

敏捷测试实践与DevOps自动化测试实践培训班（面授/直播/录播）

产品名称	敏捷测试实践与DevOps自动化测试实践培训班（面授/直播/录播）
公司名称	北京中培伟业管理咨询有限公司
价格	.00/件
规格参数	
公司地址	北京丰台夏家胡同育芳园东里3号楼B座
联系电话	400-808-2006 13910781835

产品详情

培训背景

随着越来越多的企业开展数字化转型，人们看到云存储、云计算以及以DevOps思维方式在云平台上进行战略测试的趋势不断增长。而持续测试是DevOps不可或缺的一部分。使用佳DevOps测试策略的团队，将更快的软件开发和交付视为具有竞争力的业务差异化的重要因素。

培训对象

软件测试工程师、自动化测试工程师等

培训特色

- 1.理论与实践相结合、案例分析与行业应用穿插进行；
- 2.专家精彩内容解析、学员专题讨论、分组研究；
- 3.通过全面知识理解、专题技能和实践结合的授课方式。

课程收益

- 1.探索性测试、自动化测试在敏捷测试中的应用与开展
- 2.敏捷测试团队的组建与培养

课程大纲（4天，每天6-7小时）

时间	主题	内容描述
天	敏捷测试的认知&敏捷测试策略	<ol style="list-style-type: none"> 1. 你我认知的敏捷测试 <ul style="list-style-type: none"> 敏捷开发的原则 敏捷测试下的，测试与开发的关系 协同协作 2. 敏捷测试的核心关键点 <ul style="list-style-type: none"> 关注用户体验、系统使用场景 测试尽早介入 Code review、单元测试、自动化测试 持续集成、持续测试、随时随地测试 3. 敏捷测试人员的必要条件 <ul style="list-style-type: none"> 依据风险确定需求优先级 积极引入自动化测试 增加探索性测试的使用 适应从冲刺到冲刺的变化 4. 敏捷测试策略 <ul style="list-style-type: none"> 确定被测对象测试范围 确定测试技术（自动化测试技术）与质量标准 明确项目的测试流程 确定了每一种测试技术的具体使用方式 <p>注：包括待使用的框架和工具等</p> <ul style="list-style-type: none"> 探索测试的深度和广度；探索测试的重点和难点 统一项目内使用的测试相关的术语 确定了质量度量
	敏捷测试实施过程	<ol style="list-style-type: none"> 5. 测试策略和测试计划的差异 1. 敏捷测试的实施过程

		<ul style="list-style-type: none"> 前期项目信息收集 确立质量目标 确定测试类型 确定测试工具和框架 确定测试阶段 确定测试度量 2. 持续改进和风险分析
	敏捷测试过程下的人员要求与测试需求的提取	<ul style="list-style-type: none"> 1. 测试人员的思维 <ul style="list-style-type: none"> 用户角度 工具辅助思维整理 2. 测试需求的分析 <ul style="list-style-type: none"> 明确需求的范围 业务处理过程 需求背后的隐式需求 3. 测试需求分析何时进行？ 4. 为什么要进行测试需求分析？ <ul style="list-style-type: none"> 直观的需求（用例图/活动图） 需求转变 5. 如何开展测试需求分析？ <ul style="list-style-type: none"> 了解和学习需求 6. 测试需求分析的成果物是什么？ <ul style="list-style-type: none"> 明确测试范围-----功能点 功能的处理过程----单功能点 7. 测试需求分析的成果物是什么？ <ul style="list-style-type: none"> 明确测试范围 8. 需求分析和测试需求分析的区别？

		<ul style="list-style-type: none"> 需求分析 测试需求分析 持续改进和风险分析
	敏捷测试中如何高效的开展手工与自动化	<ul style="list-style-type: none"> 1. 敏捷团队中如何高效设计测试用例 如何精简测试用例条数：测试用例不是越多越好 等价类设计法 边界值设计法 基本路径分析法 场景设计法 因果图设计法等
	敏捷下的自动化测试的实施	<p>敏捷测试过程中如何构建高质量的自动化测试</p> <ul style="list-style-type: none"> 1. 自动化测试的实施背景 2. 如何建立一个务实明确的自动化测试目标 问题1：如何估算分析自动化测试效益 问题2：如何构建高收益成本比自动化测试目标 <p>敏捷测试下的自动化测试规范</p> <ul style="list-style-type: none"> 3. 敏捷测试下自动化测试的引入要求 自动化测试准入条件 自动化测试适用场景 4. 敏捷测试下自动化测试范围 分层测试：单元、接口、UI 5. 自动化测试开展时机 6. 自动化测试规范 工具规范 单元测试规范 接口自动化测试规范 UI自动化测试规范

7. 自动化测试流程

8. 项目团队

9. 风险评估

敏捷测试下的自动化测试开展

1. 确定自动化测试类型

2. 明确自动化测试活动

3. 敏捷测试过程中测试工作的开展

| 单元测试框架：采用TDD，覆盖率检查，QA review
UT，每次构建在CI执行

| 测试数据工具: faker等

| 集成测试框架

| 集成测试工具：Swagger、Jmeter等

| UI功能测试框架：selenium、UFT

| 性能测试：压力测试，负载测试

| 安全测试：渗透测试

4. 代码单元测试方案

| 基于TDD的单元测试框架编写单元测试代码

| 使用Mock模拟外部依赖测试

| 单元测试用例的实战

5. 接口自动化测试方案

| 轻量化接口测试

| 接口自动化测试框架

6. UI自动化测试方案

| UI自动化测试

| 自动化测试框架

敏捷策划中探索性测试的应用：

1. 软件测试需求分析与管理

| 需求分析与需求评审

	探索性测试策略的引入	<ol style="list-style-type: none"> 1. 软件测试策略方案认知与设计 2. 常规测试策略：有点至面、流程式 3. 结构化测试流程 <p>1 探索性测试的基本过程</p>
第二天	敏捷策划中探索性测试的应用： 探索性测试策略的系统认知	<ol style="list-style-type: none"> 4. 探索性测试的核心 5. 探索性测试的目标 6. 探索性测试的优点 7. 自由式探索式测试 8. 基于场景的探索式测试 9. 基于策略的探索式测试 <p>1 基于反馈的探索式测试</p>
	DevOps中如何高效的开展手工与自动化	<p>DevOps自动化测试的开展</p> <ol style="list-style-type: none"> 1. DevOps自动化测试的认知 2. DevOps下自动化测试的引入要求 3. DevOps下自动化测试范围 4. 自动化测试开展时机 5. 自动化测试规范 6. 自动化测试流程 7. 项目团队 8. 风险评估
第三天	单元测试方法TDD	<p>一、TDD编程单元划分定义</p> <ol style="list-style-type: none"> 1. 从“用户故事”到“程序文件”划分定义 2. 程序文件分类组织到层次化文件夹 3. 从“程序文件”到“数据与函数”划分定义 4. 接口层次、函数单元粒度与访问范围大小 5. 动手实践：代码单元划分定义实践 <p>二、TDD编程单元的测试</p>

1. 基于编程单元代码实现的测试用例与测试覆盖率
2. 基于编程单元声明定义的测试用例与测试覆盖率
3. 好的单元测试必须遵守AIR原则
4. 编写单元测试代码遵守BCDE原则
5. 编程单元的业务依赖与技术依赖
6. 基于单元黑盒的测试脚本标准编写
7. 全面完整取值与代码路径覆盖
8. 依据单元测试脚本编写实现代码的过程
9. 执行单元测试脚本检查实现代码是否满足标准
10. 实现代码重构时必须先修改测试脚本
11. 单元测试方法之逻辑驱动测试方法
12. 动手实践：单元测试框架案例实战

三、Mock那些事

1. Mock的作用
2. 揭示mock：mock的对象、mock的目的
3. mock在测试中的应用
4. mock的注意事项
5. 动手实践：Mock动手实践

四、TDD与Mock的深度实践

1. 复杂问题的TDD开发模式
2. 分而治之的思维：更小单元粒度TDD
3. 单元独立测试与组合单元测试
4. mock与patch
5. mock断言的作用与实践
6. mock断言与单元测试框架断言的结合：Assert_*_with等

		<p>7. mock统计</p> <p>8. 动手实践：单元测试与mock 的深度案例实践</p>
	DevOps接口自动化测试规范	<p>接口测试如何开展？如何规范化</p> <p>1. 接口测试</p> <ul style="list-style-type: none"> 内容 目的 <p>2. 接口测试规范</p> <ul style="list-style-type: none"> 接口测试的原则 接口测试的规则 接口文档的规范 <p>3. 接口测试用例</p> <ul style="list-style-type: none"> 测试用例编写规则 测试用例设计思路 测试用例设计方法 <p>4. 测试工具</p> <ul style="list-style-type: none"> 接口测试流程 接口测试质量评估标准 测试用例设计思考点
	Postman实操	<p>Postman全面了解</p> <p>1. get请求</p> <p>2. post请求</p> <p>3. 断言</p> <p>Postman实操</p> <p>1. Get请求：百度搜索案例</p> <p>2. Get请求案例：微信接口</p> <p>3. Post请求案例：微信群发消息</p> <p>4. Post请求案例：微信From表单</p>

		<ul style="list-style-type: none"> 5. Post请求案例：公众号菜单维护 6. 局部环境变量 7. 全局变量 8. 参数传值
	Jmeter接口测试	<ul style="list-style-type: none"> 1. 开源的力量—Jmeter <ul style="list-style-type: none"> 解析引入和使用Jmeter的原因 对比多种工具，阐述性能测试工具选型原则 阐述Jmeter的优势和劣势 2. Jmeter重要组件 <ul style="list-style-type: none"> Jmeter中的属性和变量 Jmeter中的采样器 Jmeter中的前置处理器和后置处理器 通过实例演示Jmeter组件作用域 3. Jmeter案例 <ul style="list-style-type: none"> Get请求：百度搜索案例 Get请求案例：微信接口 多接口间参数的传递
第四天	接口自动化实践	<ul style="list-style-type: none"> 1. Fiddler抓包 <ul style="list-style-type: none"> Fiddler界面介绍 Fiddler抓包 断点 重定向 过滤等 2. Requests库 <ul style="list-style-type: none"> Request讲解 Request案例 3. Unittest

4. Base函数与Request类封装

- | Base函数封装

- | Request类

5. 数据驱动操作

- | Excel操作

- | 数据框架讲解

6. 数据库操作

7. Mock

- | Mock简介与构造器

- | Mock创建

- | Mock的统计方法

- | Mock对象管理

持续集成

接口自动化框架

8. 接口自动化框架

- | 接口测试自动化框架的认知

- | 测试是自动化框架的工作流程

- | 测试框架的规划设计

- | 框架的详细设计与实现：公共配置模块的设置

- | 框架的详细设计与实现：通用模块的实现

- | 实战：接口自动化测试框架的设计

9. 手把手实现接口自动化框架

- | 通用模块那些事儿（DB、Http、Email、Log等）？

- | requests模块再回顾

- | xml的维护与操作

- | excel的操作（读、写）

- | 数据库的操作

		<p>实战：接口自动化测试框架的实现</p> <p>10. 一步一践接口自动化框架</p> <ul style="list-style-type: none"> 测试用例的设计与组织 测试报告的维护 测试用例的运行 测试结果的分析 <p>实践：接口自动化测试框架的运行</p>
	<p>DevOps UI自动化测试规范</p>	
	<p>WebDriver Api、POM设计模式</p>	<p>Python基础</p> <p>自动化测试基础</p> <ol style="list-style-type: none"> 1. 自动化简介 2. 分层的自动化测试 3. 什么样的项目适合自动化测试 4. 自动化工具对比 5. selenium 介绍 6. 成就大神之路 7. 前端工具介绍 <p>Python自动化环境的部署</p> <p>WebDriver API</p> <ol style="list-style-type: none"> 1. 从定位元素开始 2. 控制浏览器 3. 简单元素操作 4. 鼠标事件 5. 键盘事件 6. 获得验证信息 7. 设置元素等待

8. 模块驱动测试
 9. 关键字驱动测试
 10. 数据驱动测试
 11. 模块化测试模型实例
 12. 数据驱动测试实例(txt、xml文件等)
- 单元测试框架(可选pytest)
1. 单元测试框架的认知
 2. Unittest工作原理
 3. 单元测试用例的组织
 4. 构建测试固件
 5. 组织多个测试用例
 6. 构建测试套件
 7. 测试套件的优化
 8. 多个测试类的控制
 9. 单元测试执行的控制
 10. 测试用例的跳过

测试结果的美化、POM的设计、测试报告的发送

如何生成【漂亮的】测试结果

测试设计的升华—PO设计模式

1. 为什么使用page object (PO) 设计模式
2. PageObject案例实操：基本测试用例
3. 基础场景设计
4. 场景的优化
5. 测试用例的组织

通过邮件发送测试报告

案例实战：

1. 测试案例设计

		<ol style="list-style-type: none"> 2. 以邮件的形式发送html格式的测试报告 3. 给单人发送邮件 4. 给多人发送邮件 5. 发送『漂亮的』HTML格式的邮件
	<p>自动化框架的设计与实现</p>	<p>自动化框架的设计与实现</p> <ol style="list-style-type: none"> 6. 配置文件 7. 数据驱动 8. 日志管理 9. 自动化框架的架构设计 10. 框架的实现之公共文件库：配置文件的操作 11. 框架的实现之公共文件库：外部文件的操作 12. 框架的实现之公共文件库：测试用例信息的管理 13. 框架的实现之公共文件库：邮件的配置 14. 测试用例（页面）的设计与实现 <p>自动化测试模型</p> <ol style="list-style-type: none"> 1. 自动化测试模型概览 2. 模块化驱动测试实例 3. 数据驱动测试实例
	<p>敏捷测试过程中CI的应用</p>	<ol style="list-style-type: none"> 4. 持续集成 <ul style="list-style-type: none"> ┆ 软件交付痛点 ┆ 持续集成概述 ┆ 持续集成迫切需求 ┆ 持续集成的作用、过程和优势 ┆ CI工具 ┆ 应用脚本自动化构建 ┆ 持续测试 ┆ 持续代码质量

	<ul style="list-style-type: none"> 持续部署；持续改进 5. 敏捷开发& 持续交付 什么是持续交付；持续交付的研发流程 高质量的单元测试提高开发期质量的反馈 应用CI进行每日构建以频繁地交付可工作的软件 基本脚本的自动化部署缩短系统部署时间 6. 持续集成工具(CI) 持续集成工具(CI)环境部署 持续集成工具过程：创建任务、运行构建、定时执行、自动化构建
--	--

专家团队

李老师 计算机硕士 熟练使用QTP与LoadRunner、Selenium、Jemeter自动化测试工具，熟练掌握Vbs、shell、脚本语言，熟悉主流测试技术；熟悉QC、TD等测试管理工具；拥有丰富的测试项目管理经验；熟练搭建部署Linux、win2008主流服务器；熟悉oracle数据库部署与维护，熟练使用PL/SQL语言；熟练部署MOSS门户网站，熟悉规划设计MOSS门户网站 workflow。精通课程：《测试基础》、《测试流程优化》、《loadrunner》、《敏捷测试》、《测试管理》、《测试技术》、《QuickTest professional》、《ALM&Q C》、《自动化测试工具selenium》、《Testlink与缺陷管理平台整合》、《测试管理》。

程老师 中国科学院软件研究所硕士，主要研究方向：架构设计、项目管理实践、大型高可用高并发架构设计、微服务架构、软件测试等等。熟悉网络分布式计算、数据库、网络操作系统，精通J2EE、SQL、中间件服务器。在J2EE技术领域具有理论功底和实践经验。在J2ME商务应用和基于J2ME游戏开发领域具有深厚的理论功底和丰富的实战经验。主要授课方向：DevOps落地实践、微服务架构、软件构架设计、UML、GO语言、OOAD、J2EE企业级应用开发等。擅长架构企业级应用；有独立 workflow 引擎开发、企业系统应用集成方面的丰富经验；在企业门户（Portal）、报表、 workflow 引擎和企业服务总线（ESB）等应用领域有深入的理论研究和充分的实践；对软件开发的整个流程有深刻认识，有很好的协作精神和学习能力。在架构下的系统设计和开发方面均有丰富经验。能够结合实际,在复杂的应用环境中选择适合的技术组合并合理利用开源中间件来降低项目开发风险、缩短开发周期、提高应用系统的可维护性和可扩展性。

培训费用

线下面授培训费：9800元/人（含培训费、场地费、资料费、学习期间午餐及录播视频回放一年）。

网络直播培训费：7800元/人（含培训费、平台费、资料费及直播视频回放一年）。

结业证书

本课程由中国信息化培训中心颁发《敏捷测试佳实践与DevOps自动化测试实践工程师》证书；

证书可作为技术人员职业能力考核的证明，以及技术人员岗位聘用、任职、定级和晋升职务的重要依据

。