اختبار جودة البرمجيات

د. فادي تركاوي

- تعدّ البرمجيّات العنصر الرئيسي في العديد من الأنظمة والأجهزة المنتشرة في حياتنا، وعلى الرغم من تأثير العديد من العوامل على وثوقيّة هذه البرمجيّات ونجاحها كالتصميم والتنفيذ الدقيقين، إلّا أنّ الاختبار هو الوسيلة الرئيسيّة التي تعتمدها الصناعة لتقييم البرمجيات التي لا تزال قيد التطوير

- بدايةً، تمكّنت الشركات البرمجية منذ فترةٍ ليست بطويلةٍ من تحمّل تكاليف توظيف مبرمجين لا يجيدون إجراء أيّ اختبارٍ لمشاريعهم، بالإضافة إلى مختبرين لا يعرفون أيّ شيءٍ عن البرمجة. ولم يكن من الضروري لصنّاع البرمجيات أن يعرف موظّفوهم المبادئ التقنيّة للاختبار أو حتّى تطوير البرمجيات. وبذلك كان اختبار البرمجيات في بداياته نشاطاً غيرَ تقنيّ ويتمّ النظر إليه على أنّه عملٌ إداري.

- *ومع تطوّر هندسة البرمجيّات وانتشار برمجيّاتها في حياتنا اليوميّة، ظهرت متطلّباتٌ صارمةٌ بخصوص وثوقيّة*

ما هي اختبارات جودة البرمجيّات؟

هذه البرمجيّات وصيانتها وأمنها. وكان لا بدّ للصناعة أن تتجاوب مع هذه المتطلّبات الجديدة بتطوير طرق اختبارها لهذه البرمجيّات، ممّا يعني زيادة الخبرة التقنيّة لمهندسي الاختبار بالإضافة إلى تعزيز أهميّة الاختبار وضرورته عند المبرمجين.

- تدعى عمليّة اختبار البرمجيات بـ:
  مراقبة جودة البرمجيات control quality software .
  ضمان جودة البرمجيات assurance quality software .

- الخطأ :Errorوهو أي خطأ قام به المهندس كنتيجةٍ عن سوء الفهم للمتطلّبات أو التصميم.
العيب :Faultوهو ظهور هذا الخطأ ضمن الكود، وندعوه غالباً) Bugبمعنى إصابة.(
الفشل :Failureوهو السلوك أو النتيجة غير الصحيحة والناتجة عن تنفيذ كود يحوي عيباً ما،
قد يكون هذا الفشل مباشراً)كأن ينهار النظام(أو قد يظهر لاحقاً خلال التنفيذ

- *هل يكون النظام خالياً من العيوب إن تجاوز كلّ اختباراته بنجاح؟*

للأسف لا، فقد تكون العيوب مختبئة ضمن أجزاءٍ نادرة التنفيذ من الكود، وبذلك تكون وظيفة الاختبار إثبات وجود العيوب وليس عدم وجودها وتأثيرها على النظام.

ولكن في حال قمنا بتشكيل مجموعة اختباراتٍ واسعةٍ تشمل كافة وظائف النظام، فاجتياز هذه المجموعة يعني أنّ نظامنا ذو جودةٍ عاليةٍ وقابلٌ للنشر.

١ تجميع كافّة وظائف وسلوكيّات النظام على اعتبار أنّها فضاء واسع.
٢ تقسيم هذا الفضاء إلى أجزاء مختلفة الوظيفة)أو التنفيذ.(
٣تكوين اختباراتٍ تماثل نتيجة كلّ جزءٍ من هذا الفضاء وتنفيذ هذه الاختبارات

مثال

- لنفرض أنّه لدينا برنامجٌ يقوم بإيجاد القاسم المشترك الأكبر لعددين x و y، فمن الممكن أن تكون أجزاء الفضاء في هذه الحالة:
اختبار حالةٍ شائعة مثل x= ٦ و y= ٩) القاسم المشترك الأكبر هنا = ٣).
اختبار كون أحد العددين هو القاسم المشترك الأكبر مثل x= ٢ و y= ٤) القاسم المشترك الأكبر = ٢).
اختبار عددين أوليّين مثل x= ٣ و y= ٥) القاسم المشترك الأكبر = ١).
اختبار كون أحد العددين مساوياً للصفر مثل x= ٩ و y= ٠) لا يوجد قاسم مشترك أكبر.)
اختبار كون أحد العددين سالباً مثل x= -٣ و y= ٩) القاسم المشترك الأكبر هنا هو العدد السالب.)
ويتضّح لنا من المثال السابق أنّ اختبارَ نظامٍ ما بالكامل هو أمرٌ مستحيل

أنواع الاختبارات

١. اختبار الوحدة :Testing Unitويهتمّ بالمفاهيم منخفضة المستوى من النظام، إذ يَجري الاختبار هنا ضمن الكود البرمجي ومن أجل كلّ نموذجٍ وكلّ عمليّة.
مثال: من أجل كلّ تابعٍ أو إجرائيّةٍ ()foo سنجد تابعاً لاختباره يدعى ()testFoo

٢. اختبار التكامل :Testing Integrationويهتمّ بضمان عمل جميع الأجزاء بشكلٍ صحيحٍ بعد دمجها معاً وخاصّةً عندما يعمل أكثرُ من مطوّرٍ واحدٍ على كتابة الكود.

٣. اختبار النظام :Testing Systemللتأكّد من عمل جميع الوظائف الكبيرة ضمن النظام وتطبيق جميع متطلّبات الزبون.

٤. اختبار القبول :Testing Acceptanceويجري عادةً من قبل المستخدمين للتأكّد من تلبية النظام لاحتياجاتهم.

٥. - اختبارات الكود:
-١اختبار الصندوق الأسود :Testing Box Blackهل سلوك النظام مطابقٌ لما جاء في المتطلّبات؟
-٢اختبار الصندوق الرمادي :Testing Box Greyهل سلوك النظام مطابقٌ لما جاء في المتطلبات مع وجود معرفة ضئيلة ببنية هذا النظام.
-٣اختبار الصندوق الأبيض :Testing Box Whiteاختبار جميع أجزاء الكود من عباراتٍ وتفرّعاتٍ وغيرها.

٦ . اختبار التراجع :Testing Regressionويقصد به إعادة اختبار البرمجيّة بعد أن تمّ تعديلها أو إصلاح أخطاء سبق وتمّ العثور عليها . ويتطلّب هذا النوع أكبر قدرٍ من الجهد وغالباً ما يتمّ إجراء العديد من جولات الاختبار التراجعيّة في الأنظمة الكبيرة، لأنّ أيَّ تعديلٍ بسيطٍ ضمن أحد أجزاء البرمجيّة قد يسبّب غالباً مشاكل في أماكن بعيدةٍ حتّى، وتكون وظيفة الاختبار التراجعيّ اكتشاف هذا النوع من الأخطاء .

٧ . الاختبارات المؤتمتة :Testing Automated إن تطبيق جميع الاختبارات السابقة يدوياً)وخصوصاً اختبارات الكود(أمرٌ قد يكون في غاية الصعوبة، ولكن من حسن الحظ، هناك العديد من الأدوات التي تقوم بتغطية وتعقّب الكود بأكمله . وتقوم هذه الأنظمة المؤتمتة بتوليد تقارير لإظهار النسبة المئوية المُنجَزة من اختبار الكود، كما أنّ بعضها يظهر لنا العبارة البرمجية التي تخضع حاليّاً للاختبار