

TDD/BDD

Temario

1. Conceptos previos
 - 1.1. Introducción a las metodologías ágiles
 - 1.2. Prácticas esenciales en XP
 - 1.3. Test Driven Development
 - 1.4. Pruebas de caja negra y caja blanca
 - 1.5. Probando que funcione el software
 - 1.6. Probando las cualidades del software
 - 1.7. Calidad o ¿Por qué no se puede codificar sin errores?
 - 1.8. Imposible probar todo
 - 1.9. Los requerimientos, casos de uso y su impacto en pruebas
 - 1.10. Principios de diseño de software
 - 1.10.1. Diseño simple
 - 1.10.2. Dile, no preguntes
 - 1.10.3. SOLID
 2. Un vistazo a la gestión de pruebas
 - 2.1. ¿Quién debe hacer, Qué, Cuándo?
 - 2.2. Roles y responsabilidades
 - 2.3. El esfuerzo de prueba y el factor de certeza
 3. Métodos de prueba
 - 3.1. MITs Most Important Test
 - 3.2. MIM Most Important Metrics
 4. Inventario de pruebas
 5. Análisis de Datos
 - 5.1. Valores frontera
 - 5.2. Como reducir los datos
 6. Análisis de rutas y bifurcaciones
 - 6.1. ¿Qué es la complejidad ciclomatica?
 - 6.2. Divide y vencerás
 - 6.3. De casos de uso a casos de prueba
 7. Metodología Test Driven Development (TDD)
 - 7.1. Requisitos
 - 7.2. Ciclo de desarrollo conducido por pruebas
 - 7.3. Ventajas
 8. JUnit
 - 8.1. Tests y Suites
 - 8.2. Anotaciones
 - 8.3. Asertos
 - 8.3.1. Hamcrest: Librería para predicados
 - 8.4. Testeando excepciones
 - 8.5. Restricciones Temporales
 - 8.6. Test paramétricos
 - 8.7. Teorías
16. Uso de Mocks, Fakes y Stubs
 - 16.1. Mockito
17. Cobertura de las pruebas (Code Coverage)
 - 17.1. EclEmma

- 17.2. Cobertura
- 18. Calidad del código
- 19. Refactorizaciones
 - 19.1. Principios de refactorización
 - 19.2. Malos olores en el código
 - 19.3. Construcción de pruebas
 - 19.4. Hacia un catálogo de Refactorizaciones
 - 19.5. Composición de los métodos
 - 19.6. Traslado de funciones entre objetos
 - 19.7. Organizando los datos
 - 19.8. La simplificación de las expresiones condicionales
 - 19.9. Simplificación de la realización de Métodos
 - 19.10. Tratar con la generalización
 - 19.11. Grandes Refactorizaciones
 - 19.12. Reutilización
 - 19.13. Herramientas de refactorización: Eclipse
- 20. Pruebas de Aceptación (ATDD/BDD)
 - 20.1. Definiendo los requisitos de su proyecto
 - 20.2. Configurar su entorno para pruebas de aplicaciones web
 - 20.3. Escribe tus historias en un lenguaje fácil de usar
 - 20.4. La anatomía de un escenario textual
 - 20.5. Escribir y anotar los métodos
 - 20.6. Pasando parámetros a método
 - 20.7. Compartir datos entre métodos
 - 20.8. Organizando tus escenarios
 - 20.9. Configuración y Limpieza
- 21. Herramientas BDD
 - 21.1. Concondion
 - 21.2. Cucumber
 - 21.3. Fitnesse