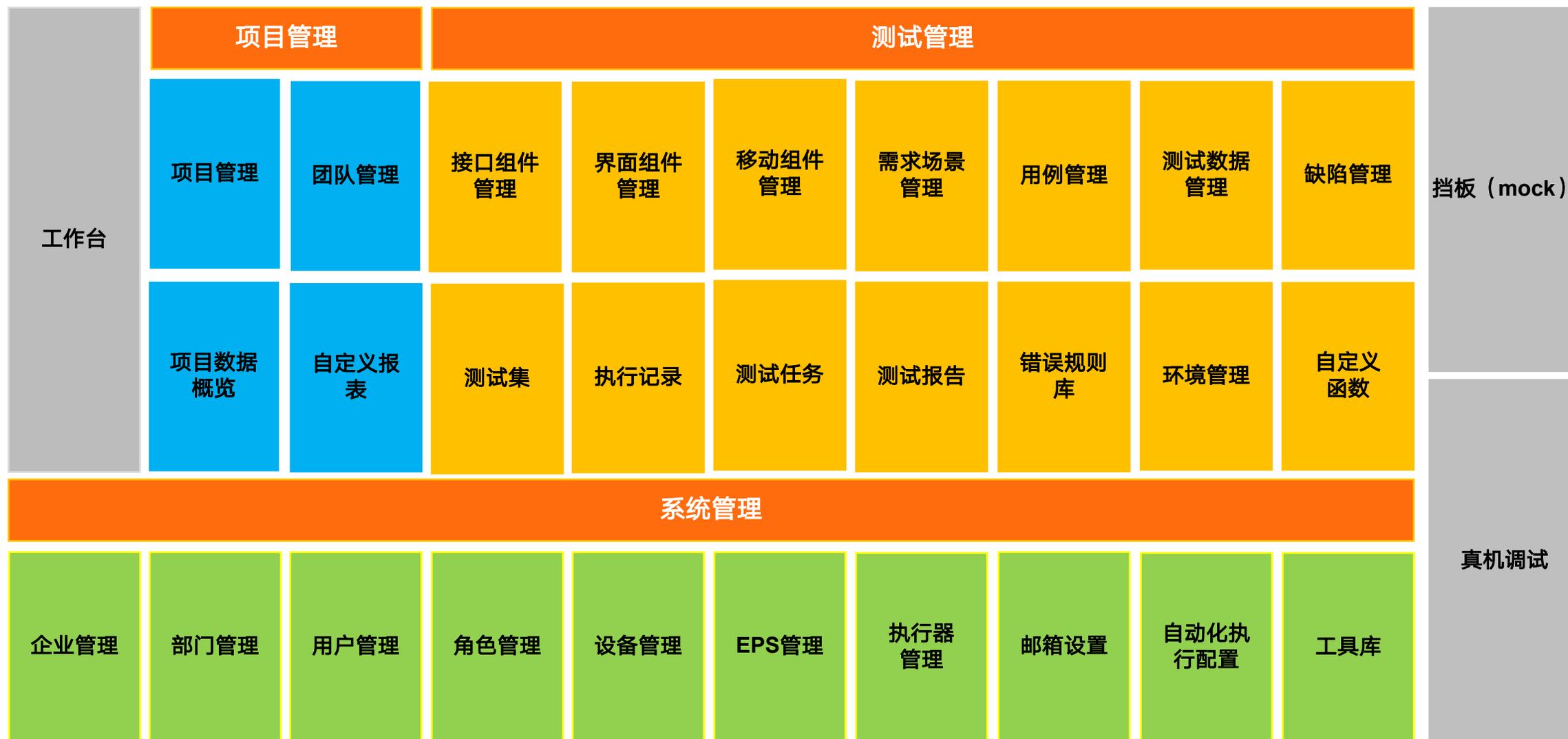
自动化测试组织流程管理



质量就是生命，效率决定未来



平台功能模块



平台功能特点

功能		S
界面自动化	浏览器类型	Chrome、FireFox、Edge、IE、搜狗、360、紫鸟
	桌面应用程序	.net(c#、vb、SWF)、visual c++、Java SWT/AWT/SWING、Flex、Delphi、Powerbuilder、Silverlight对象、WPF对象、QT对象
移动自动化	操作系统类型	Android、IOS、Harmony（原生）
	XPath	支持查看XPath路径、属性、对象等；可通过XPath直接生成脚本语句；
接口自动化	协议类型	HTTP/HTTPS、TCP/IP、支持通过jar自定义扩展协议类型 第三方工具脚本：Postman、Jmeter
视觉识别		支持OCR文字识别、图像识别、AI物体检测，举例：滑块验证码、图片验证码等
脚本生成方式		录制生成 自然语言 脚本，拖拽编辑方式维护脚本
数据驱动		支持脚本与数据分离、以数据进行构建测试用例；支持多种数据来源如:Excel、数据库、传递等；支持数据标记、以场景法、边界值、等价类基于数据规则方式进行正交自动生成用例
错误分类		支持批量执行后能自动匹配错误原因分类
自定义		支持自定义方法、公共函数的操作指令(界面、接口、移动)、Mock挡板（服务器、客户端）
精准测试		可统计回归代码行覆盖率、分支覆盖率、函数覆盖率、用例代码行密度比、回归缺陷逃逸率等

平台优势

测试智能化

- 智能执行
- 视觉识别与低代码
- 用例设计智能化
- 自动错误日志分析与归类、搜索

多维自动化

- 面向服务的接口测试
- 面向UI的端到端测试
- 支持不同类型脚本的用例

服务化

- 测试工具的服务化
- 微服务架构横向扩展
- 独立的数据管理平台服务

集成与扩展

- 与测管平台微服务无缝集成
- 接口测试的adapter结构扩展自定义协议



PART
TWO
平台功能介绍

上海泽众软件科技有限公司

目录

01

自动化测试脚本设计

02

自动化测试用例设计

03

自动化测试执行管理

04

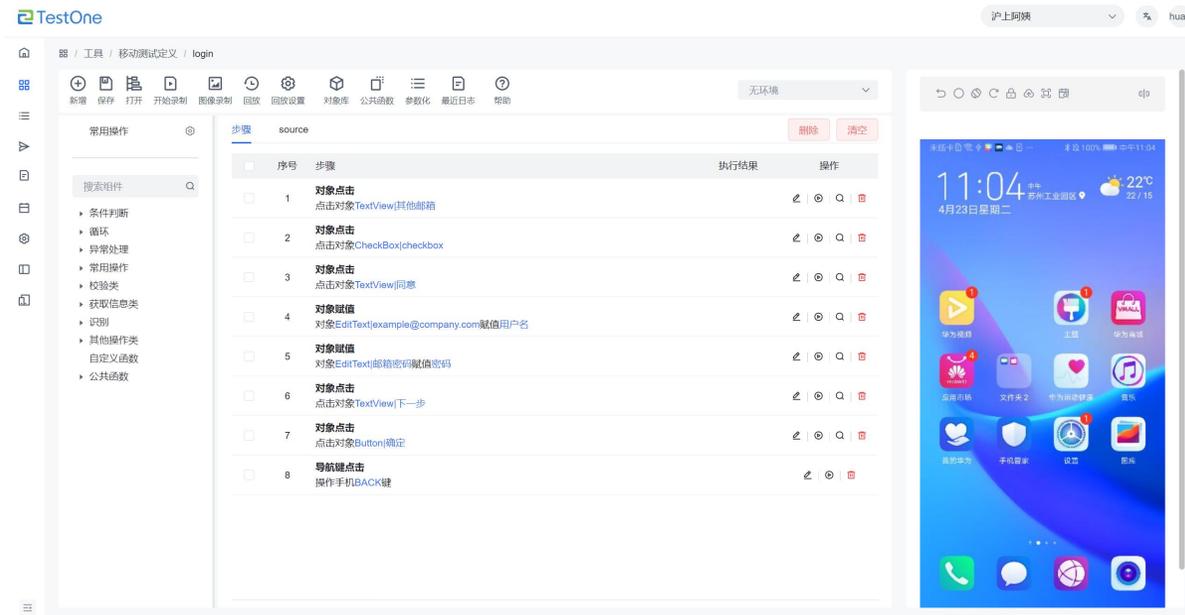
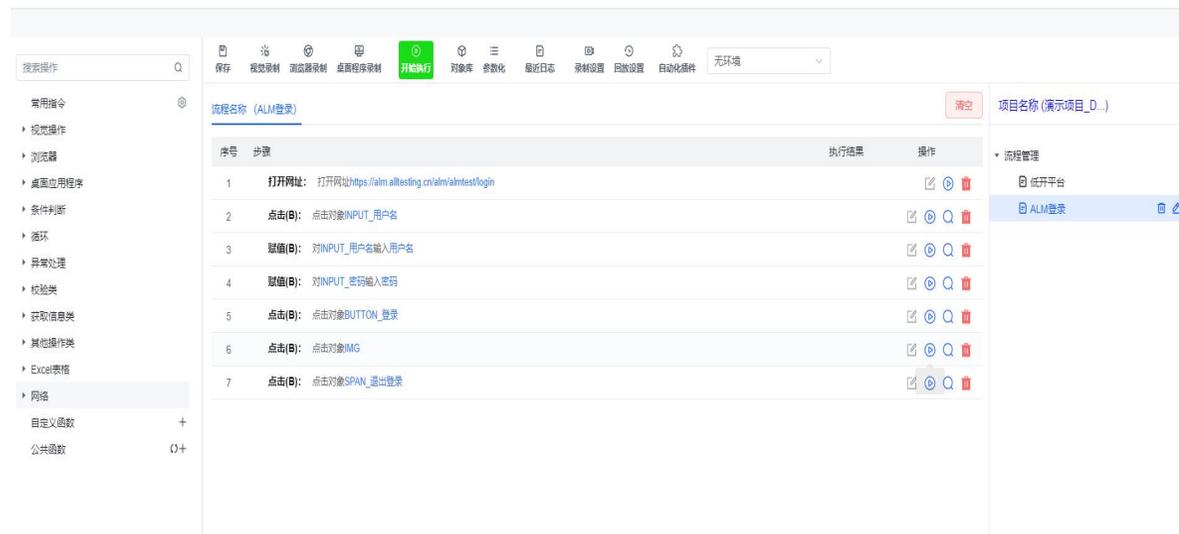
自动化测试项目管理

自然语言展示业务逻辑

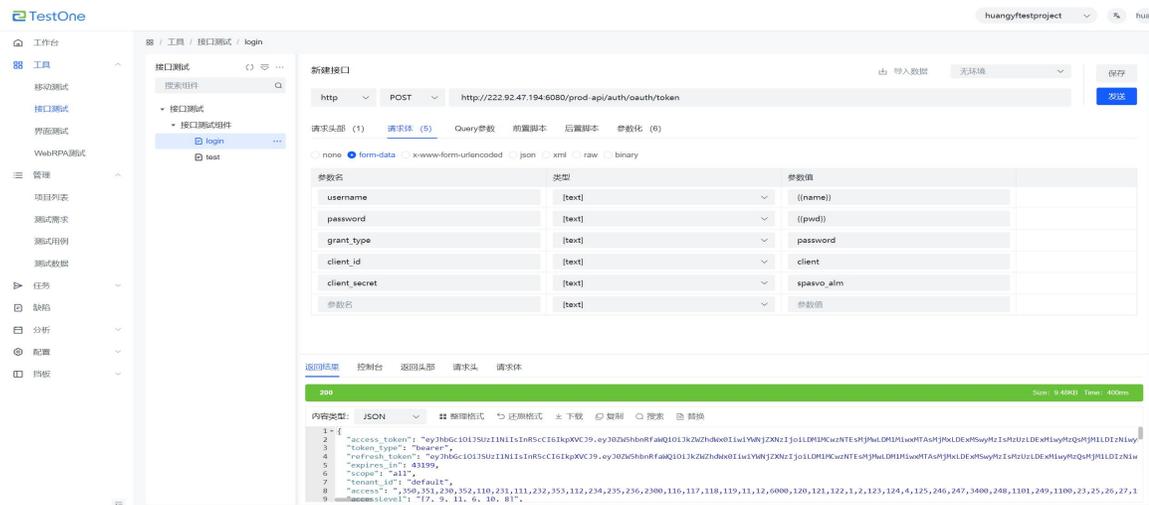


质量就是生命，效率决定未来

通过录制方式实现脚本的生成，以拖拽的形式进行调整脚本的逻辑、判断、分支；以便实现界面自动化的脚本设计。

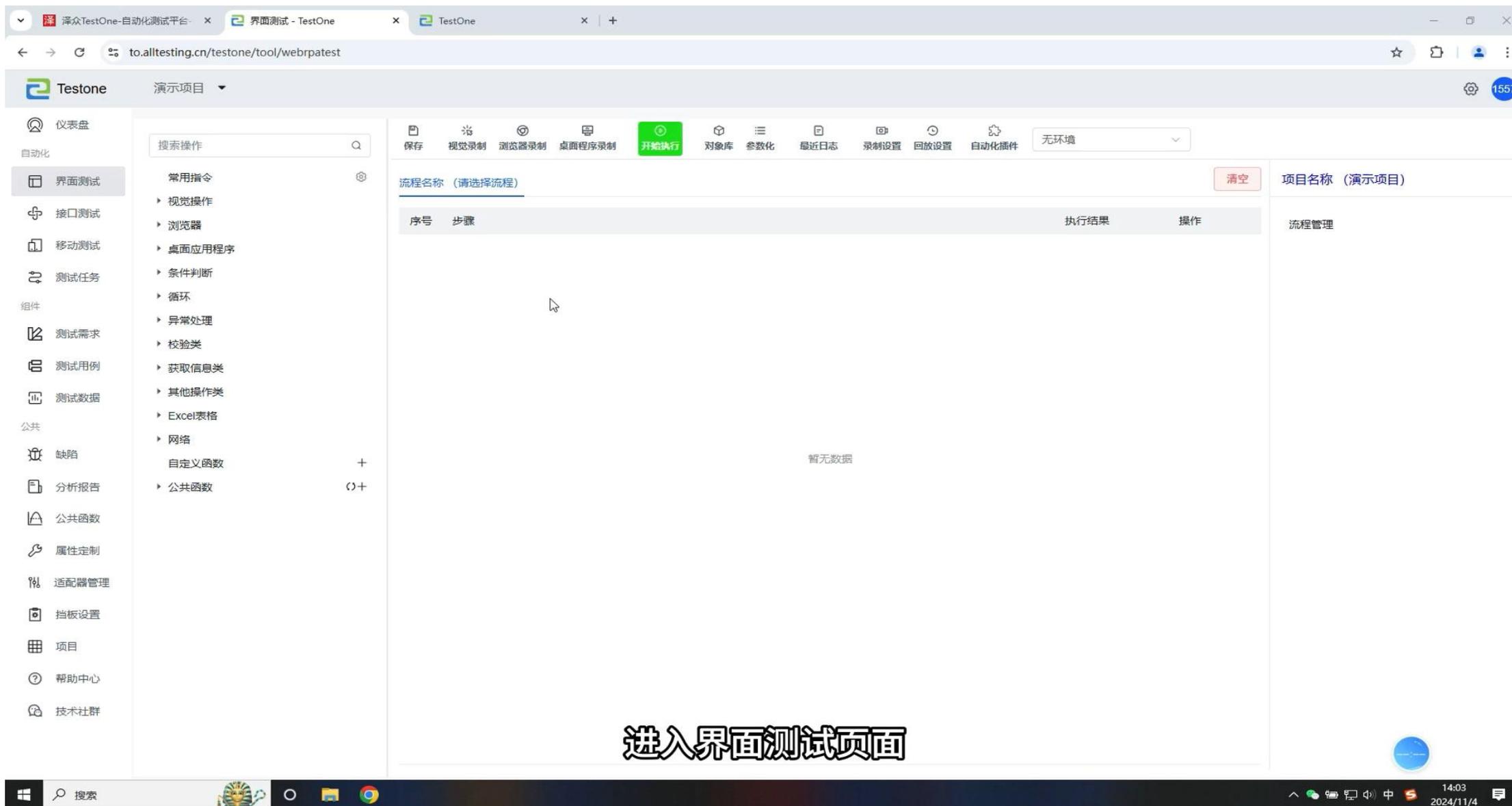


通过录制方式实现脚本的生成，以拖拽的形式进行脚本的逻辑与判断，以便实现移动端应用的脚本设计；



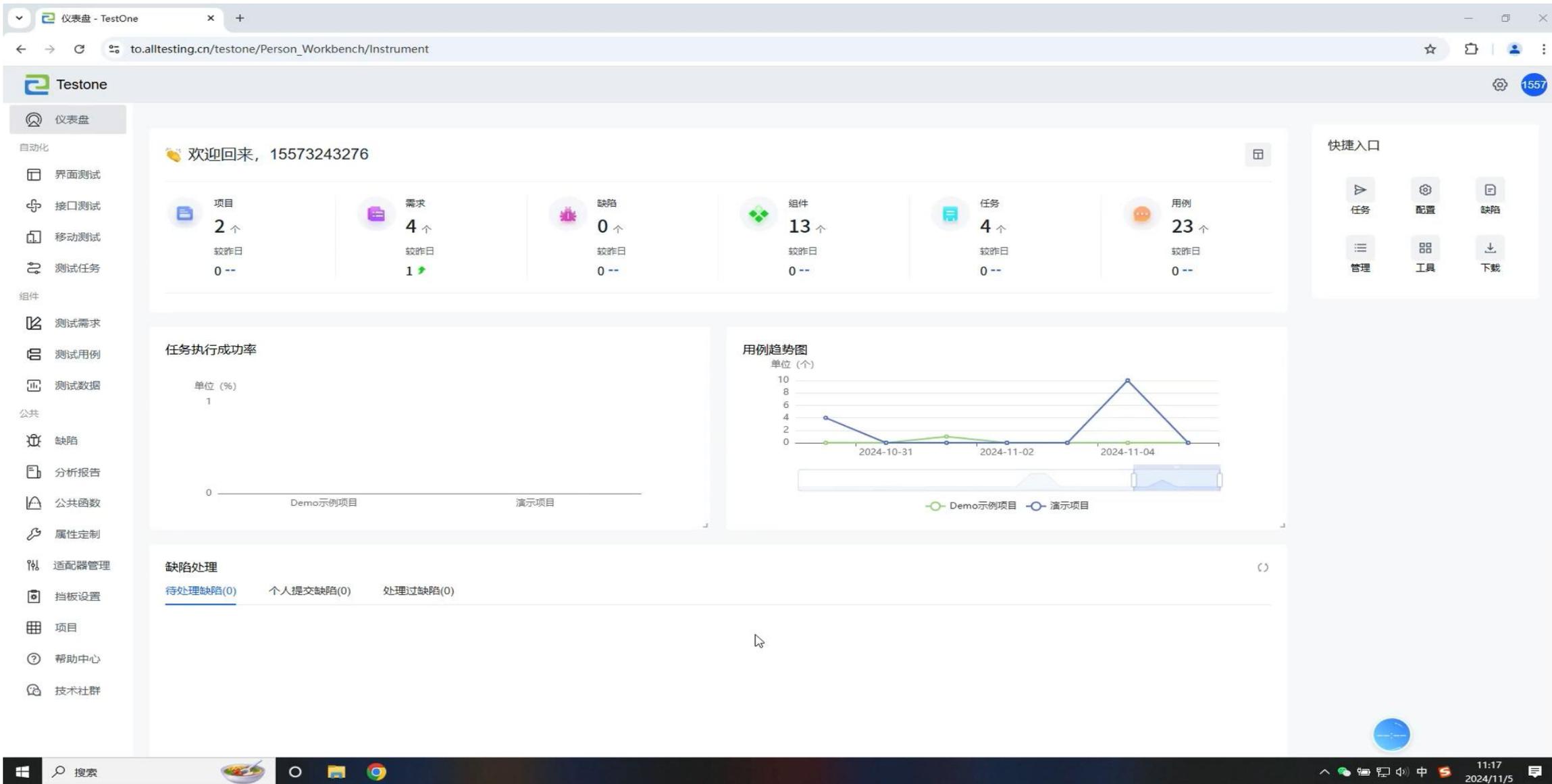
通过导入以及模板化编辑方式，模拟客户端向服务端发送请求，并验证接口的功能

界面自动化组件设计样例视频



The screenshot shows the Testone web interface. The browser address bar displays 'to.alltesting.cn/testone/tool/webpratest'. The interface includes a left sidebar with navigation options like '仪表盘', '自动化', '界面测试', '接口测试', '移动测试', '测试任务', '组件', '测试需求', '测试用例', '测试数据', '公共', '缺陷', '分析报告', '公共函数', '属性定制', '适配器管理', '挡板设置', '项目', '帮助中心', and '技术社群'. The main content area features a search bar, a toolbar with icons for '保存', '视觉录制', '浏览器录制', '桌面程序录制', '开始执行', '对象库', '参数化', '最近日志', '录制设置', '回放设置', '自动化插件', and a '无环境' dropdown. Below the toolbar is a '流程名称 (请选择流程)' field with a '清空' button. A table with columns '序号', '步骤', '执行结果', and '操作' is shown, containing the text '暂无数据'. On the right side, there is a '项目名称 (演示项目)' field and a '流程管理' section. A large white text overlay at the bottom center reads '进入界面测试页面'. The Windows taskbar at the bottom shows the search bar, task icons, and system tray with the time '14:03' and date '2024/11/4'.

移动自动化组件设计样例视频



The screenshot displays the Testone dashboard interface. At the top, there's a navigation bar with the Testone logo and a user profile icon labeled '1557'. Below this is a sidebar menu with categories like '自动化' (Automation) and '组件' (Components), listing various testing functions such as '界面测试' (UI Testing), '接口测试' (API Testing), and '测试用例' (Test Cases). The main content area features a welcome message '欢迎回来, 15573243276' and a row of six summary cards: '项目' (2 items), '需求' (4 items), '缺陷' (0 items), '组件' (13 items), '任务' (4 items), and '用例' (23 items). Below these are two charts: '任务执行成功率' (Task Execution Success Rate) and '用例趋势图' (Case Trend Chart). The '用例趋势图' shows data for 'Demo示例项目' and '演示项目' from 2024-10-31 to 2024-11-04. At the bottom, there's a '缺陷处理' (Defect Handling) section with tabs for '待处理缺陷(0)', '个人提交缺陷(0)', and '处理过缺陷(0)'. The Windows taskbar at the bottom shows the time as 11:17 on 2024/11/5.

接口自动化组件设计样例视频

The screenshot displays the Testone dashboard interface. At the top, there's a navigation bar with the Testone logo and a user profile icon labeled '1557'. Below this is a sidebar menu with categories like '自动化' (Automation), '组件' (Components), and '公共' (Public). The main content area features a welcome message '欢迎回来, 15573243276' and a row of six summary cards for '项目' (2), '需求' (3), '缺陷' (0), '组件' (11), '任务' (4), and '用例' (19). Below these are two charts: '任务执行成功率' (Task Execution Success Rate) and '用例趋势图' (Case Trend Chart). The '用例趋势图' shows data for 'Demo示例项目' and '演示项目' from 2024-10-30 to 2024-11-03. At the bottom, there's a '缺陷处理' (Defect Handling) section with counts for '待处理缺陷(0)', '个人提交缺陷(0)', and '处理过缺陷(0)'. The Windows taskbar at the bottom shows the time as 16:53 on 2024/11/4.

自动化测试组件的构成



界面自动化：业务逻辑脚本 + 参数化参数值 + 对象库对象

移动自动化：业务逻辑脚本 + 参数化参数值 + 对象库对象

接口自动化：接口实现方式 + 参数化参数值

参数化：通过数据驱动实现业务逻辑覆盖

运行环境：可定义公共参数作为全局变量

自定义函数：可自定义方法作为全局方法，可被所有脚本引用

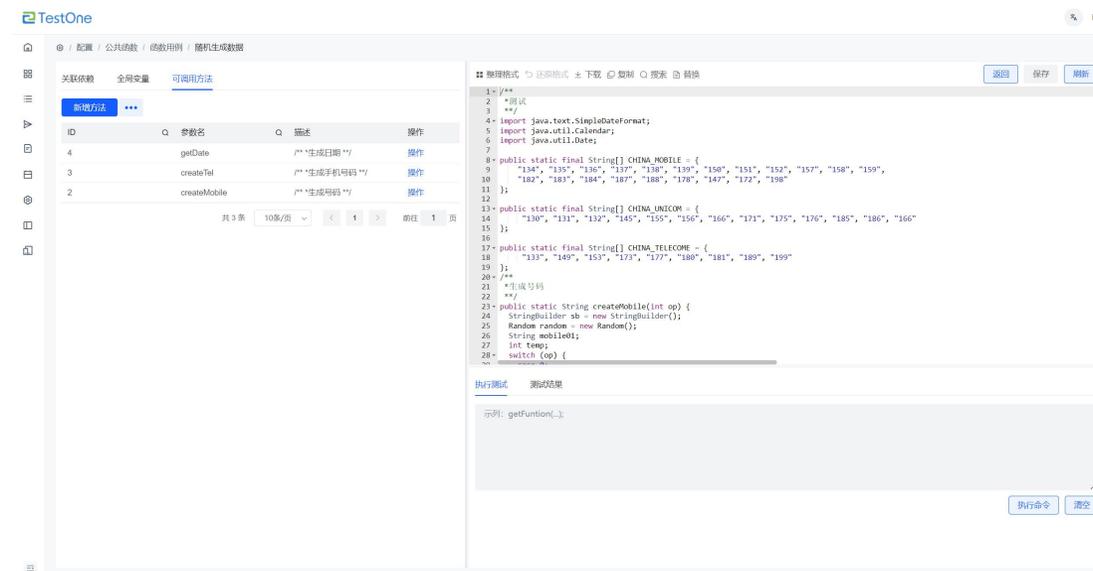
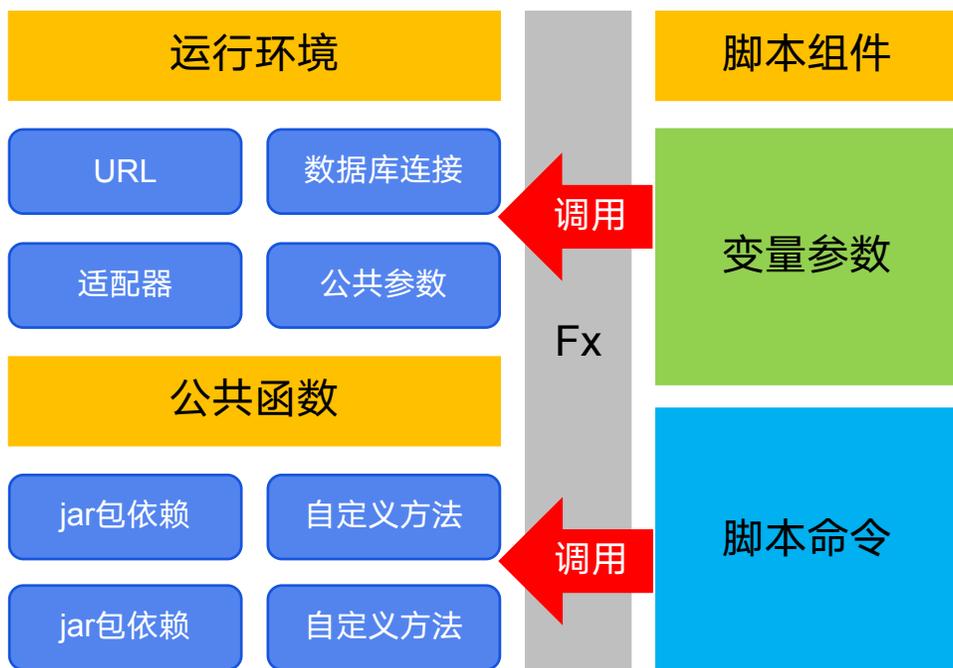
控件识别-对象识别与视觉识别

对象识别：将对象的各个属性捕获，在对象库生成一个个对象。回放根据对象属性定位对象。

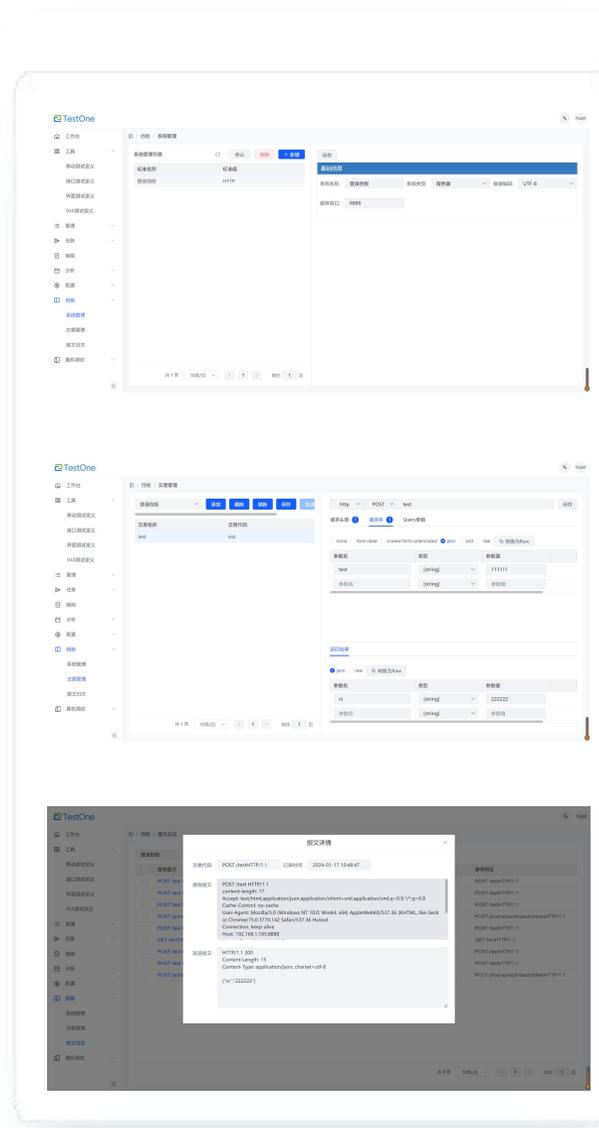
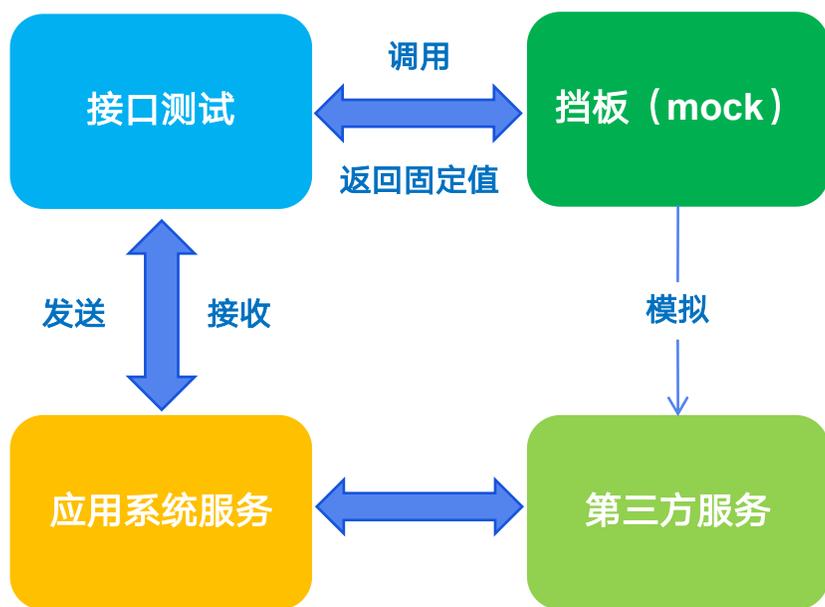
视觉识别：将页面文字与图片作为对象，回放时根据识别模型找到当前页面文字及图片所在位置。

	对象识别	视觉识别
优势	<ul style="list-style-type: none">➢ 录制直接生成对象操作命令；➢ 支持获取对象属性信息；➢ 回放执行效率较高；	<ul style="list-style-type: none">➢ 不受控件框架影响；➢ 当前页面控件上文字不变➢ 文字识别：无需录制对象，所见即所得；➢ 图像识别（移动端）：录制直接生成对象操作命令；
劣势	<ul style="list-style-type: none">• 移动端不支持webview对象录制；• PC端不支持非标准控件录制；• 页面变更频繁的系统，维护成本相对较高；	<ul style="list-style-type: none">• 需要基于较高的识别模型；• 相对对象识别，回放速度较慢；• 图像识别录制直接生成的图片对象需修改对象名称（默认时间戳），不利于脚本业务展示；• 获取页面信息相对对象识别操作更复杂。

运行环境管理与公共函数管理



挡板(Mock)



配置系统

- 1) 配置系统类型：服务器or客户端
- 2) 配置数据编码
- 3) 配置系统端口

配置交易

- 1) 配置交易名称、代码；
- 2) 配置交易类型：接受or发送
- 3) 配置该交易的请求方式、API、请求头、请求体；
- 4) 配置该交易的返回结果

日志查看

在报文日志中，可查看该系统交易调用的报文详情信息

自动化脚本维护

针对单个脚本

1) 对象库对象属性维护

----对象重新录制or修改对象库对象属性

2) 脚本业务逻辑维护

----重新录制脚本流程or拖拽脚本语句调整执行顺序

3) 脚本参数数据维护

----参数化维护

针对项目框架

1) 公共组件脚本维护

----通过公共组件变动实现业务组件批量修改

2) 公共环境参数维护

----公共参数维护在公共环境中，批量修改环境参数

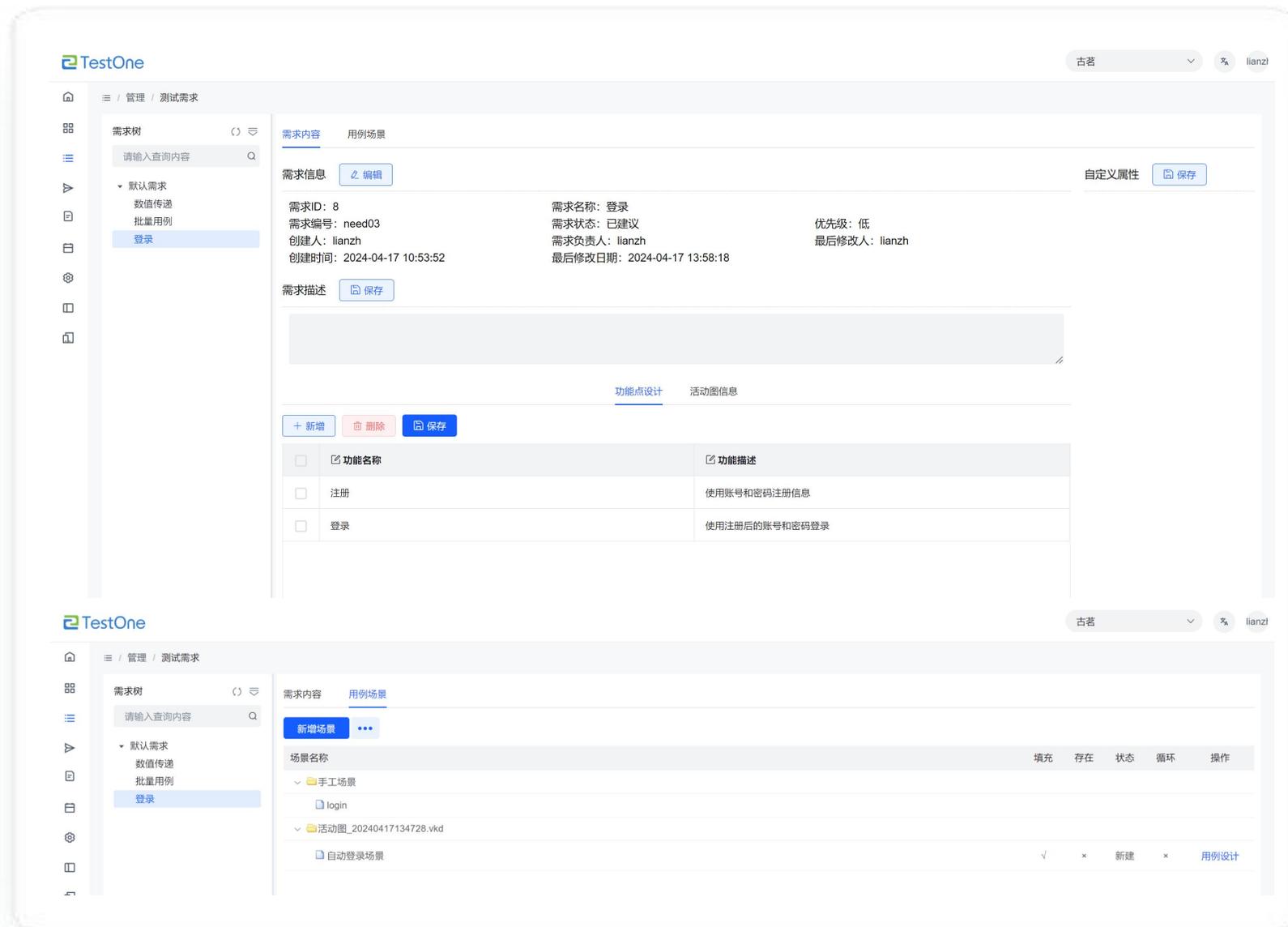
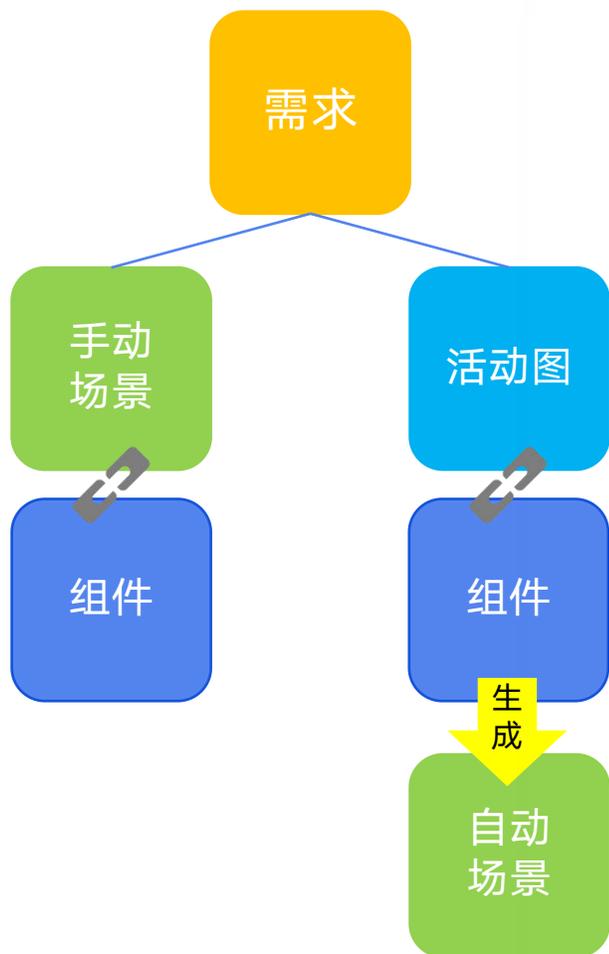
3) 公共函数维护

----常用方法定义公共函数，修改函数改变方法

目录

- 01 自动化测试脚本设计
- 02 自动化测试用例设计
- 03 自动化测试执行管理
- 04 自动化测试项目管理

需求-场景管理



自动化测试用例的构成



测试用例：**组件+数据+校验点**

测试用例特点：

- 1) 支持界面、接口、移动组件”单独”或”混合”执行；
- 2) 支持组件之间数值传递；
- 3) 支持用例数据引用数据池中数据，实现用例数据一对多关系绑定；
- 4) 支持设置出参参数；
- 5) 支持设置校验点参数及校验点参数预期值；

自动化测试用例的数据来源

固定数据

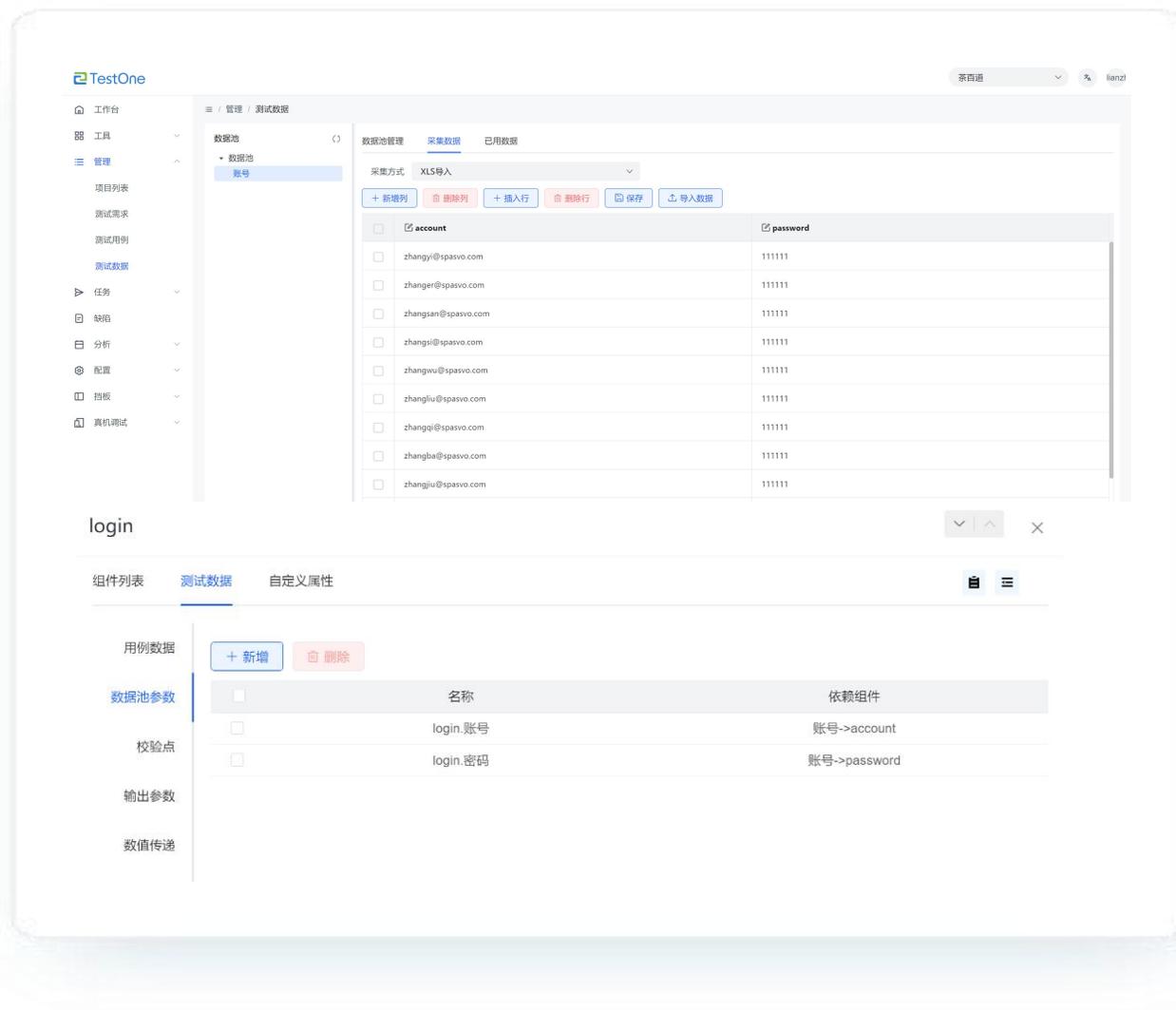
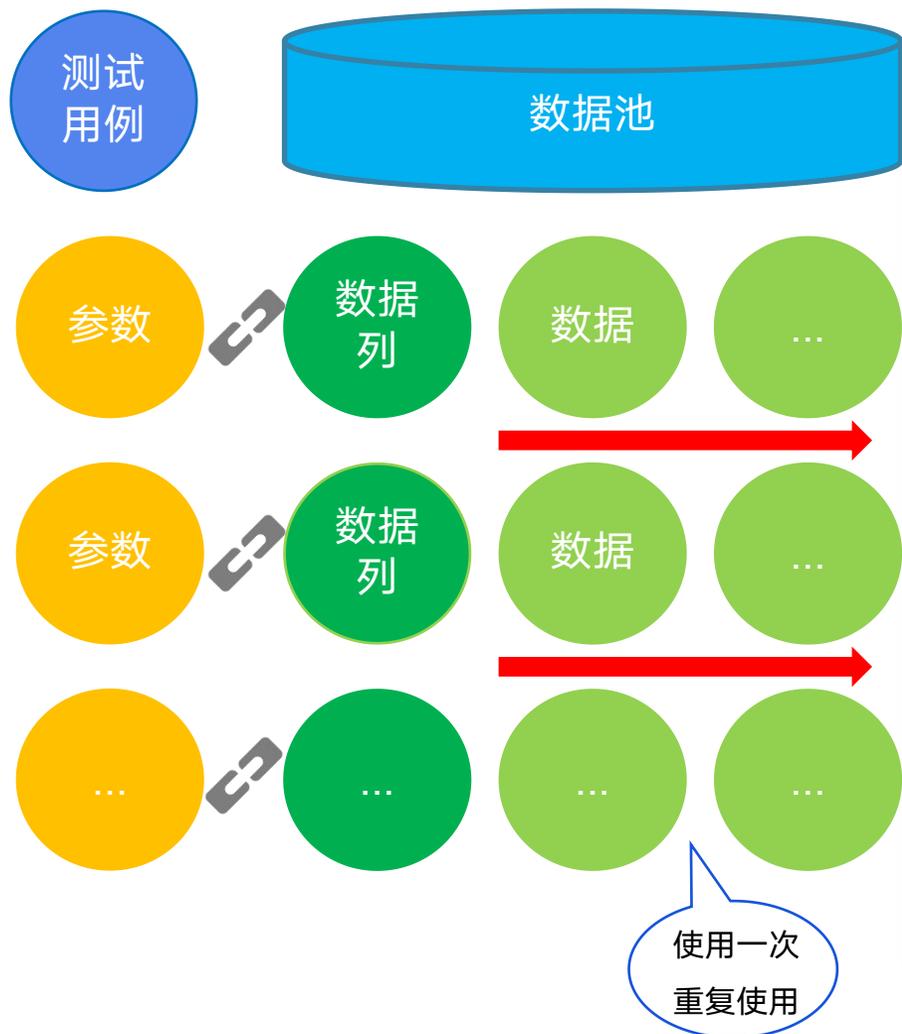
- 1) 固定值
- 2) 数据池

动态数据

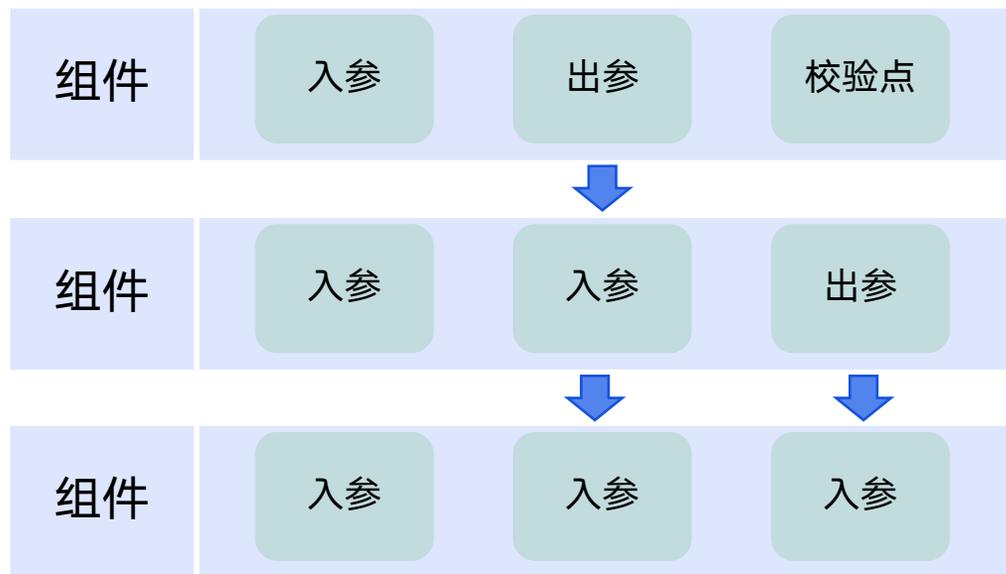
- 1) SQL查询
- 2) 数据传递
- 3) 自动生成
- 4) 读写文件

数据来源	描述
固定值	测试用例固定的一组数据
数据池	通过数据库查询、Excel导入生成的数据集合
SQL查询	在组件业务操作前，通过SQL语句从数据库查询到的数据作为执行数据
数值传递	多组件串联执行时，组件之间数据的传递
自动生成	通过代码或者工具生成的数据
读写文件	从文件（例如：excel、txt等）中获取的数据

数据池



数值传递与校验点



新增值传递

search	WEB	数据源	目标数据源
现可用	已选中	<input type="checkbox"/> login_1.账号	login_2.账号
message	参数	<input type="checkbox"/> login_1.密码	login_2.密码
		<input type="checkbox"/> login_2.账号	search.message
		<input type="checkbox"/> search.message	WEB.参数

操作按钮: 匹配, 追加, 删除

数据池参数 输出参数 校验点 **数值传递**

刷新 + 新增 删除 参数自动匹配

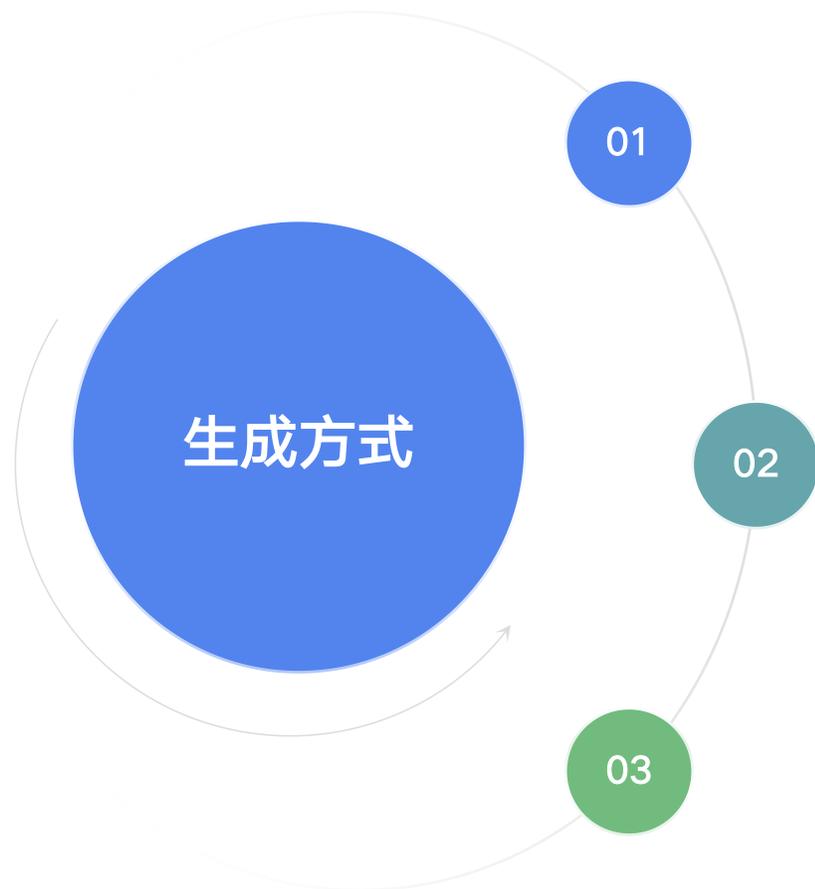
名称	数据源
<input type="checkbox"/> login_1.账号	login_2.账号
<input type="checkbox"/> login_1.密码	login_2.密码
<input type="checkbox"/> login_2.账号	search.message
<input type="checkbox"/> search.message	WEB.参数

数据池参数 输出参数 **校验点** 数值传递

刷新 + 新增 保存 删除

数值栏位	预期值
<input type="checkbox"/> login_1.账号	lianzh@spasvo.com

平台自动化用例生成方式



用例模块新增测试用例

在“测试用例模块”，直接绑定测试组件，设计测试数据进行新增测试用例。

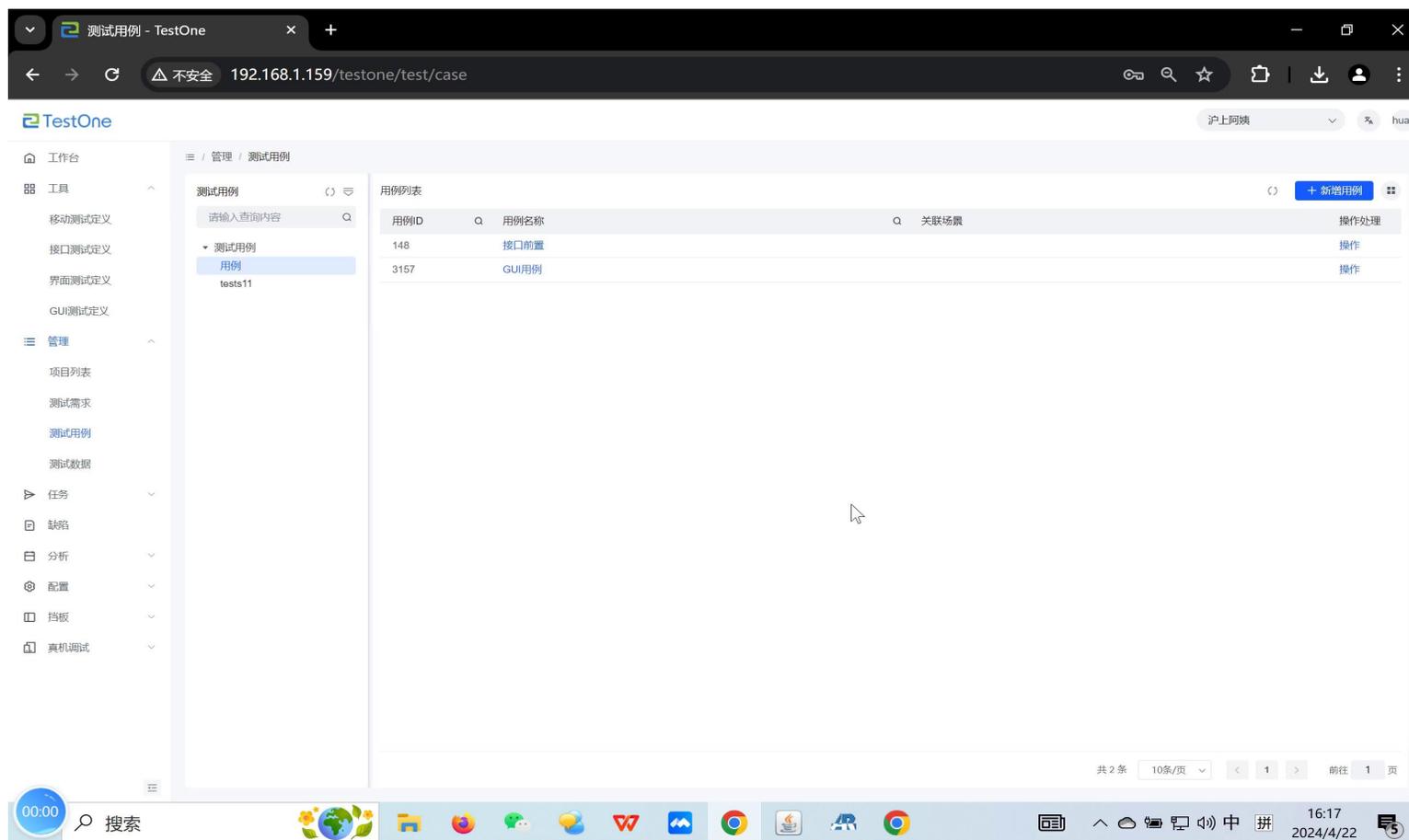
手工场景批量导入测试用例

通过在需求下手动创建测试场景后，在场景下绑定测试组件，设计用例数据来源，在线批量添加或Excel导入测试用例数据后，批量生成测试用例。

TCD自动生成测试用例

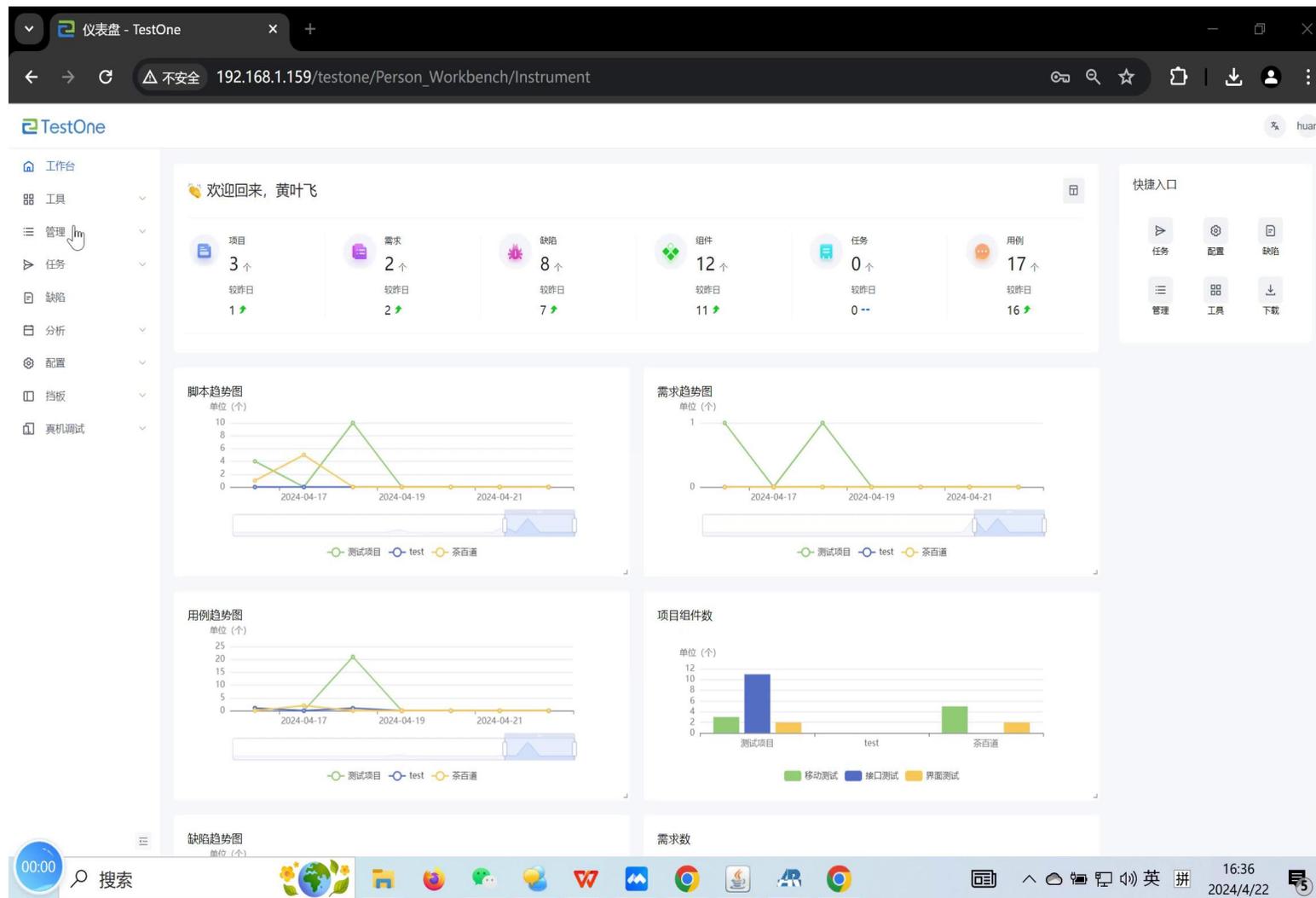
通过活动图分析需求后，自动生成测试场景并绑定组件，对组件的参数进行参数设计及业务规则的添加，最后基于数据项正交自动生成测试用例。

单用例添加执行

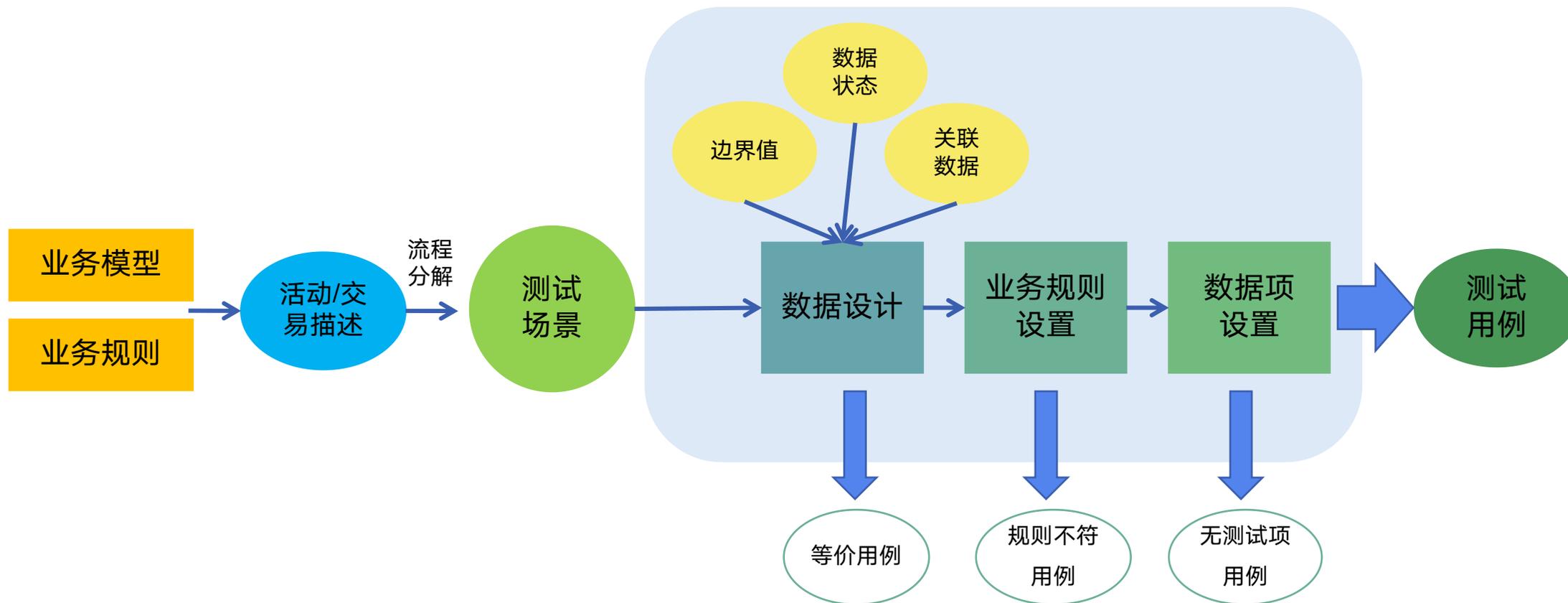


用例组件添加及数据设计

需求场景批量导入用例



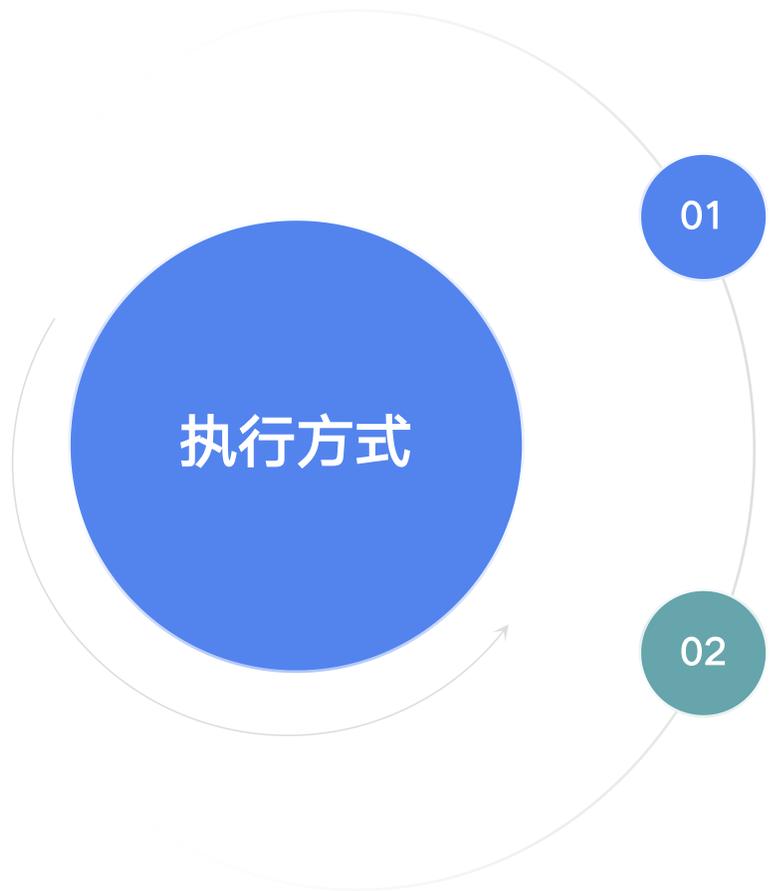
TCD自动生成测试用例



目录

- 01 自动化测试脚本设计
- 02 自动化测试用例设计
- 03 自动化测试执行管理
- 04 自动化测试项目管理

测试执行方式



项目内测试集执行

测试集是测试用例的集合，在项目内可以对测试集指定执行器、运行环境、执行方式（定时、预约、立即执行），生成运行明细与日志。

跨项目测试任务执行

测试任务是测试集的集合，在多个项目存在待执行测试集时，可生成跨项目测试任务，指定执行器、运行环境、执行方式（定时、预约、立即执行）的同时，支持设置邮件通知，发送执行结果。

测试集与任务执行

任务执行

测试集执行



执行方式:

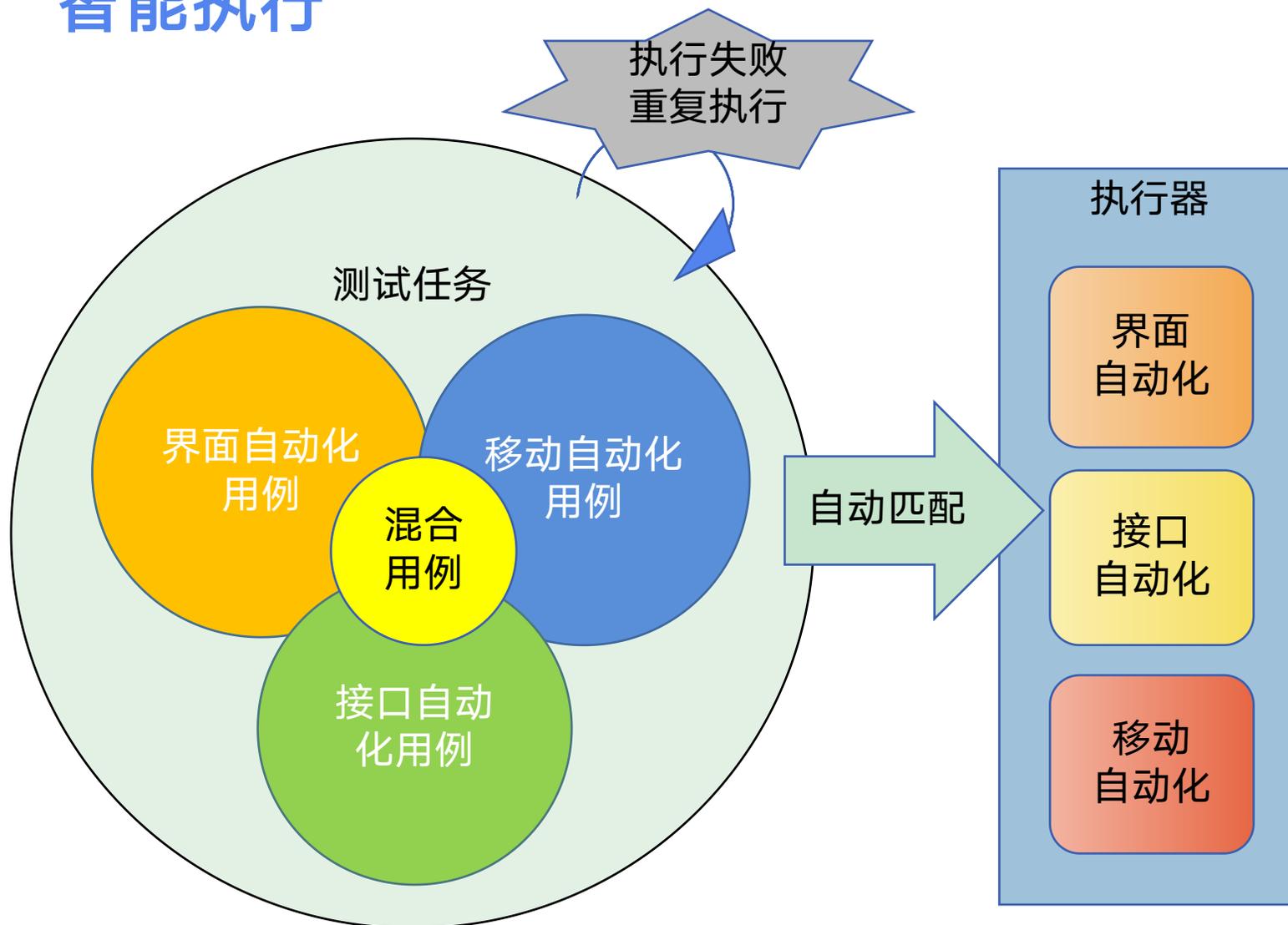
- 1) 定时: 每天指定时间执行
- 2) 预约: 指定时间执行一次
- 3) 立即执行: 马上执行

运行环境: 当前测试用例执行的运行环境, 可调用运行环境中的公共参数及接口环境。

执行次数: 用例执行失败时, 再次发起执行, 至该用例执行成功or执行次数达到设置的次数。

邮件通知: 通过邮箱服务器将执行结果发送至邮件订阅的用户邮箱。

智能执行



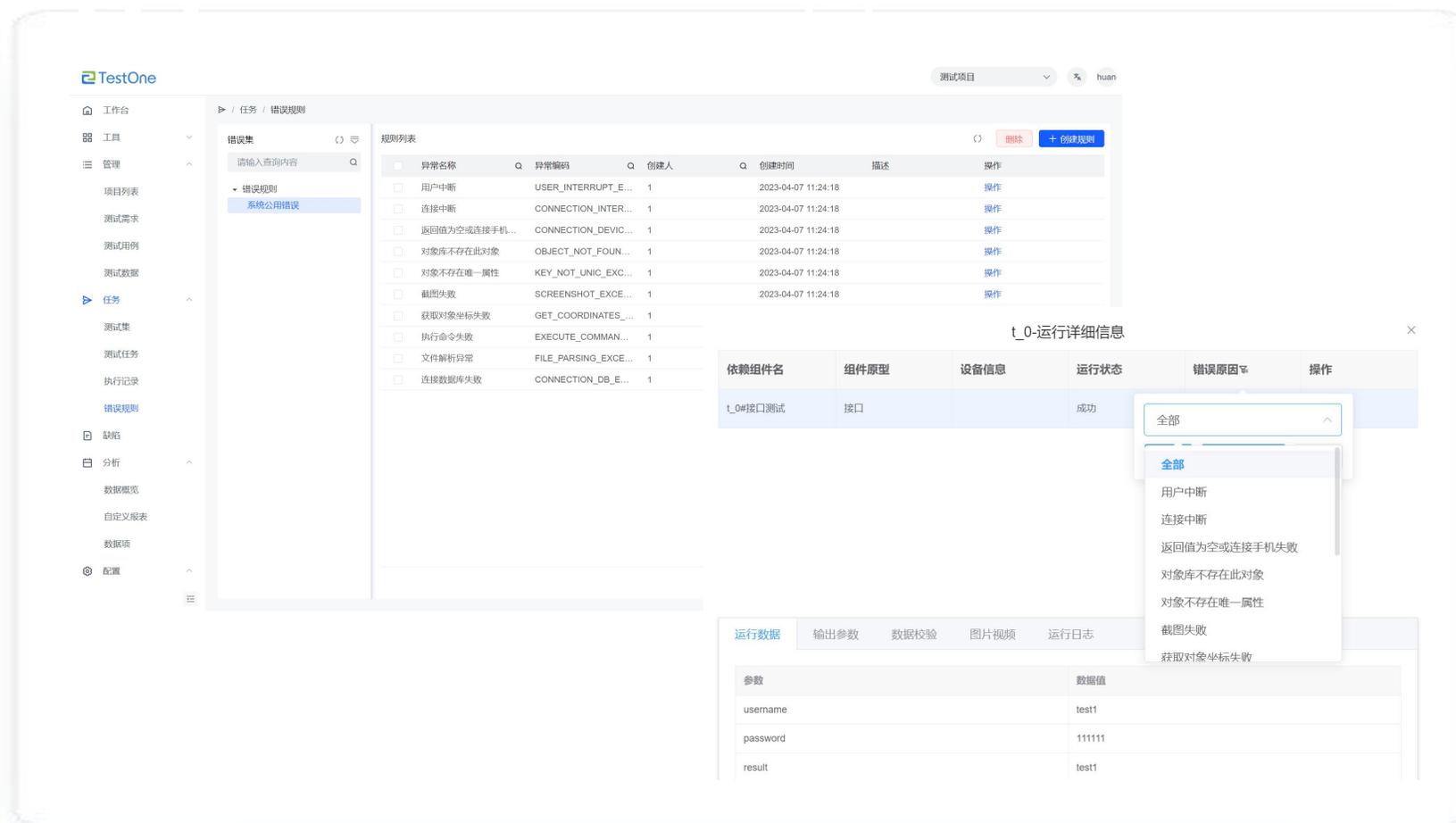
智能执行：

1) 单个测试任务支持添加多类型的测试用例。执行时，自动识别用例绑定组件的类型，智能选择执行器的执行方式，进行自动化执行。

2) 支持定义执行失败时用例重复执行次数

错误日志分类

执行完成，执行日志中对执行失败的用例自动判别错误类型



错误类型：

- 用户中断
- 连接中断
- 返回值为空或连接手机失败
- 对象库不存在此对象
- 对象不存在唯一属性
- 截图失败
- 获取对象坐标失败
- 执行命令失败
- 文件解析异常
- 连接数据库失败
- 对象无分辨率
- 对象图片为空
- 文件传输异常
- 对象类型错误

执行明细与日志

自动运行明细

基本信息

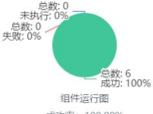
运行名称	1708768539833
运行人	huangyf
开始时间	2024-02-24 17:55:39
结束时间	2024-02-24 17:56:45
运行总数	2
成功数	2
失败数	0
未执行数	0
成功率	100.00%

用例运行图



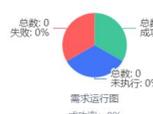
成功率: 100.00%

组件运行图



成功率: 100.00%

需求运行图



成功率: 0%

需求运行情况

类别	运行总数	成功数	失败数	未执行数	成功率
用例	2	2	0	0	100.00%
组件	6	6	0	0	100.00%
需求	0	0	0	0	0%

需求运行明细

设备	用例总数	成功数	失败数	未执行数	成功率
47S0220A10028399	2	2	0	0	100.00%

[导出PDF](#) [导出Word](#)

t_1-运行详细信息

依赖组件名	组件原型	设备信息	运行状态	错误原因	Q	操作
t_1#1	登录	47S0220A10028...	成功			组件场景
t_1#2	登录	47S0220A10028...	成功			组件场景
t_1#3	登录	47S0220A10028...	成功			组件场景

运行数据 输出参数 数据校验 图片视频 **运行日志**

```

mr.touch("TextView|其他邮箱")--Succeeded
mr.touch("CheckBox|checkbox")--Succeeded
mr.touch("TextView|同意")--Succeeded
mr.setValue("EditText|example@company.com","huangyf@spasvo.com")--Succeeded
mr.setValue("EditText|邮箱密码","111111")--Succeeded
mr.touch("TextView|下一步")--Succeeded
mr.touch("Button|确定")--Succeeded
mr.touchKey("BACK")--Succeeded
47S0220A10028399:执行成功;
                    
```

1. 支持查看项目运行明细，包含运行基本信息，自动运行状况，需求运行概况，需求运行明细、错误用例等；
2. 支持将执行明细导出为PDF、Word压缩包文件；
3. 支持查看执行日志，包括运行数据、输出参数、数据校验、图片视频、运行日志；

01.

生成原始覆盖率报告

包含代码行覆盖率、方法覆盖率、分支覆盖率等基本信息。

分析原始报告，初步了解项目的覆盖率情况，找出覆盖率较低的模块和代码区域，为后续优化提供方向。

```
--: 0:Source:example.c
--: 0:Graph:example.gcno
--: 0:Data:example.gda
--: 0:Runs:1
--: 0:Programs:1
--: 1:#include <stdio.h>
--: 2:
2: 3:void foo(int x) {
2: 4:   if (x > 0) {
1: 5:     printf("Positive\n");
--: 6:   } else {
1: 7:     printf("Non-positive\n");
--: 8:   }
2: 9:}
--: 10:
1: 11:int main() {
1: 12:   foo(1);
1: 13:   foo(-1);
1: 14:   return 0;
--: 15:}
```

```
/*! CONFIGURE_TASKS_LIST */
/*-----*/
31: BaseType_t vTaskResumeAll(void)
31: {
31:   BaseType_t uxTCB = NULL;
31:   BaseType_t uxSchedulerSuspended = pdFALSE;
31:   /* If uxSchedulerSuspended is zero then this function does not match a
31:   * previous call to vTaskSuspendAll(). */
31:   configASSERT(uxSchedulerSuspended == 0);
31:   /* It is possible that an ISR caused a task to be removed from an event
31:   * list while the scheduler was suspended.  If this was the case then the
31:   * removed task will have been added to the uxPendingReadyList.  Once the
31:   * scheduler has been resumed it is safe to move all the pending ready
31:   * tasks from this list into their appropriate ready list. */
31:   taskENTER_CRITICAL();
31:   ..uxSchedulerSuspended;
31:   if (uxSchedulerSuspended == (UBaseType_t)pdFALSE)
31:     if (uxCurrentNumberTasks > (UBaseType_t)0)
31:     {
31:       /* Move any ready tasks from the pending list into the
31:       * appropriate ready list. */
31:       while (listLIST_IS_EMPTY(&uxPendingReadyList) == pdFALSE)
31:       {
31:         /* Get the task object from the pending ready list. */
31:         tsk_t *pxTCB = (tsk_t *)listGET_OWNER_OF_NEXT_ITEM(&uxPendingReadyList);
31:         /* Remove it from the pending list. */
31:         listREMOVE_ITEM(&uxPendingReadyList);
31:         /* Place it in the appropriate ready list. */
31:         listINSERT_ITEM(pxTCB, pxTCB->uxPriority);
31:         /* If the moved task has a priority higher than or equal to
31:         * the current task then a yield must be performed. */
31:         if (pxTCB->uxPriority >= uxCurrentPriority)
31:         {
31:           vTaskYield();
31:         }
31:       }
31:     }
31:   /* A task was unblocked while the scheduler was suspended,
31:   * so we must check to see if we need to reschedule. */
31:   if (uxTCB != NULL)
31:     /* A task was unblocked while the scheduler was suspended,
31:     * so we must check to see if we need to reschedule. */
31:     vTaskSchedule();
31:   uxSchedulerSuspended = pdFALSE;
31:   return (UBaseType_t)0;
31: }
```

02.

合并多测试用例覆盖率

当项目有多个测试用例时，将不同测试用例生成的覆盖率数据进行合并，得到更全面的覆盖率报告，反映整体测试情况。

检查合并后的报告是否准确反映各测试用例的覆盖率贡献，避免数据冲突或遗漏，确保报告的准确性。

Current view: top level - FreeRTOS/FreeRTOS-Kernel		Hit	Total	Coverage
Test: freertos_simulator.info	Lines: 585	2250	26.0 %	
Date: 2021-09-13 00:20:40	Functions: 56	191	29.3 %	
	Branches: 191	1037	18.4 %	

Filename	Line Coverage	Functions	Branches
event_groups.c	0.0 % 0 / 172	0.0 % 0 / 15	0.0 % 0 / 50
list.c	75.6 % 31 / 41	80.0 % 4 / 5	83.3 % 5 / 6
queue.c	30.8 % 184 / 597	32.6 % 15 / 46	21.2 % 64 / 302
stream_buffer.c	0.0 % 0 / 277	0.0 % 0 / 25	0.0 % 0 / 116
tasks.c	27.1 % 250 / 921	33.3 % 24 / 72	18.3 % 87 / 476
timers.c	49.6 % 120 / 242	46.4 % 13 / 28	45.5 % 35 / 77

03.

生成HTML可视化报告

将合并后的覆盖率数据转换为HTML格式的可视化报告，方便团队成员直观看代码覆盖率情况，快速定位未覆盖代码。

在报告中添加交互功能，如点击代码行查看未覆盖原因，提升报告的易用性和实用性，便于团队进行代码审查和优化。

精准测试分析报告

项目概况

项目名称 xxx服务
构建版本 v1.2.3
测试周期 2025年02月01日 - 2025年02月15日

覆盖率摘要

指标	覆盖率	行业基准
行覆盖率	78%	80%
分支覆盖率	72%	75%

覆盖率热点图

- 未覆盖模块：订单模块 (风险等级：高)
- 未覆盖模块：用户认证模块 (风险等级：中)
- 未覆盖模块：日志模块 (风险等级：低)

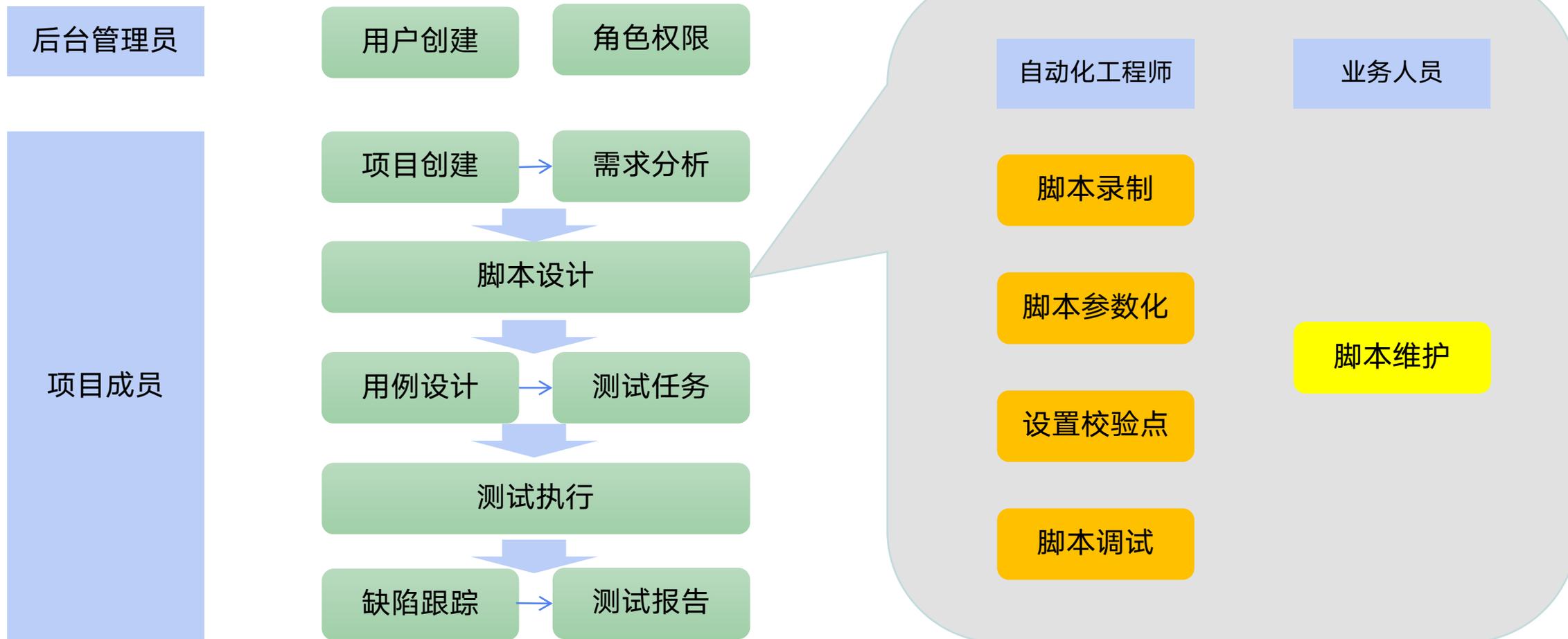
优化建议

- 建议补充5个边界用例
- 高风险模块订单模块需进行代码审查
- 优化3个低效测试用例

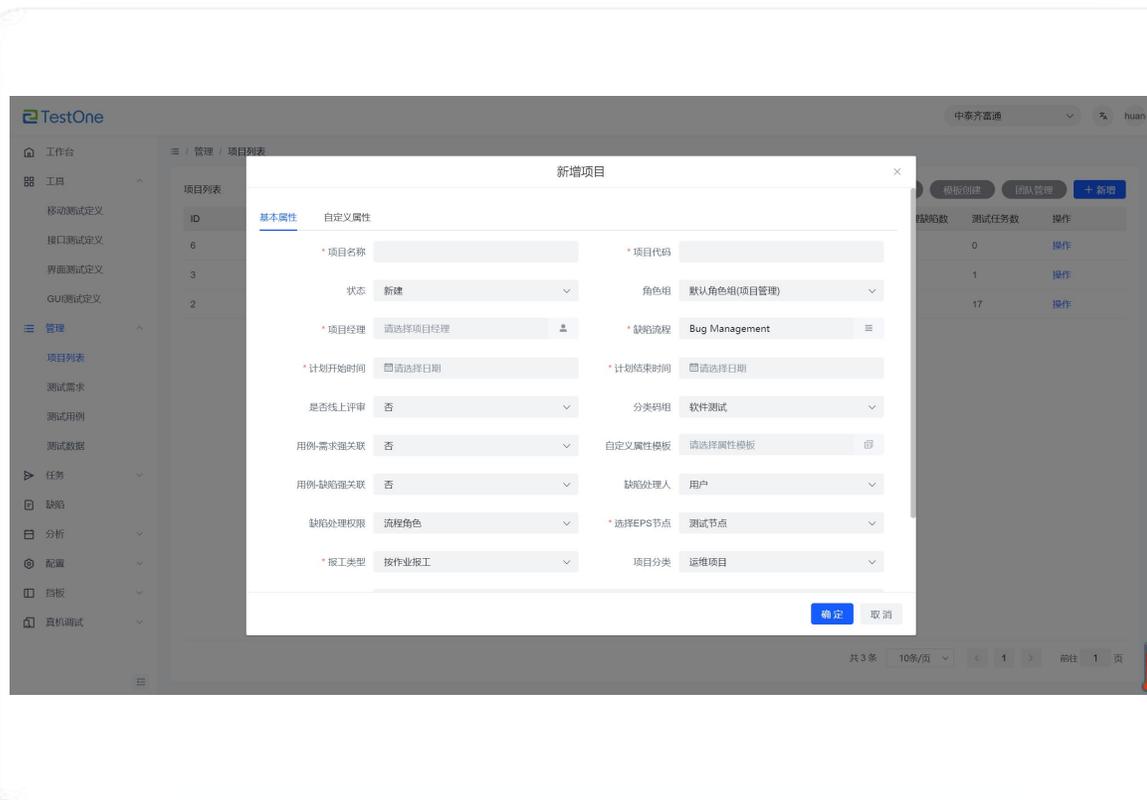
目录

- 01 自动化测试脚本设计
- 02 自动化测试用例设计
- 03 自动化测试执行管理
- 04 自动化测试项目管理

测试过程管理



项目团队管理



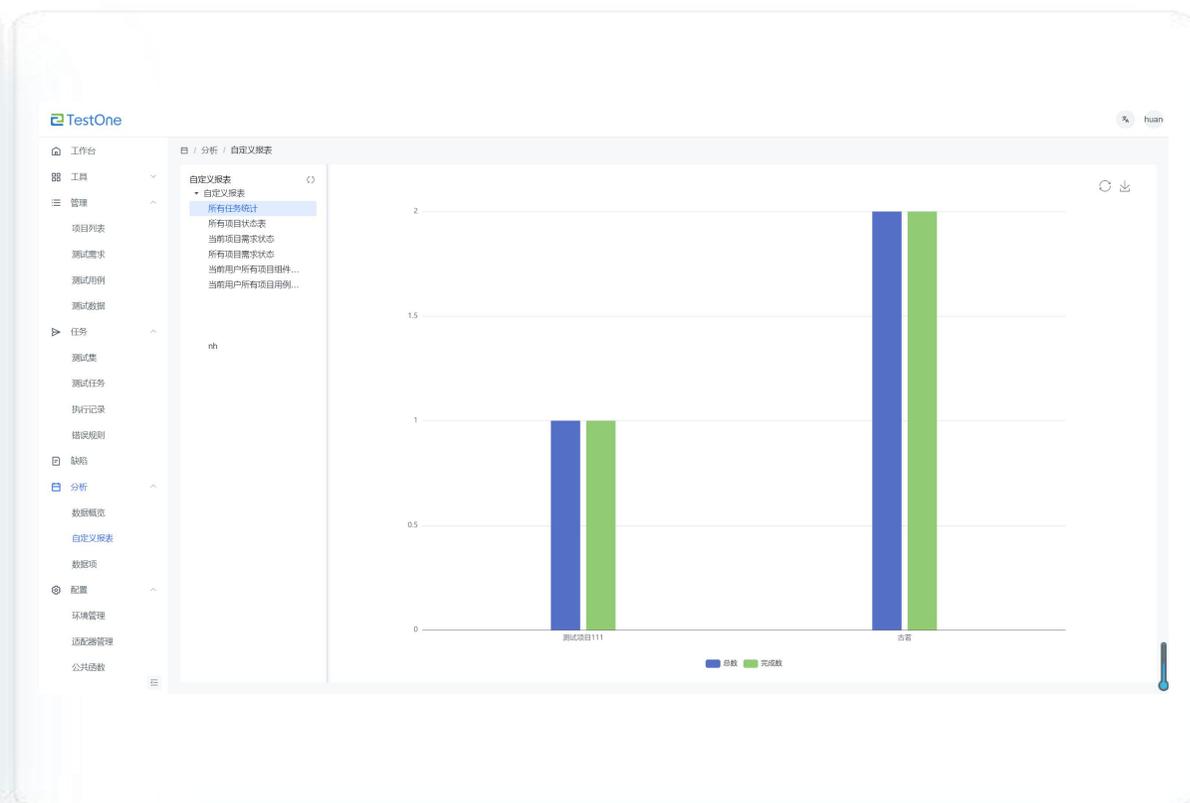
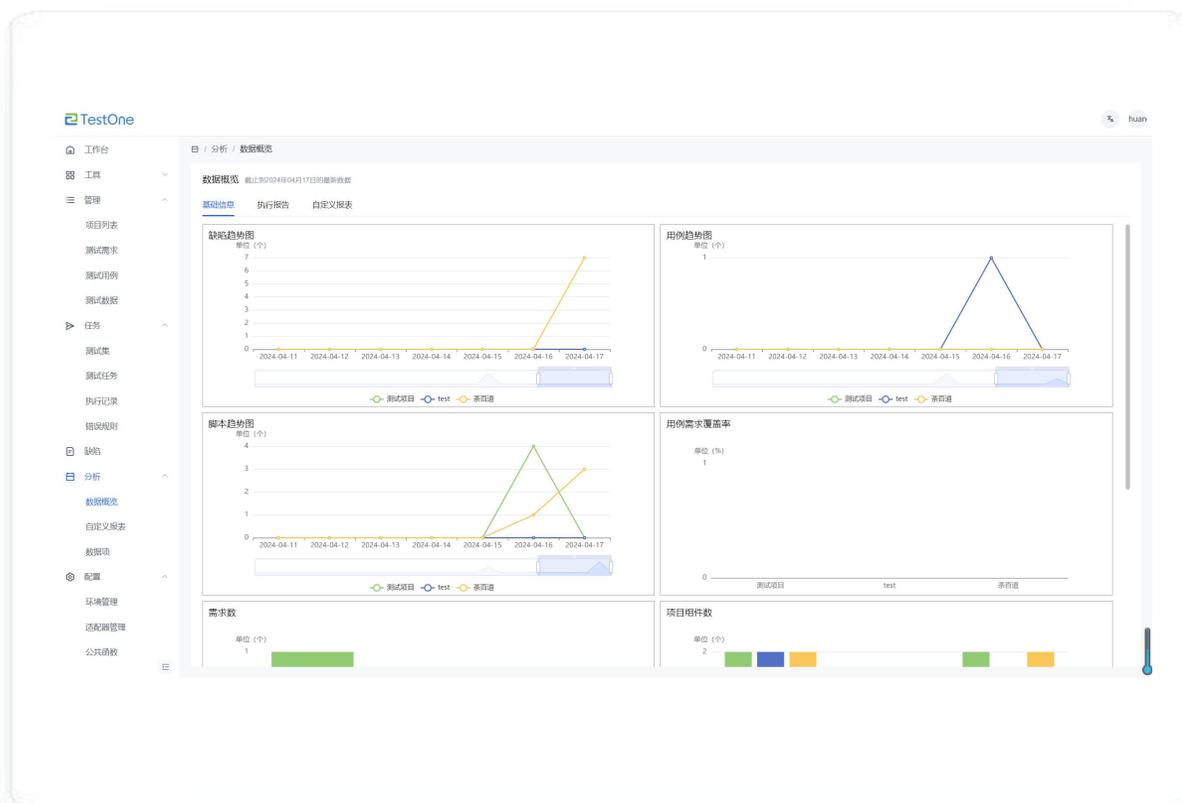
创建项目时指定项目经理、缺陷处理流程。
支持以其他项目为模板，创建新项目。



项目经理对后台人员赋予不同的角色，给予在此项目中的不同权限。

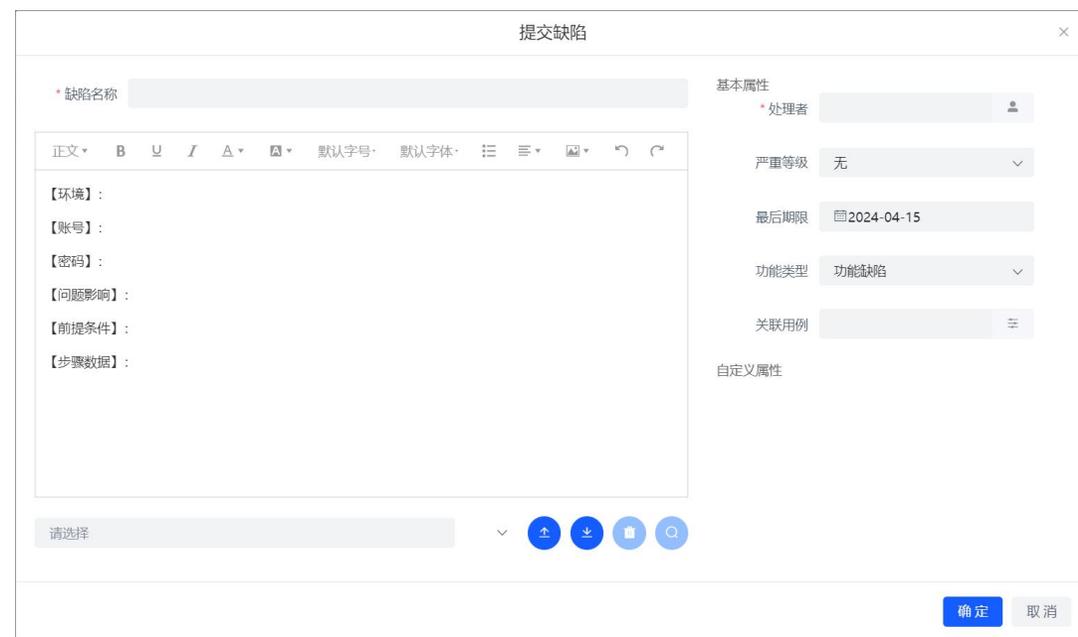
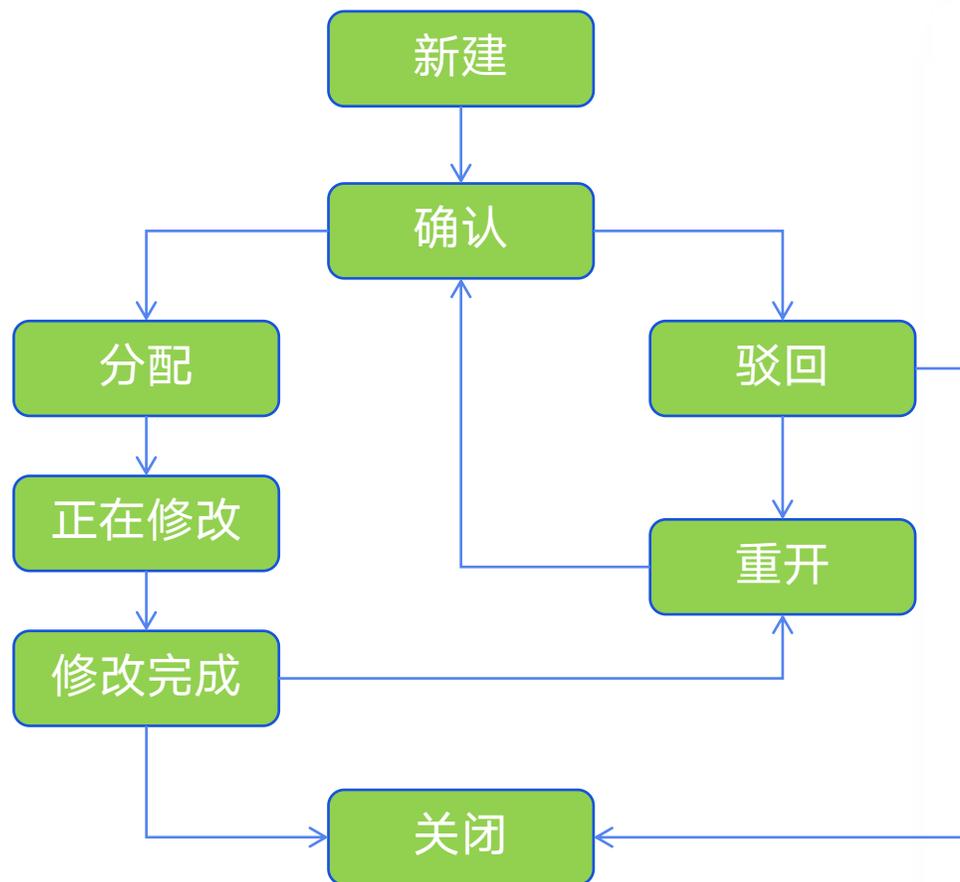
支持通过邮件订阅功能，将项目中任务的运行结果发送至对应用户邮箱。

项目运行过程监控



支持查看当前用户参与项目的需求、用例、脚本、缺陷每日新增趋势图，需求个数，用例需求覆盖率及项目组件个数；
支持用户通过数据库查询出数据作为数据项，自定义测试报表；
报表支持在工作台展示。

缺陷流程管理



提交缺陷

* 缺陷名称

基本属性

- * 处理者
- 严重等级 无
- 最后期限 2024-04-15
- 功能类型 功能缺陷
- 关联用例

自定义属性

正文

【环境】：
【账号】：
【密码】：
【问题影响】：
【前提条件】：
【步骤数据】：

请选择

确定 取消

目录

01

案例分享

用例覆盖率

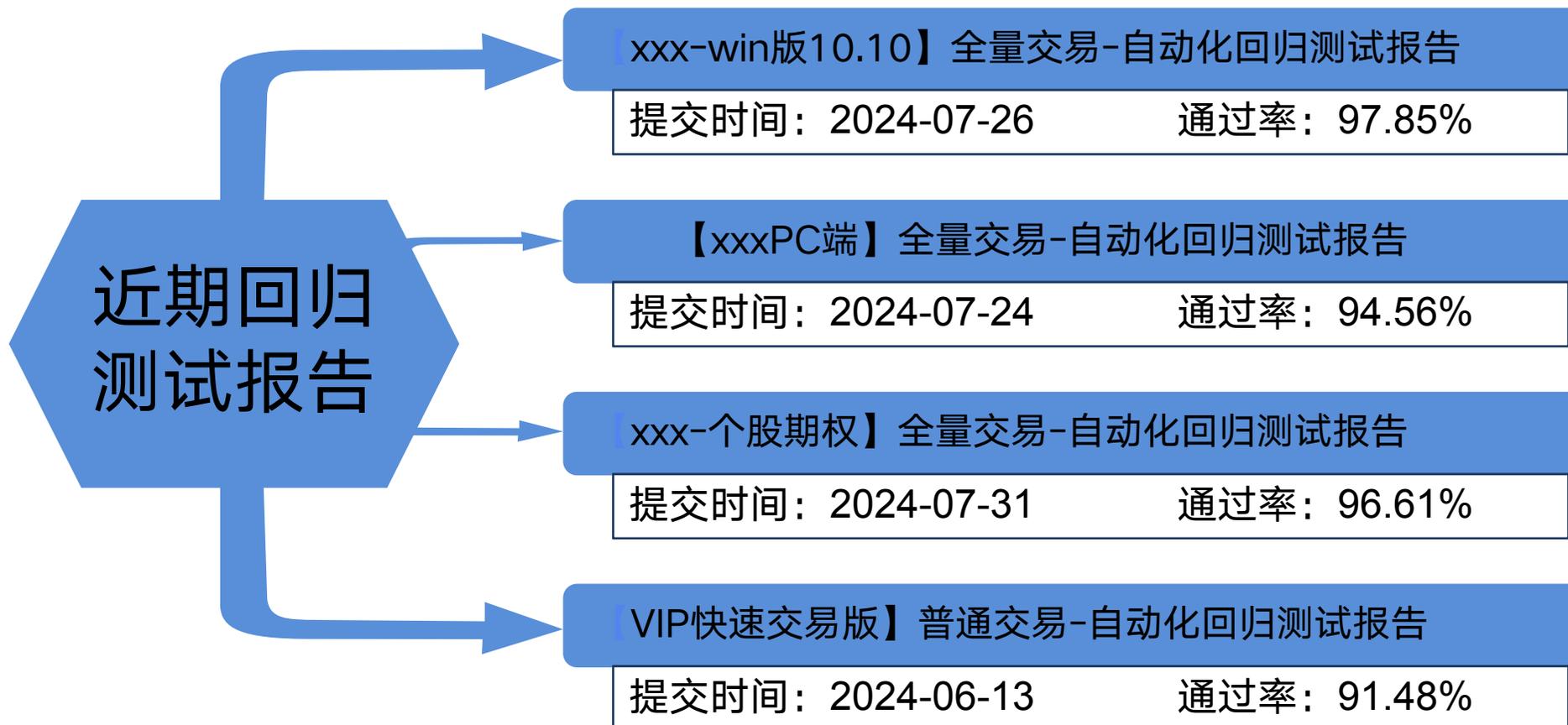


质量就是生命，效率决定未来

列出了xxx手工案例数量与实现自动化案例数量对比，从表格可以看出：目前，xxx版整体自动化案例数量为728条，整体手工案例数为776条，自动化覆盖率约为93.81%。

模块	手工案例数	自动化案例数	自动化比例
集中交易	190	190	100.00%
融资融券	90	90	100.00%
港股通	190	190	100.00%
适当性管理	36	24	66.67%
科创板	128	92	71.88%
创业版	60	60	100.00%
北交所	82	82	100.00%
合计	776	728	93.81%

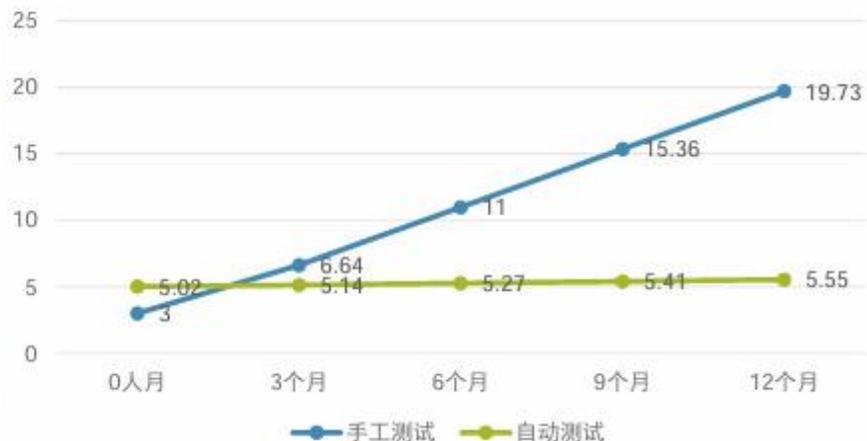
整体自动化测试覆盖率已经达到90%以上，自动化测试通过率在95%，后续新增案例纳入回归范围，通过率也在90%左右，实现了跨电商客户端，跨柜台的自动化测试的统一解决方案。



以100个交易为例

对比项	需求分析	脚本设计	用例设计 (800用例)	回归测试	总计
人工测试	25人天	0	25人天	16人天 (1天50个)	66
自动测试	50人天	50人天	10人天	0.5人天	110.5
结论	1, 自动化测试在第一次的投入比较大, 主要是人力和时间; 2, 在后期自动化回归测试的执行效率远远高于手工测试; 3, 第一次完成: 人工投入66天, 3人月; 自动投入110.5天, 5.02人月				

手工测试与自动化测试12个月人力投入



用例设计效率提高2.5倍
回归测试效率提高32倍
12个月, 累计节省12.2人月