**به نام خدا**

تمرین کامپیوتری اول – تست نرم افزار – بخش تئوری

امیرحسین راحتی 810100144

علی ممتحن 810100213

لینک مخزن پروژه : <https://github.com/AmirhosseinRHT/Software-Testing>

شناسه آخرین کامیت: **6b03d76f6df0a8cb53c9913b25a091dc0c980e2c**

* پاسخ سوالات بخش تئوری

1. در State Verification، ما وضعیت نهایی سیستم را پس از اجرای عملیات بررسی می‌کنیم. یعنی، نتیجه و خروجی نهایی مورد ارزیابی قرار می‌گیرد .نه جزییات هر مرحله از اجرا

در Behavior Verification، رفتار و تعاملات بین اجزای سیستم مورد بررسی قرار می‌گیرد. یعنی اینکه آیا متدها با

پارامترهای درست و به ترتیب صحیح فراخوانی شده‌اند.

در State Verification معمولاً از Stub استفاده می‌کنیم چون فقط نیاز داریم داده‌های از پیش تعیین شده برگردانده شوند.

در Behavior Verification از Mock استفاده می‌کنیم زیرا نیاز به بررسی دقیق تعاملات و فراخوانی‌ها داریم.

1. Test Spy‌ها ابزارهایی هستند که رفتار و تعاملات حین اجرای برنامه را ثبت می‌کنند تا بعداً بتوانیم آن‌ها را بررسی کنیم. انواع اصلی Test Spy‌ها عبارتند از:
2. Method Call Spy: فراخوانی متدها و پارامترهای آن‌ها را ثبت می‌کند. می تواند به صورت partial هم باشند . یعنی بخشی از یک تابع این خاصیت را داشته باشد و بقیه تابع به کار خودی خود بپردازد
3. Property Spy: دسترسی و تغییرات property‌ ها را ثبت می‌کند
4. Mock Spy: ترکیبی از mock و spy است که به تست‌کننده امکان می‌دهد تا علاوه بر ثبت فراخوانی‌ها، رفتار خاصی را نیز تعریف کند که در صورت نیاز فعال شود.
5. Event Spy: event ها و پارامترهای آن‌ها را ثبت می‌کند

Test Spy ‌ها برای موارد زیر استفاده می‌شوند:

* بررسی ترتیب فراخوانی متدها
* ثبت پارامترهای ورودی
* بررسی تعداد دفعات فراخوانی یک متد

1. وقتی ایجاد fixture زمان‌بر است - وقتی fixture تغییر نمی‌کند و بین تست‌ها مشترک است - وقتی حجم داده‌های fixture زیاد است
2. مزایای: Lazy Setup مصرف منابع کمتر - اجرای سریع‌تر تست‌ها - خوانایی بیشتر کد – کاهش فشار به سیستم برای لود دیتای بلا استفاده – کاهش زمان اجرا

معایب: Lazy Setup پیچیدگی بیشتر در مدیریت - احتمال تکرار کد - نیاز به کنترل بیشتر و checking بیشتر در کد

1. استفاده از الگوی Builder برای ساخت fixture - ایجاد یک نسخه جدید از fixture برای هر تست - استفاده از ترنزکشن ها برای بازگرداندن وضعیت قبلی : در این روش ، با تعیین یک شرط ، خود برنامه در صورت برآورده شدن آن شرط ، برنامه را به حالت قبل اجرای آن متد میبرد (در اسپرینگ وجود دارد) - پیاده‌سازی مکانیزم Rollback برای بازگشت به حالت اولیه

* باگ های کد

1. پارامتر های متد های Reservation Controller چک نمیشد که مقادیر درستی داشته باشند.