

PRINCIPIOS DE TESTING ÁGIL

INTRODUCCIÓN

En el desarrollo ágil el equipo entero toma responsabilidades por entregar software de alta calidad que satisfaga al usuario y haga al negocio más rentable.

Los testers ágiles toman un enfoque mucho más integrado y orientado a equipos que los testers de proyectos tradicionales con un flujo de cascada. Ellos adoptan sus habilidades y experiencia a cada equipo y proyecto.

PRINCIPIOS

La siguiente es una lista de principios de testing ágil que se deriva de los principios incluidos en el Manifiesto Ágil, que definen cómo encarar el desarrollo de software.



MANTENERLO SIMPLE

El testing ágil significa hacer los tests más simples posibles que verifiquen que una pieza de funcionalidad existe o que se han conseguido los estándares de calidad del cliente. Los testers y otros miembros del equipo deberían proveer información a los clientes y ayudarlos a considerar todos los aspectos de calidad, incluyendo requisitos no funcionales



COMUNICACIÓN CARA A CARA

El desarrollo ágil depende de una colaboración constante. Cuando un tester ágil sospecha una suposición oculta o un requerimiento mal entendido, este va a conseguir que el cliente y los desarrolladores hablen al respecto.



AUTO ORGANIZARSE

Los tester ágiles son parte de un equipo multidisciplinario autoorganizado. Cuando tienes a alguien con múltiples habilidades y diferentes conocimientos para atacarlo, cualquier problema de testing será más fácil de resolver.



PRACTICAR MEJORAS CONTINUAS

Las iteraciones cortas facilitan probar cosas nuevas durante poco tiempo y ver si vale la pena adoptarlas a largo plazo. Las Sprint Retrospective son claves para permitir a los equipos usar la experiencia de ayer para mejorar el trabajo de mañana. Los testers ágiles usan esta oportunidad para introducir asuntos relacionados al testing de modo que el equipo ayude a solucionarlos.



RESPONDER AL CAMBIO

En el desarrollo ágil los clientes saben que pueden obtener los cambios que quieran ya que ellos controlan las prioridades. Responder al cambio es un aspecto fundamental de las prácticas ágiles, y uno de los más difíciles de adoptar para los testers.



ENTREGAR VALOR AL CLIENTE

El desarrollo ágil se trata de entregar valor en una cantidad pequeña de releases que proveen exactamente la funcionalidad que el cliente haya priorizado más recientemente, lo que implica limitar el alcance. Los testers ágiles poseen una vista general y se mantienen enfocados en entregar la funcionalidad más crítica.



TENER CORAJE



Para los equipos nuevos en el mundo del desarrollo ágil es normal experimentar miedo al poseer una lista de preguntas que necesitan ser respondidas. Los testers ágiles necesitan coraje para permitirse fallar rápido y ser capaces de aprender de esas fallas.



ENFOCARSE EN LAS PERSONAS

Los proyectos triunfan cuando a las personas se les permite hacer su mejor trabajo. Los testers ágiles saben que agregan un valor único a sus equipos, y los equipos de desarrollo saben que son más exitosos cuando incluyen a gente con experiencia y habilidades en testing.



PROVEER FEEDBACK CONTINUO



La retroalimentación es una manera de superar los obstáculos. El tester ágil ayuda al Product Owner o cliente a articular los requerimientos de cada Story en forma de ejemplos y tests. Luego trabaja con el equipo para transformar esos requerimientos en tests ejecutables.

CONCLUSIÓN

Ahora más que nunca es necesario probar continuamente, ya que es una de las claves para mantener la calidad de los productos. Por ello, debe haber una evolución hacia el "testing ágil", en lugar de un testing tradicional, donde este es una etapa más dentro del desarrollo y no aporta un feedback continuo. Además el tester ágil, va a ser una figura muy importante, que aporte mucho valor a un equipo; ayudando en la definición del producto, enfocando el desarrollo y ayudando a la automatización de pruebas funcionales.



Para mas información:

https://docs.google.com/document/d/1rrldjcd1XO0mbc3DE6cDyta8arOuy.MNBX7OQEjuN_MM/edit?usp=sharing



UTN FRC - Ingeniería en Sistemas
Ingeniería de Software - 4K4

Barella Martina, Moyano Gonzalo, Mungi Simon,
Salinas Alercia Gerardo, Ulbrich Danilo