

第11期-单元测试&集成测试&功能测试

2019年11月29日 13:52

复杂的软件在修改了代码后，一般都要进行测试，手动测试是最慢、最不可靠且最昂贵的软件测试方式，遗憾的是，如果应用程序在设计上不具备可测试性，这可能是唯一可行的方式，尽量让设计的程序具有可测试性是架构的一个重要内容。

自动测试类型

软件应用程序自动测试具有多种类型。最简单最低级别的测试是单元测试。级别稍高的测试包括集成测试和功能测试。其他类型的测试不在本文讨论范围之内，例如 UI 测试、负载测试、压力测试和版本验收测试。

单元测试

测试应用程序逻辑单元部分。单元测试并不测试代码如何处理依赖项或基础结构 - 这是集成测试的用途，单元测试不测试编写代码所用的框架，单元测试完全在内存或进程中运行。它不会与文件系统、网络或数据库通信。单元测试仅测试方法代码。

集成测试

与单元测试相比，由于集成测试通常依赖外部依赖项和基础结构，因此速度较慢，难度较大。如果能够使用单元测试，请优先使用单元测试。如果不能，再考虑使用集成测试。集成测试通常具有更复杂的初始化和拆卸过程。

功能测试

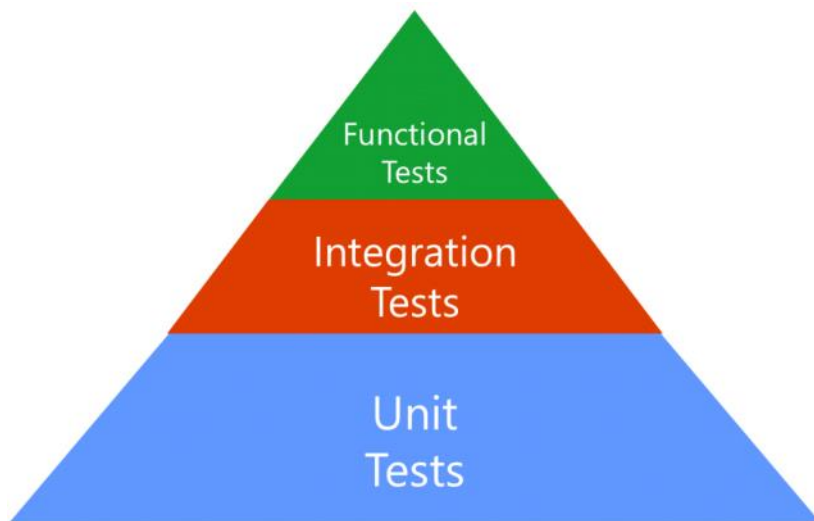
单元测试和集成测试从开发者角度编写，用于验证系统的一些组件是否能共同正常运行。

功能测试从用户角度编写，用于基于其要求验证系统的正确性。

由于功能测试在系统级别运行，所以可能需要一定程度的 UI 自动化。与集成测试一样，它们通常也适用于某种类型的测试基础结构。这使其功能测试比单元测试和集成测试更慢、更易发生故障。

测试金字塔

该金字塔的不同层次及其相对大小表示不同的测试类别，以及应为应用程序编写的测试数量。如图所示，建议以大量单元测试作为基层，中间以较小的集成测试层作为支持，顶端为更小的功能测试层。理想情况下，每层应仅包含较低层中无法充分执行的测试。确定特定方案所需的测试类型时，请考虑此测试金字塔。

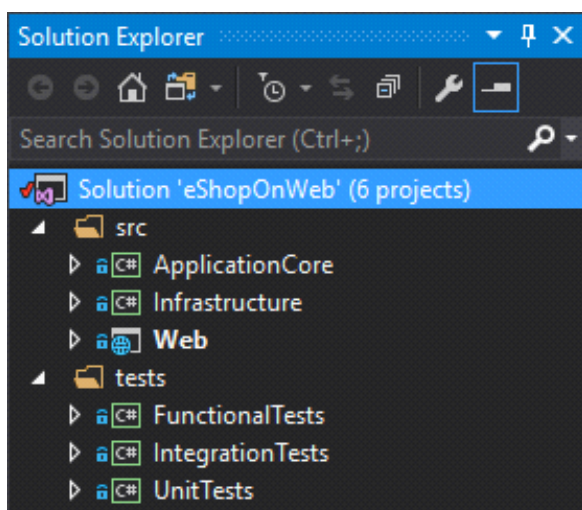


要测试的内容

对缺乏编写自动测试经验的开发人员而言，一个经常遇到的问题是确定测试内容。不妨从测试条件逻辑开始。如果方法中包含基于条件语句（if-else、switch 等）更改的行为，应能至少编写数个用于在某些条件下确定行为是否正确的测试。如果代码存在错误条件，建议通过代码为“主逻辑”编写至少一个测试（不含任何错误），并为“副逻辑”编写至少一个测试（含错误或非典型结果），以确保应用程序在错误情况下按预期运行。最后，重点测试可能失败的项，而不是关注代码覆盖率等指标。一般而言，高代码覆盖率比低代码覆盖率更好。但是，与仅仅为了改善测试代码覆盖率指标而编写自动属性测试相比，编写复杂的业务关键型方法的测试更有意义。

组织测试项目

一种常见方式是在“src”文件夹下组织应用程序项目，在平行的“tests”文件夹下组织应用程序的测试项目。如果你认为这种组织方式有用，可以在 Visual Studio 中创建匹配的解决方案文件夹。



《ASP.NET Core 视频教程》

第35期-单元测试

认识单元测试，单元测试的最佳做法，使用xUnit 进行单元测试，使用MSTest 进行单元测试，使用NUnit 进行单元测

试，三种测试框架的最佳实践，单元测试管理器，使用Live Unit Testing 实时显示测试结果，查看代码测试覆盖率。

第36期 集成测试&远程调试&源码跟踪

在ASP.NET Core 中测试控制器逻辑，为控制器编写单元测试，单元测试中的Fake、Stub 和Mock 术语，使用Moq 框架进行单元测试，轻量级集成测试，了解Selenium、Html Agility Pack 和AngleSharp 框架，远程调试服务器程序，源码跟踪调试配置。

测试命名规则

避免使用 ControllerTests.Test1 等模糊的测试名称，因为查看测试结果时，此类名称不能提供任何价值。

命名规则1：

CatalogControllerGetImage.CallsImageServiceWithId

CatalogControllerGetImage.LogsWarningGivenImageMissingException

CatalogControllerGetImage.ReturnsFileResultWithBytesGivenSuccess

CatalogControllerGetImage.ReturnsNotFoundResultGivenImageMissingException

命名规则2：

CatalogControllerGetImage**Should**.CallImageServiceWithId

CatalogControllerGetImage**Should**.LogWarningGivenImageMissingException

如果使用类似上述会产生众多小测试类的命名约定，建议使用文件夹和命名空间进一步组织测试。

