

Capítulo 7: Pruebas

Víctor Iranzo

28 de mayo de 2018

1. Tipos de pruebas

El siguiente esquema es una modificación del cuadrante de Marick empleada para catalogar pruebas en distintos tipos:

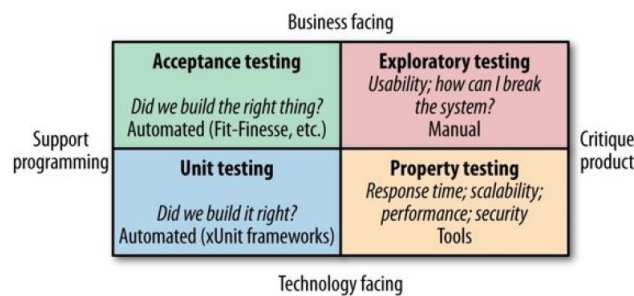


Figura 1: Modificación del cuadrante de Marick hecha por L.Crispin y J.Gregory en su libro Agile Testing.

Las pruebas situadas en la mitad inferior están relacionadas con la tecnología empleada, como son las pruebas unitarias o las de rendimiento, que deben ser implementadas por los desarrolladores y que se pueden automatizar. Las pruebas situadas en la mitad superior están orientadas a que los stakeholders conozcan cómo funciona el sistema y que no guardan relación con aspectos tecnológicos. Entran dentro de esta mitad las pruebas de aceptación o las pruebas manuales.

La proporción necesaria de cada tipo de test depende del sistema a desarrollar. La tendencia actual es en favor de los tests de pequeña escala y automatizados. En un sistema basado en microservicios, son esenciales las pruebas de este tipo ya que validarán que no se despliegue a producción código defectuoso.

Podemos clasificar las pruebas en los microservicios en 3 tipos según su alcance: unitarias, de servicios y de extremo a extremo. Estos términos pueden llevar a discusión: por ejemplo, en una prueba unitarias se puede incluir para algunas personas la validación de diferentes clases o funciones mientras que para otras no.

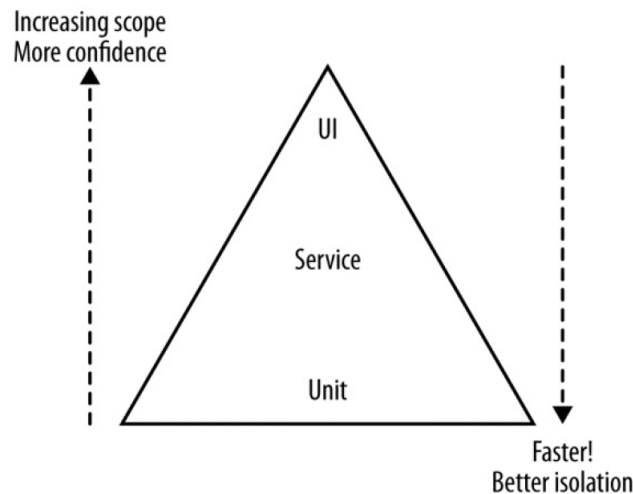


Figura 2: Pirámide de pruebas diseñada por Mike Cohn en su libro *Succeeding with Agile*.

1.1. Pruebas unitarias

Son pruebas que validan una única función. Se pueden generar fruto de procesos de desarrollo como el test-driven development (TDD). En general, se prefiere tener un gran número de este tipo de pruebas por su rapidez, porque pueden ayudar a la refactorización del código y porque es donde mayor cantidad de defectos se suele capturar.

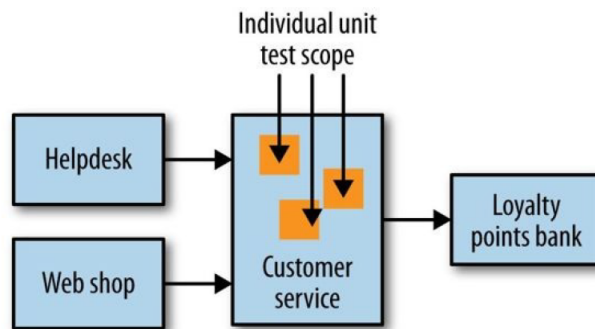


Figura 3: En las pruebas unitarias no se ejecuta ningún servicio, sino que se prueban métodos específicos de ellos.

1.2. Pruebas de servicios

En estas pruebas se verifica la funcionalidad de los servicios sin emplear las interfaces de usuario. Cada prueba verifica una de las funcionalidades que el servicio expone. Se pretende comprobar la separación entre los servicios y para solventar las dependencias que el servicio bajo pruebas tiene sobre otros, se reemplazan los servicios colaboradores por stubs o fakes.

Este tipo de pruebas pueden ser igual de rápidas que las unitarias siempre que no se tengan que emplear un gran número de infraestructuras como bases de datos o redes.

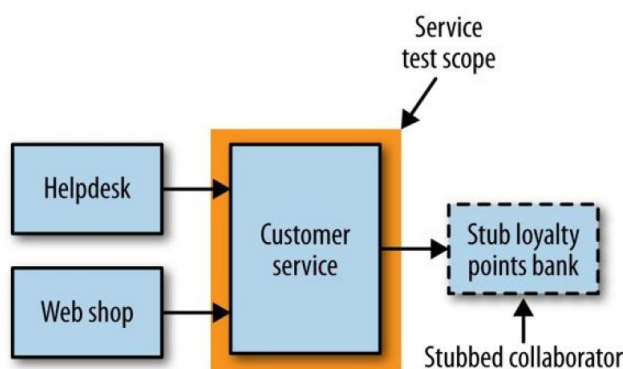


Figura 4: Las pruebas de servicio se ejecutan sobre un único servicio y emplean fakes para reemplazar a los colaboradores de este.

1.3. Pruebas de extremo a extremo

Son pruebas que se ejecutan sobre todo el sistema. Cubren gran parte de código, por lo que su correcta ejecución dan mucho grado de confianza.

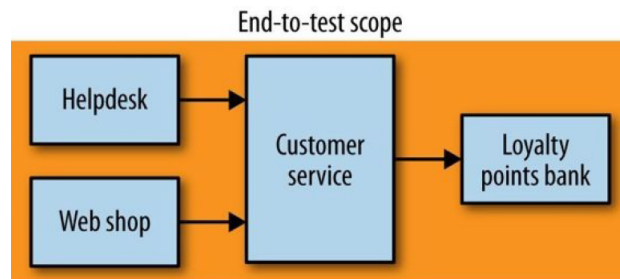


Figura 5: Las pruebas de servicio se ejecutan sobre un único servicio y emplean fakes para reemplazar a los colaboradores de este.

1.4. Pruebas dirigidas por el consumidor

2. Balance de pruebas a realizar

3. Pruebas frágiles y escamosas

4. Pruebas antes de producción

4.1. Despliegues verde y azul

4.2. Canary releases

4.3. Tiempo medio entre fallos y tiempo medio entre reparaciones

5. Pruebas de requisitos no funcionales

5.1. Pruebas de rendimiento