

before

上课时间

上午：8:30 - 12:30

下午：2:30 - 4:30

关于讲师

张开，测试、python、elasticsearch(搜索引擎)

博客地址：<https://www.cnblogs.com/Neeo/p/10864123.html>

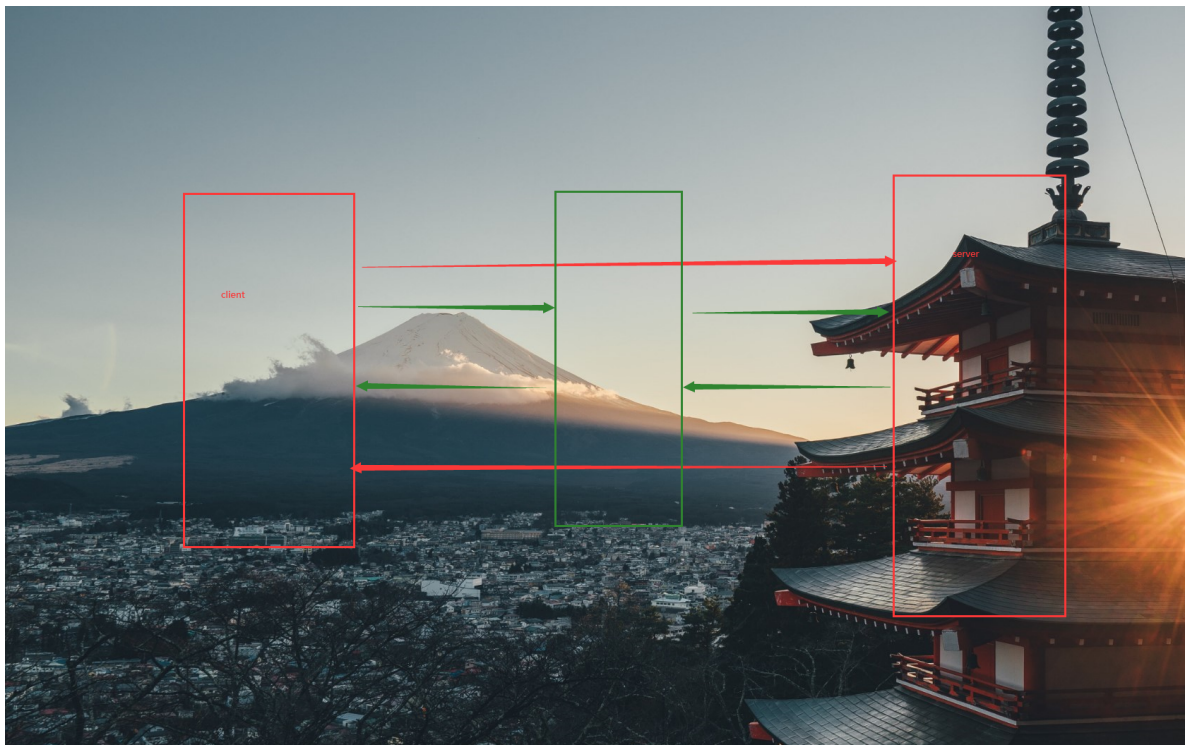
<https://www.cnblogs.com/Neeo/p/10864123.html>

课程安排

暂定讲3周

主要讲：

- 测试基础
 - 测试模型
 - 常用的测试方法
 - 缺陷，禅道
- 接口测试
 - requests
 - unittest
 - pytest
 - 发邮件
 - 自动化测试脚本
 - postman
 - 基于django的接口测试平台
- selenium
 - 模拟人对浏览器的操作
- fiddler，抓包工具，原理如下



- 移动端测试
- 性能测试
 - jmeter, 接口测试工具, 跟postman功能有类似的地方
 - 压测
 - 分布式压测

测试部分, 不难, 知识点多, 工具多

今日内容

测试理论

二八原则

测试用例

1. 为什么需要测试

1. 软件本身越来越复杂
2. 需要一直在变
3. 有些业务必须进行测试

1. 支付业务
2. 银行, 金融

2. 为什么选择测试

1. 好就业
2. 测试本身的技术门槛提高, 含金量提高
3. 岗位需求慢慢变多

3. 什么是测试?

Glenford J. Myers 在《软件测试的艺术》一书中有这样的定义：**测试是为了发现错误而执行程序的过程。**

另外，软件专家温伯格和 Cem Kaner 也提出了自己对软件测试的理解，在温伯格的《完美软件》一书中提到：**测试是一个获取信息的过程，用来降低决策风险。**Cem Kaner教授也提出：**软件测试是一种技术调查，其目的是向关系人提供有关产品（软件、系统或服务）质量的实验信息。**

总结，软件测试的定义：通过手工或者相关工具，对被测对象进行测试操作。从而验证实际与预期结果是否存在差异。

二八原则

缺陷存在群集现象，首先要了解一个**二八理论**，即对于软件的功能来说，核心功能占20%，非核心功能占80%（当然，不是绝对的）。那么在测试中，我们会集中测试20%的核心功能。所以，这部分发现缺陷的概率会远高于80%非核心部分。也因此我们遇到的缺陷就都会集中在20%的核心功能这块。

杀虫气现象：

杀虫剂现象，同样的测试用例不能重复执行多次，因为软件会对它产生免疫。比如说，你用 $3 * 3$ 测试出代码不等于 9，把这个缺陷提交给开发，开发随后解决了这个bug，那我们再测试的时候，就不要用 $3 * 3$ 来测试了，因为开发在改bug的时候，想法设法的让 $3 * 3 = 9$ ，所以，同样的用例，软件会对它产生**免疫**。

测试用例定义：

测试用例（Test Case）是为特定的目的而设计的一组测试输入、执行条件和预期的结果，以便测试是否满足某个特定需求，通过大量的测试用例来检验软件的运行效果，它是指导测试工作进行的重要依据。

要牢记，测什么和怎么测

测试用例编写重点

- 用例编号：产品名字-测试阶段
- 测试项目：对应一个功能模块
- 测试标题：直接对测试点进行细化得出
- 重要级别：高/中/低
- 预置条件：需要满足一些前提条件，否则用例无法执行
- 测试输入：需要加工的输入信息，根据具体情况设计
- 操作步骤：明确给出每个步骤的描述，执行人员根据该步骤执行工作
- 预期结果：根据预期输出对比实际结果，判断被测对象是否符合需求。
- 实际结果：根据实际结果，填写报告。（可写可不写）

<https://img2018.cnblogs.com/blog/1168165/201911/1168165-20191105002800487-476153924.jpg>

测试计划书编写重点

<https://img2018.cnblogs.com/blog/1168165/201911/1168165-20191105002935587-319846490.png>

兼容性测试

向前兼容性测试(forward compatibility testing): 测试的应用程序或软件在新的或即将到来的版本, 并且应用程序的早期版本能够打开较新版本中的文件并忽略早期版本中未实现的功能。比如USB1.0能够兼容USB3.0, 或者是MS office2003能够使用转换器打开MS office2007的文件, 并忽略MS office2007的新功能。

向后兼容性测试(backward compatibility testing): 测试的应用程序或者软件处于旧版本, 并且应用程序的新版本能够顺利处理旧版本的程序数据。比如说USB3.0兼容USB1.0, 或者MS office 2007能够打开MS office 2003的文件。

版本管理工具

常用的版本管理工具:

- SVN, 稍微老了点, 简单易用, 现在用的不多了
- Git, 优势借助GitHub, 私有仓库, 目前Git用的较多
 - 掌握, git的基本操作
 - 处理合并冲突

软件开发完毕, 创建一个版本, v1, 在该版本下修复一些bug, 完事发布

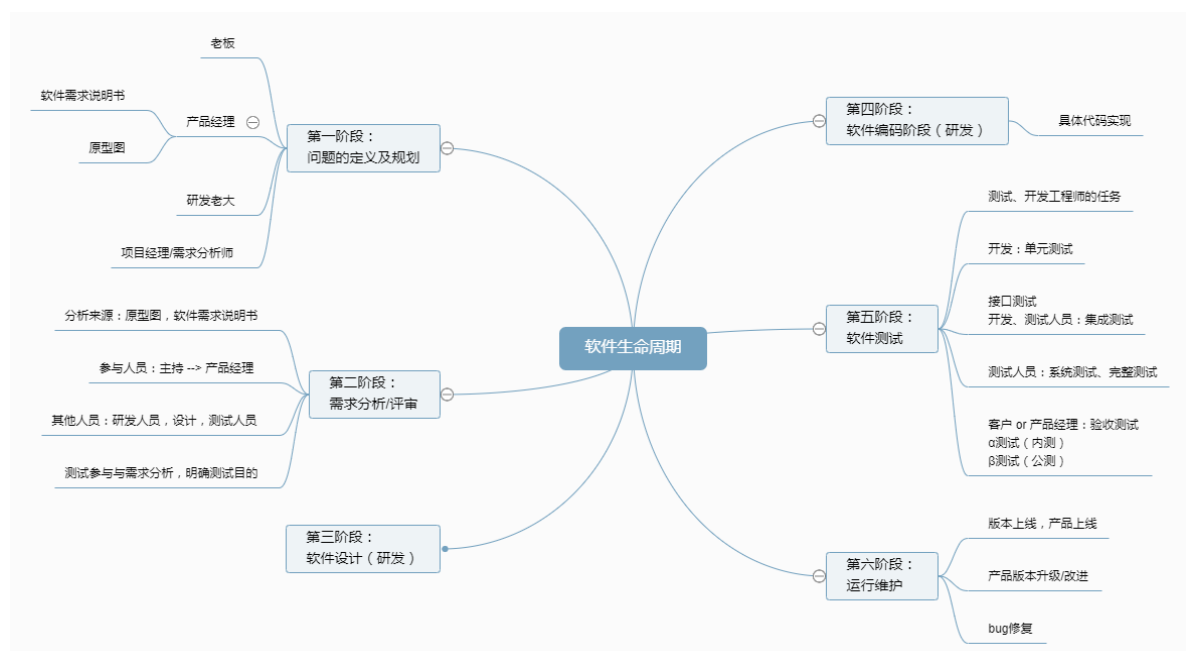
开发v2版本, 此时发现的v1版本的问题, 在v2版本中修复

在v2版本中, 做描述

软件的开发/测试模型和测试分类

参考: <https://www.cnblogs.com/Neeo/articles/11795967.html>

软件的生命周期



在周期内，我们无论是开发还是测试都依赖于某个模型进行作为依据，有效地提高开发、测试效率。

开发模型

常见的开发模型：

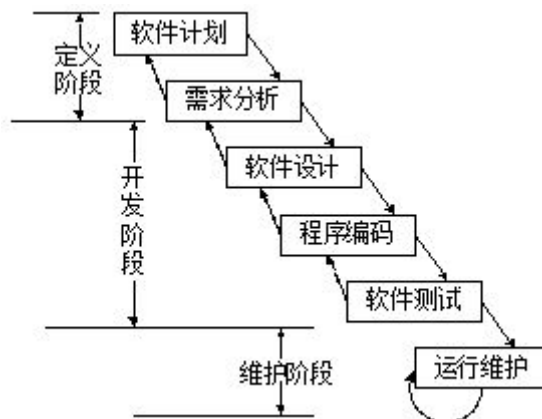
- 瀑布模型

优点

- 明确划分了软件生命周期的各个环节。
- 强调早期软件计划，需求分析比较重要。
- 清晰的工作流程，便于分工协作。
- 适合需求稳定的产品开发。
- 每个阶段都有一个检查点。

缺点

- 线性的开发流程，存在巨大的风险。
- 依赖于早期的需求调查，不适应需求的变化，单一流程不可逆。
- 风险在后期可能才会暴露，且无法纠正，导致项目的失败。
- 无法保证用户的产品需求不变，开发过程无法更改。比如盖房子，照着图纸打好的地基可以承载7层楼，现在用户突然要加五层，那么地基就得重打，已经盖好的楼就得爆破，当然这是不可能的操作。



- 快速原型模型

第一步是构建一个快速原型，实现用户与系统的交互，用户对原型进行评价，进一步细化待开发软件的需求，通过逐步调整原型使其满足用户的要求，开发人员可以确定用户的真是需求是什么。

第二步是在第一步的基础上开发出用户满意的软件产品。

优点

克服瀑布模型的缺点，更好的满足用户的需求并减少由于软件需求不明确带来的项目开发风险，适合预先不能确切定义需求的软件开发。适合开发小型的、灵活性高的系统。

缺点

不适合大型系统的开发，前提要有一个展示性的产品原型作为参考，因此在一定程度上可能会限制开发人员的创新。

- 螺旋模型
- 敏捷开发

测试模型

V模型

V模型和瀑布开发模型有着一些共同的特性，V模型中的过程从左到右，描述了基本的开发过程和测试行为。

V模型的价值在于它非常明确地标明了测试过程中存在的不同级别，并且清楚地描述了这些测试阶段和开发过程期间各阶段的对应关系。

局限性：把测试作为编码之后的最后一个活动，需求分析等前期产生的错误直到后期的验收测试才能发现。

V模型的优点

- V模型既包含了底层测试有包含了高层测试：
 - 底层测试，如单元测试。
 - 高层测试，系统测试。
- V模型清晰的标出了软件开发的各个阶段。
- V模型自顶而下逐步求精的方式把整个开发过程分成不同的阶段，每个阶段的工作都很明确，因此便于控制开发过程，当所有的阶段都完成后，该软件开发过程也随之结束。

V模型的缺点

- 缺点也显而易见，正是它自身的顺序性导致，只有到了测试阶段，才开始执行测试流程。但有些错误直到这时才被发现，甚至无法发现，沉珂已久，回天乏术.....
- 另外，在实际的开发过程中，在需求阶段很难把用户的需求完全明确下来，因此，当需求变更时将会导致阶段反复（如重新分析需求、设计、编码、测试等过程），返工量非常大，模型灵活性比较低。

W模型优点

- 测试伴随整个软件开发生命周期，测试对象不仅仅是程序，需求和概要同样需要测试。
- 更早介入测试，可以更好的发现需求设计中的缺陷，修复成本也更低。
- 同样分阶段工作，便于控制项目过程。

W模型缺点

- 依赖于软件开发和测试的前后线性关系，还是无法解决需求变更调整的问题。也就是无法支持迭代，自发性和需求变更调整。
- 对于有些项目，在执行过程中根本不产生文档，那么W模型也基本无法适用（小型公司直接产出原型图，并不写需求说明书）。
- W模型的技术复杂度较高，对于需求设计和测试人员要求较高，实践起来，门槛较高。

H模型的优点

- H模型揭示了软件测试除了测试执行外，还需要有很多的其他工作。
- 软件测试完全独立，贯穿整个生命周期，且与其他流程并行进行。
- 软件测试活动可以尽早的进入准备阶段，尽早执行，灵活性较强。
- 软件测试可以根据被测对象的不同可以分为多层次、分阶段、分次序的执行，同时也是可以被迭代。

H模型的优点

- 管理型要求高：由于模型的灵活性，必须有相应清晰的规划和管理制度。否则测试过程将非常难以管理控制。
- 技能要求高：H模型要求能够很好的定义每个迭代的规模，不能太大也不能太小。

- 测试就绪点分析困难：在测试阶段，我们并不知道测试准备到什么程度算是合适的，也就是就绪点很难把握。比如说就绪点在哪里？就绪的标准是什么？这对测试的执行带来很大的困难。
- 对于整个项目的人员素质要求非常高：在规范的制度下，工作的效率很高，但是由于个人的技能不足，会导致因为一个点出问题，导致整个项目的进度受到干扰。

V、W、H模型总结

V模型适用于中小企业。

W模型适用于中大型企业，因为对于项目组成员要求高。

H模型对项目组成员要求非常高，很少有公司采用。

翻墙

<https://freemycloud.xyz/auth/register>

谷歌上网助手

今日作业

预习软件分类：

<https://www.cnblogs.com/Neeo/articles/11795967.html#%E6%B5%8B%E8%AF%95%E5%88%86%E7%B1%BB>

预习常见的测试方法：<https://www.cnblogs.com/Neeo/articles/11795983.html>