

Testing
Software



SOFTWARE TESTING



تست نرم افزار

تابستان ۹۸ - ناوشگران عصر پارسه

مهدی رحمان زاده

شما میتوانید جهت استفاده از سایر ارائه ها به آدرس [NapTeam](#) مراجعه نمایید.

تست نرم افزار چیست؟

تست نرم افزار یک فعالیت برای بررسی این است که آیا نتایج واقعی (Actual Result) با نتایج مورد انتظار (Expected Result) مطابقت دارند یا نه!

همچنین برای حصول اطمینان از این موضوع است که سیستم نرم افزاری بدون نقص (Defect Free) است.



تست نرم افزار به شناسایی خطاها (Error)، شکافها (Gap) یا نیازمندی های (Requirement) گم شده در مقابل نیازمندی های واقعی کمک می کند.

تست نرم افزار را می توان به صورت دستی یا با استفاده از ابزارهای خود کار انجام داد.



چرا تست نرم افزار مهم است؟

تست نرم افزار مهم است، چرا که باگ های نرم افزاری می توانند گران یا حتی خطرناک باشند.

باگ های نرم افزاری می توانند به طور بالقوه باعث خسارت های مالی و یا انسانی شوند، تاریخ پر از نمونه هایی از این دست است

مثلا Starbucks مجبور شد حدود ۶۰ درصد از فروشگاه های ایالات متحده آمریکا و کانادا را به علت نارسایی نرم افزاری در سیستم POS خود ببندد چون فروشگاه قهوه را به صورت رایگان سرو می کرد، به شکلی که آنها قادر به پردازش تراکنش نبودند یا یک باگ نرم افزاری باعث شد حساب بانکی ۸۲۳ مشتری از یک بانک بزرگ در ایالات متحده به مبلغ ۹۲۰ میلیون دلار آمریکا شارژ اعتبار شود.

وارسی و اعتبارسنجی

- تست نرم افزار به طور رسمی جزئی از بازبینی و اعتبارسنجی نرم افزار می باشد.

- این دو واژه به صورت زیر تعریف و با هم مقایسه می شوند:

وارسی: آیا محصول را به درستی می سازیم؟

اعتبارسنجی: آیا محصول درستی را می سازیم؟

- وارسی بررسی می کند که آیا نرم افزار از مشخصاتش پیروی می کند یا نه.

- اعتبارسنجی باید تضمین کند که نرم افزار انتظارات مشتری را برآورده می سازد.

- توجه: گاهی آنچه که در مشخصات می آید ممکن است دقیقاً خواسته های مشتری را برآورده نسازد.



• در واقع هدف فرآیند تست در یک جمله "ایجاد اعتماد نسبت به سیستم" است. اما چطور می توان نسبت به سیستم ایجاد اعتماد نمود؟ عوامل مختلفی در این امر دخیل هستند اما نقش تست پررنگ تر می باشد. به همین دلیل تست دو هدف اصلی را دنبال می کند:

• ۱- تست اعتبارسنجی: موفقیت آن در درست کار کردن سیستم است.

• ۲- تست نقص: موفقیت آن در نمایان سازی خطاهایی است که موجب کارکرد نادرست

سیستم می شود.

اهداف تست

- تست، فرآیند اجرای برنامه به قصد یافتن خطاهاست.
- تست موفق، تستی است که خطاهای کشف نشده را کشف می کند.
- یک دیدگاه اشتباه: تست موفق، تستی است که در آن خطایی یافت نشود!
- تست نشان می دهد که عملکردهای نرم افزار (ظاهرا) مطابق انتظار کار می کنند، و خواسته های رفتاری و کارایی مشتری برآورده شده اند. به علاوه، داده های جمع آوری شده به موازات انجام تست، شاخص خوبی از قابلیت اطمینان نرم افزار و شاخصی از کلیت کیفیت نرم افزار به دست می دهند ولی تست نمی تواند نبود خطاها و نقایص را ثابت کند بلکه فقط می تواند نشان دهد که خطاها و نقایص وجود دارند.
- در واقع هدف فرآیند تست در یک جمله "ایجاد اعتماد نسبت به سیستم" است.

انواع تست نرم افزار

- به طور معمول تست نرم افزار به سه دسته تقسیم می شود:

- تست کارکردی یا Functional

- تست غیر کارکردی یا Non-Functional

- تست نگهداشت یا Maintenance

Functional Testing

- تست یونیت (Unit Testing)
- تست یکپارچه سازی (Integration Testing)
- تست دود (Smoke Testing)
- تست پذیرش کاربر (UAT-User Acceptance Testing)
- تست محلی سازی (Localization Testing)
- تست جهانی سازی (Globalization Testing)
- تست قابلیت همکاری (Interoperability Testing)
- و تست های دیگر...

Non-Functional Testing

تست کارایی (Performance Testing)

تست تحمل (Endurance Testing)

تست بار (Load Testing)

تست حجم (Volume Testing)

تست مقیاس پذیری (Scalability Testing)

تست کاربرد پذیری (Usability Testing)

و تست های دیگر...

Maintenance Testing

- تست رگرسیون (Regression Test)

- تست نگهداشت (Maintenance Testing)

- تست نرم افزار فرآیند تایید و تصدیق یک سیستم کامپیوتری/برنامه کامپیوتری است تا بر اساس آن تصمیم بگیرید آیا آن برنامه با نیازمندی های مشخص شده مطابقت داشته یا خیر، و اینکه آیا نتایج مورد نظر را تولید می کند یا نه. در نتیجه این فرآیند، اشکالات را در محصول/پروژه نرم افزاری شناسایی می کند.

- تست نرم افزار برای ارائه یک محصول با کیفیت بدون هیچ مشکلی یا مشکل اساسی وارد عمل می شود. البته در نظر داشته باشید که طبق اصول تست نرم افزار هیچ برنامه بدون باگی وجود ندارد، و اصطلاح Bug Free به معنی این است که طی یک مدت زمان از پیش تعیین شده، باگی مشاهده نشده است، و سیستم به درجه مناسبی از بلوغ رسیده است.

۷ اصل تست نرم افزار

- این مهم است که در حین انجام تست نرم افزار شما بتوانید بدون انحراف از هدف، Test Result های مطلوبی بدست آورید. اما چگونه معلوم می شود که استراتژی درستی برای تست دنبال می کنید؟ به همین دلیل، شما باید به اصول اولیه تست پایبند باشید. در اینجا، هفت اصل اساسی تست ارائه شده است که به طور گسترده ای در صنعت نرم افزار مورد استفاده قرار می گیرند.

یک مثال

- برای درک این موضوع، یک سناریو را در نظر بگیرید که در آن فایل را از پوشه A به پوشه B منتقل می کنید.
- به تمام راه‌های ممکن که می توانید این را موضوع را تست کنید، فکر کنید.
- به غیر از سناریوهای معمول، می توانید شرایط زیر را نیز تست کنید:
- تلاش برای انتقال فایل هنگامی که باز است
- شما حق امنیتی برای قرار دادن فایل در پوشه B ندارید
- پوشه B بر روی درایو مشترک (Shared Drive) است و ظرفیت ذخیره‌سازی کامل است
- پوشه B یک فایل با همان نام دارد
- ...
- در واقع این لیست بی پایان است.

- مثلاً می‌توانید فرض کنید شما ۱۵ فیلد ورودی برای تست دارید، هر کدام دارای ۵ مقدار ممکن است، تعداد ترکیباتی که باید مورد تست قرار گیرند، برابر 5^{15} (پنج به توان ۱۵).
داریم.
- اگر شما کل ترکیبات احتمالی را تست کنید، زمان و هزینه‌های اجرایی پروژه به شکل بی‌حد افزایش می‌یابد. در اینجا نیاز به اصول و استراتژی خاصی برای بهینه‌سازی تلاش تست داریم.

اصل اول

• تست جامع ناممکن است

- بله! تست کامل قابل انجام شدن نیست در عوض، ما بر اساس ریسک برنامه به مقدار مطلوبی تست نیاز داریم.
- البته این یک مثال بسیار ساده بود. اگر شرایط مقداری واقعی تر شود تعداد حالات هم به صورت تصاعدی رشد می کند.
- حالا که فهمیدیم تست کامل ناممکن است، و باید در تست های خود ریسک را مد نظر داشته باشیم، سوال میلیون دلاری این است که چگونه این ریسک را تعیین می کنید؟
- برای پاسخ دادن به این بیایید یک تمرین انجام دهیم:
به نظر شما، کدام عملیات به احتمال زیاد باعث خراب شدن سیستم عامل شما خواهد شد؟؟؟!
- احتمالا باز کردن برنامه های مختلف در یک زمان جواب شما باشد
- بنابراین مثلا اگر شما این سیستم عامل را تست کنید، متوجه خواهید شد که این نواقص احتمالا در فعالیت های Multi Tasking یافت می شوند و باید کاملا تحت تست قرار بگیرند تا ما را به اصل بعدی یعنی خوشه بندی نواقص هدایت کنند.

اصل دوم

- خوشه‌بندی نقص
- خوشه‌بندی نقصی بیان می‌کند تعداد کمی از ماژول‌ها دارای بیشترین نواقص هستند. این همان کاربرد اصل پارتو برای تست نرم‌افزار است: تقریباً ۸۰٪ از مشکلات در ۲۰٪ از ماژول‌ها یافت می‌شوند.
- با تجربه، می‌توانید چنین ماژول‌های پر ریسکی را شناسایی کنید. اما این رویکرد مشکلات خاص خود را دارد.
- اگر تست‌های یکسان بارها و بارها تکرار شوند، در نهایت همان Test Case ها، باگ‌های جدید را پیدا نخواهد کرد.

اصل سوم

- **پارادوکس آفت کش ها (ضعف آفت کش ها)**
- استفاده مداوم از یک ترکیب آفت کش برای از بین بردن حشرات در حین کشاورزی، در طول زمان منجر به ایجاد حشرات مقاوم در برابر آفت کش مذکور می شود. بنابراین آفت کش ها بر آن حشرات ناکارآمد می شوند. همین امر در مورد تست های نرم افزاری نیز صادق است. اگر یک مجموعه از تست های تکراری مرتب انجام شوند، متد شما برای کشف نواقص جدید بی فایده خواهد بود.
- برای غلبه بر این موضوع، **Test Case** باید به طور منظم بررسی و مورد تجدید نظر قرار گرفته و **Test Case** های جدید و مختلف برای کمک به یافتن نواقص بیشتر اضافه شوند.
- تسترها به سادگی نمی توانند به تکنیک های تست موجود وابسته باشند. تستر باید به طور مداوم به دنبال بهبود متدهای موجود برای تست موثرتر باشد. اما حتی پس از این همه کار سخت در تست، شما هرگز نمی توانید ادعا کنید محصول شما بدون اشکال است.

اصل چهارم

- **تست نشان می دهد نقص وجود دارد**
- این اصل از تست بیان می کند که تست در مورد وجود نواقص صحبت می کند، نه در مورد عدم وجود آنها. یعنی تست نرم افزار، احتمال بروز نقص های ناشناخته باقی مانده در نرم افزار را کاهش می دهد، اما حتی اگر هیچ نقصی پیدا نکند، این به معنی عدم وجود نقص نیست.
- علاوه بر پاراگراف بالا همانطور که قبلا دیدیم تست کامل ناممکن است، بنابراین حتی اگر تمام تست های شما موفق هم باشد، به دلیل عدم کمال تست شما نمی توانید بگویید محصول شما فاقد نقص است.
- اما اگر حتی بسیار سخت کار کردید، و تمام اقدامات احتیاطی را انجام داده و محصول نرم افزار خود را ۹۹٪ بدون خطا ارائه دهید ممکن است باز هم این نرم افزار نیازمندی ها و الزامات مشتریان را برآورده نکند.

اصل پنجم

- عدم وجود خطا، استفاده از نرم افزار را تضمین نمی کند
- ممکن است نرم افزار شما ۹۹٪ بدون مشکل باشد، اما همچنان غیر قابل استفاده باشد. این موضوع می تواند در شرایطی رخ دهد که نرم افزار به طور کامل برای نیازمندی های غلط تحت تست قرار گرفته است و این یعنی نرم افزار به اشتباه پیاده سازی شده است. تست نرم افزار نه تنها برای یافتن نواقص است، بلکه بررسی این موضوع که نرم افزار به نیازمندی های کسب و کار پاسخ می دهد نیز جز وظایف تست است. عدم وجود خطا برای مناسب بودن نرم افزار صرفاً یک سفسطه است. این یعنی اگر سیستم غیر قابل استفاده ای ساخته شده باشد، دیگر پیدا کردن و رفع نواقص هم کمکی نکرده و نیازهای کاربر برآورده نمی شود.
- اصل بعدی راه حل این مشکل است.

اصل ششم

• تست زودهنگام

- تست باید در چرخه حیات توسعه نرم افزار در اسرع وقت شروع شود. بنابراین هرگونه نقص در نیازمندی ها یا مرحله Design در مراحل اولیه تشخیص داده می شود. رفع نقص در مراحل اولیه تست بسیار ارزانتر است. اما چگونه در اوایل باید تست را شروع کنیم؟ توصیه می شود یافتن باگ ها را زمانی که نیازمندی ها تعریف شد، آغاز کنید.

اصل هفتم

- تست وابسته به زمینه محصول است

- تست وابسته به زمینه است که اساسا به این معنی است که روش تست کردن سایت تجارت الکترونیک متفاوت از روش تست برنامه کسب و کاری موجود در فروشگاه‌هاست. تمام نرم‌افزارهای توسعه یافته یکسان نیستند. شما ممکن است از رویکردها، متدولوژی‌ها، تکنیک‌ها و انواع تست‌های متفاوت بر اساس نوع برنامه استفاده کنید. برای مثال تست هر سیستم POS در یک فروشگاه خرده فروشی متفاوت از تست دستگاه ATM است.

خلاصه ۷ اصل اساسی تست نرم افزار

- ۱- تست جامع ناممکن است
- ۲- خوشه بندی نقص
- ۳- پارادوکس آفت کش ها (ضعف آفت کش ها)
- ۴- تست نشان می دهد نقص وجود دارد
- ۵- عدم وجود خطا، استفاده از نرم افزار را تضمین نمی کند
- ۶- تست زودهنگام
- ۷- تست وابسته به زمینه محصول است

چه کسی تست می کند؟

- در صنعت IT، شرکت های بزرگ، تیمی برای بررسی نرم افزارهای تولیدی در چارچوب الزامات و فعالیت های شرکت دارند که این تیم، تیم تست می باشد.
- در اغلب موارد تستر در یکی از بخش های زیر قرار می گیرد:
 - - بخش تست نرم افزار
 - - بخش توسعه نرم افزار
- شرکت ها طراحی های متفاوت و وظایف و نقش های متفاوتی برای اشخاصی که تست نرم افزار انجام می دهند، قائل می شوند. این نقش ها عبارتند از : تستر نرم افزار، تضمین کیفیت نرم افزار (QA)، تحلیلگر و ...

نتیجه گیری

تست را می توان یکی از زیرمجموعه های مبحث کیفیت نرم افزار با نام "تضمین کیفیت" در نظر گرفت. در واقع تست نرم افزار به دنبال خطایابی و عیب یابی محصول، قبل از تحویل به مشتری است، یعنی اینکه هم توسعه دهندگان و هم کاربران نهایی بر روی یک نرم افزار کارآمد و قابل بکارگیری که پاسخگوی نیازمندی های تعریف شده باشد، هم نظر باشند.

منابع

- <http://testology.ir/>
- <http://tisten.ir>



با تشکر از همراهی شما دوستان

پاسخ به سوالات