## 软件测试流程

1. 测试需求分析阶段：阅读需求，理解需求，参与需求评审会议。通过产品需求整理出测试需求。
2. 测试计划阶段：主要任务是编写测试计划，计划内容有：测试目标、测试范围(来自需求文档)、测试策略(方法)、资源配置(人力物力的分配)、进度安排、测试周期，风险评估及规避措施等。

3）测试设计阶段：主要任务是编写测试用例，用例编写完成后会进行评审。

测试环境的规划、自动化测试框架的设计，测试数据和测试脚本的设计、测试工具的设计和选择。

4）测试执行阶段：首先搭建测试环境，执行预测，以判定当前版本可测与否，如果预测通过，正式进入系统测试，遇到问题提交bug到缺陷管理平台，并对bug进行跟踪，直到被测软件达到测试需求要求，没有重大bug，测试结束----（完善测试用例）

5）测试总结阶段：出测试报告，对整个测试的过程和版本质量做一个详细的评估，确认是否可以上线。

#### 面试题：你们公司的测试流程是怎么样的？各个阶段的输出是什么？

测试需求阶段：了解测试需求，整理出测试需求；

测试计划阶段：编写测试计划；

测试设计阶段：编写测试用例；

测试执行阶段：提交bug到问题库，直到没有重大bug，测试结束，完善测试用例；

测试评估阶段：确认是否可以上线，编写测试报告

## 软件测试生命周期

#### 、基于瀑布模型（自上而下的）

##### 1、可行性研究及计划

开发方和需求方共同讨论，确定软件的开发目的及可行性，并制定实施计划。通过确定软件开发目的，给出软件的功能、性能、可靠性、接口等方面的设想。研究完成这个项目的可行性，问题的解决方案，对资源、成本、完成时间的估计，制定实施计划。

##### 需求分析

在确定软件开发可行的情况下，对软件需要实现的各个功能进行详细分析。弄清用户对软件系统的全部需求，并给出确切描述，产出《需求规格说明书》。

##### 软件设计

根据需求分析的结果，对整个软件系统进行设计，如：系统框架设计、数据库设计等，产出《设计说明书》。

##### 编码

按照软件设计的结果，程序员开始编写代码。

##### 软件测试

单元测试：对代码的测试，一般由开发完成

集成测试：对接口的测试，在单元测试之后进行，由开发完成

系统测试：比对需求规格说明书，根据测试用例进行完整的测试，如各功能是否满足需求，系统运行是否存在漏洞

验收测试：用户对软件进行验收，客户拿到软件后，会根据用户需求来进行判定软件是否达到需求

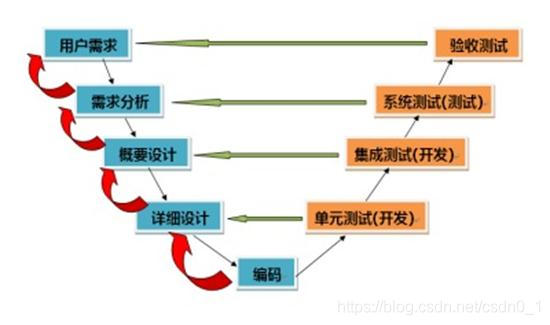
##### 运行维护

#### 、V模型（瀑布模型的变种）

以“编码”为黄金分割线，将整个过程分为开发和测试，并且开发和测试之间是串行的关系，同时，测试过程中存在的若干不同的测试阶段，并与每一个开发阶段对应。

优点：是软件开发瀑布模型的变种，改进了瀑布模型，反映了测试活动与分析和设计的关系。

缺点：测试过程作为需求分析、概要设计、详细设计及编码之后的一个阶段。该模型使人容易理解为：主要是针对程序进行测试，寻找错误，忽略了对需求、设计的测试。需求分析阶段的错误，直到后期才被发现，没有体现“尽早、不断地进行软件测试” 的原则。这种开发周期长，修复错误的周期长，所以修复成本也高。



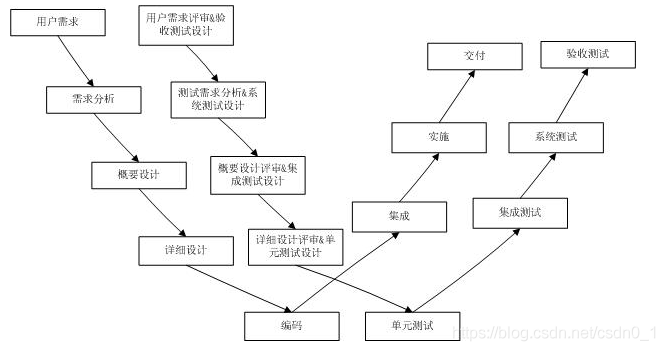
* ****单元测试****：是模块测试，验证软件的基本组成单位的正确性，是白盒测试
* ****集成测试****：是模块间的测试，测试接口（软件各模块之间的接口和软件与硬件之间的接口）是否正确，是灰盒测试（白盒和黑盒结合）
* ****系统测试****：系统测试包括：冒烟测试 系统测试 回归测试  
   (1) 冒烟测试：主干流程测试，确认软件的基本功能正常，可以进行后续的测试工作  
   (2) 系统测试：是检测系统的功能、质量、性能能否满足系统的要求，包括功能、性能、界面、可靠性、兼容性等等，是黑                          盒测试  
   (3) 回归测试：修改了旧代码之后重新进行测试，确认修改后的代码没有引入新的错误或导致其他代码产生新的错误
* ****验收测试****：是确保软件能否满足用户的需求或合同的要求

#### W模型（V模型的改进）

W模型：是由两个V模型组成，一个是开发阶段，一个测试阶段 ，可以看出：在W模型中，开发和测试是并行的关系

优点：基于“尽早、不断地进行软件测试”的原则下，在V模型中增加软件各开发阶段应同步进行的测试，演化为W模型。强调：测试伴随着整个软件开发周期，而且测试的对象不仅仅是程序功能，需求和设计同样要测试，测试与开发是同步进行的，从而有利于尽早地发现问题。

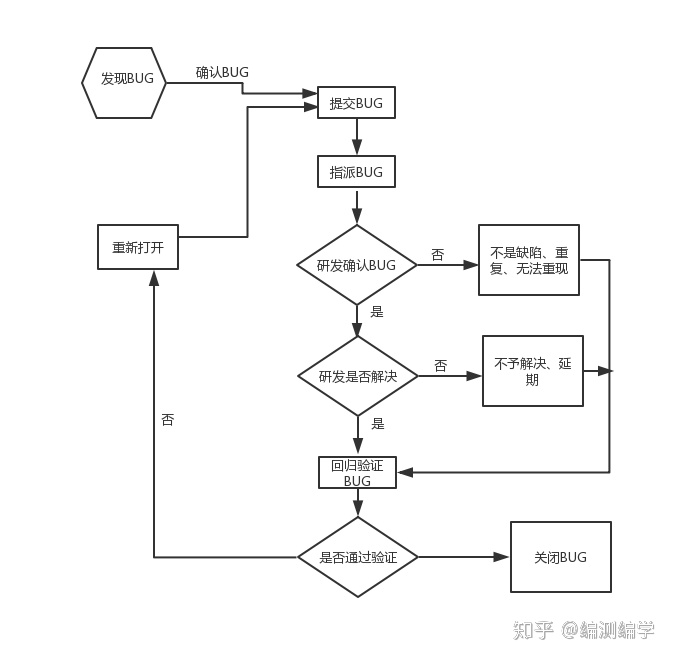
缺点：虽说开发和测试是并行的，但该模型的整体还是串行的 (以需求为起点，到测试结的过程)只有上一阶段完成后，才可以开始下一阶段的活动，无法支持敏捷开发模式



## 测试用例的生命周期

确定要写测试用例→ 设计测试用例(用到的技术) →编写测试用例→ 执行测试用例 → 管理测试用例

## BUG的生命周期



**不是缺陷、设计如此**: 1、核对需求规格说明书  2、找业务或者产品进行确认

**重复**：确定提交重复后直接关闭后提交的

**无法重现**：

1、首先出现难以复现的bug一定要截图提交bug

2、其次评估bug的重要程度以及对整个项目的影响，如果影响小，就记录下来，继续跟踪

3、如果对项目影响较大并且多次测试（20次左右），还是无法复现，提交给项目经理组织团队进行研究

**不予解决、延期**：与项目经理确认

## BUG的严重等级

#### 、致命错误：

（1）常规操作引起的系统崩溃，死机，死循环

（2）造成数据泄漏的安全性问题，比如恶意攻击造成的账户私密信息泄露

（3）涉及金钱

（4）用户数据受到破坏，数据丢失。

#### （二）、严重错误：

（1）重要功能不能实现

（2）错误的波及面广，影响到其他重要功能正常实现

（3）非常规操作导致的程序崩溃、死机、死循环

（4）密码明文显示

#### （三）、一般错误：

（1）次要功能不能正常实现

（2）操作界面错误，例如：列名与列名下的内容不一致

（3）查询错误、数据错误显示

（4）删除操作未给出提示

#### （四）、轻微错误：

（1）界面不规范

（2）辅助说明描述不清楚

（3）提示窗口文字未采用行业术语

（4）界面存在文字错误

（5）改进意见：可以提高产品质量的建议， 包括新需求和对需求的改进

## BUG的优先级

1.immediate（立刻，一类）

      问题马上要解决，否则系统根本无法达到预定需求

2.Urgent（紧要、优先，二类）

      问题修复很紧要，关系到主要功能模块

3.high(高度重视，二类或者三类）

      有时间就要马上解决，否则系统偏离需求要求较大或预定功能

4.Normal(正常，三类)

      进入个人计划解决，表示问题不影响需求实现，但是影响其它使用方面， 比如：页面调用出错，调用了错误等。

## 七、提交bug包括哪些内容

项目名称，所属模块，提交人员，指派人员，版本号，严重程度、优先级、bug标题,重现步骤，预期结果，实际结果，截图、报错日志等。

## 八、bug类型

代码错误、设计缺陷、用户界面、需求变动、新增需求、兼容性问题、性能问题、安全相关

## bug的定义

狭义指软件程序的漏洞或缺陷；广义指与需求文档存在差异的功能实现测试工程师，或用户提出的软件可改进的细节。

## 软件测试的目的

狭义上是为了尽可能多的找出软件存在的漏洞或缺陷；广义上软件测试是为了提高软件质量，保证软件安全、降低商业风险，提高用户体验。