

XVI JORNADAS ODOO

VALENCIA

Organizan:




ASOCIACIÓN
ESPAÑOLA
DE ODOO



Automatizar pruebas y no morir en el intento (o sí)

Maximiliano Mannise @mmannise



TESTING | PRUEBAS DE SOFTWARE
CARACTERÍSTICAS BAJO TEST
FOCO DEL TEST
DOMINIO DE APLICACIÓN
TIPO DE DISPOSITIVOS
INFRAESTRUCTURA

Informático de profesión, apasionado por el mar. Ya tengo canas en la barba.
En testing ... de casualidad. Echando raíces en Valencia, trabajo en el ITI
(Instituto Tecnológico de Informática) y tengo el gusto de dirigir el
departamento de calidad del software

Maxi
@mmannise





Servicios avanzados
en TI & TIC

Investigación

@ITI_TIC



ITI INVESTIGATE
TO INNOVATE

Innovación

Difusión

Transferencia



ES MI EXPERIENCIA, NO VERDAD UNIVERSAL



ACTIVA TÚ SENTIDO CRITICO, NO ME CREAS



EXPERIMENTA, ENCUENTRA TÚ SOLUCIÓN



Autómata

“Máquina dotada de un mecanismo que le permite moverse, en particular la que imita la figura y movimientos de un ser animado, normalmente humano”

Automatización

“Acción y efecto de automatizar o hacer automático algo”

Automatización de Test

“Acción y efecto de automatizar: la creación del entorno, juego de datos, ejecución de casos de test y comprobación de resultados”

Filón de Bizancio 280 a.C

Autómata “El sirviente” Museo Kotsanas

Automatización de test

“Acción y efecto de automatizar: la creación del entorno, juego de datos, ejecución de casos de test y comprobación de resultados”

**I HATE
NOTHING
ABOUT U**



¿Promesas vacías?

Beneficios

1. Red de seguridad
2. Mayor cobertura de test
3. Mejora tiempos de entrega
4. Menos errores humanos
5. Integrado en motor de CI
6. ~~Reemplaza~~ reduce la cantidad de TM_(test manual)
7. Libera tiempo a los testers
8. Resultado de ejecución test inmediato
9. Selección de CT según riesgo, impacto ...
- 10....

**Hay tanto
para hacer
... y tan
diverso**

Clasificación

- Funcional | no funcional | riesgo
- Foco de test
- Unitario | Integración | API Test | UI | Rendimiento | UAT | Exploratorio | Smoke Test | Regresión | Despliegue
- Tipo de dispositivo
- Escritorio | Web | App | Embebido | Wearable
- Dominio de aplicación
- SW propósito general | industria | empresa | seguridad crítica
- Infraestructura
- Cloud | On-Premise | distribuida ...

No es
*“ponerse a
automatizar”*

Objetivo: automatización

1. Definir ALCANCE (qué)
2. Definir ESTRATEGIA (cómo)
3. Seleccionar PERFILES (quiénes)
4. PoC HERRAMIENTA
5. PoC INFRAESTRUCTURA
6. Definir IMPLANTACIÓN (proyecto)
7. Manos al teclado ...
 1. Preparación
 2. Especificación
 3. Ejecución
8. Analizar > Regresa a ... (dónde haga falta)

Todo un
desafío ...
toda una
inversión

Fácil NO es

1. Recursos necesarios
2. Habilidades y competencias
3. Impacto en día a día
4. Cultura organizacional
5. Sostenible / mantenible en el tiempo

MIND THE GAP

Mind the gap!

¡Cuidado!

1. No contar con un FRAMEWORK
2. Calidad de los datos
3. Uso de entornos “inestables”
4. No tratar al código de CT como código
5. Insistir con una herramienta inadecuada
6. Dependencias de tiempo (fechas)
7. No reutilizar / copy + paste
8. Automatizar todo
9. Subestimar en general ... todo
10. Casos de test basados en otros casos de test* (no es taaaan grave)
11. ...

Mind the gap!

~~¡Odio~~ Amo testing!

1. Valor de test no percibido
2. Hay mucho MUCHO por hacer
3. Tiempo: nunca hubo, no hay y no habrá
4. No es fácil, se requieren habilidades y perfiles muy específicos
5. Los cambios son continuos y lo serán aún mas
6. Release vs Calidad vs Tiempo

Mind the gap!

Infraestructura

1. Herramientas de automatización
2. Creación, mantenimiento juego de datos
3. Entornos de despliegue y ejecución
4. Dispositivos necesarios
5. Espacio físico
6. Motor de integración continua
7. Gestión de resultados, logs, etc

Mind the gap!

Resultados

1. Visualización y presentación de resultados
2. Retorno de la inversión*
3. Falsos positivos (susto)
4. Falsos negativos (peligro)
5. Logs y resultados post-ejecución
6. Cobertura funcional
7. Cobertura de código

Una apuesta segura



Una apuesta segura

Receta

1. ¿Por dónde comenzar?
2. Tratarlo como un proyecto (ágil)
3. Estrategia de automatización
4. Condiciones necesarias
5. Selección y configuración de herramientas
6. Infraestructura: dispositivos, server, bbdd
7. Automatizar > Automatizar > Automatizar
8. Visibilidad (métricas y retorno inversión)

Una apuesta segura

¿Por dónde comenzar?

1. Analizar madurez de la organización
2. Conseguir apoyos* necesarios
3. Definir objetivos, realistas, medibles
4. Definir alcance
5. Formar equipo
6. ¡Tenemos proyecto!

Una apuesta segura

Tratarlo como un proyecto (ágil)

1. Decisión: black-ops vs trinchera
2. Black-ops: proyecto “independiente” acotado en el tiempo (sprints)
3. Trinchera: se integra en los equipos actuales
4. Definir test automation backlog
5. Establecer capacidad equipo
6. 3 roles + 3 artefactos = 6 eventos

Una
apuesta
segura

Estrategia de automatización

1. Establecer claramente ¿Para qué?
 1. Seguridad durante el desarrollo
 2. Eficiencia en regresión
2. Cobertura
 1. funcional basada en riesgo / uso
 2. Cobertura tipos de test (pirámide testing)
3. Necesidades infraestructura, herramientas
4. Formación equipo (aplanar curva aprendizaje)
5. Restricciones (legales, técnicas, etc.)
6. Definir hoja de ruta, orden y dependencias

Una apuesta segura

Condiciones necesarias

1. ¿Contamos con un framework?
2. ¿La arquitectura soporta automatización?
3. ¿Se pueden atacar las capas?
4. ¿Tenemos los recursos necesarios?
5. ¿Las personas están disponibles?
6. ¿Se pueden comprobar los resultados?
7. ... la lista sigue por cada componente de arquitectura, infraestructura, personas

**Una
apuesta
segura**

Tener muy en cuenta las restricciones y necesidades de los puntos anteriores

Selección herramientas

1. Si no pruebas (PoC) no la vas a encontrar
2. Ninguna herramienta es totalmente “gratis”
3. Ninguna herramienta se ajustará 100% a tus necesidades
4. No te cases
5. No te obsesiones

Una apuesta segura

Infraestructura

1. Presupuesto para infraestructura
2. Entorno en exclusiva
3. Juego de datos ofuscados
4. Creación y destrucción de datos
5. Dispositivos emulados y físicos
6. Automatizar vs Manual
7. Motor de integración continua

Una
apuesta
segura

Automatizar (N)

1. Un caso a la vez
2. Comenzar por casos simples
3. Manual >> Automático (regresión)
4. Enriquecer suite de test SIEMPRE
5. Seguir estrategia (no lo que te guste)
6. Si es complejo ... parar ... decidir
7. Los CT automáticos es código
8. Usar técnicas caja blanca de test

Una apuesta segura

Desde el comienzo del proyecto de automatización hay que dar visibilidad en todos los niveles

Visibilidad

1. Proceso de integración continua
2. Historial de resultados casos de test
3. Poder navegar en los resultados
4. Información para investigar problemas
5. Cobertura funcional
6. Cobertura técnica
7. Crecimiento suite de test por sprint (velocidad)
8. Retorno de la inversión
9. Cumplimos (o no) los objetivos

Una apuesta segura

Receta

1. Visibilidad (métricas y retorno inversión)
2. ¿Por dónde comenzar?
3. Tratarlo como un proyecto (ágil)
4. Estrategia de automatización
5. Condiciones necesarias
6. Selección y configuración de herramientas
7. Infraestructura: dispositivos, server, bbdd
8. Automatizar > Automatizar > Automatizar

**Nuestra mejor arma
al alcance de todos**

WAKAMITI

CREAR SOFTWARE ES EMOCIONANTE,
PERO SABER QUE ESTÁ BIEN HECHO

ES OTRO NIVEL



ITI INVESTIGATE
TO INNOVATE

framework

Wakamiti

1. Framework de automatización de test
2. Filosofía BDD (Behavior Driven Development)
3. Uso de gherkin y cucumber
4. Visualización de resultados
5. Aplana curva de aprendizaje
6. Perfiles “tradicionales” de QA
7. Código abierto
8. Comunidad en crecimiento

framework

Demo 2'18''

1. Definición de casos de test (alto nivel)
2. Implementación de caso de test (técnico)
3. Ejecución
4. Caso OK
5. Caso KO
6. Presentación de resultados

EXPLORADOR ...

EDITORES ABIERTOS

SPRING... [+] [x] [↺] [↻]

- > .vscode
- > kukumo
 - > data
 - > features
 - ! kukumo.yaml
 - J application-kukumo....
 - i readme.md

ESQUEMA

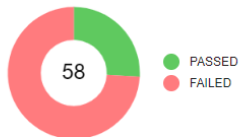
LÍNEA DE TIEMPO



- Mostrar todos los comandos ⌘ ⌘ P
- Ir al archivo ⌘ P
- Buscar en archivos ⌘ ⌘ F
- Iniciar depuración F5
- Alternar terminal ^ `

Resultado ejecución

Feature Results



Scenario Results



Errors By Type



Run info

Wakamiti version	1.7.0
Execution start	May 23, 2023, 7:30:29 AM
Execution end	May 23, 2023, 7:46:14 AM
Total duration	15m 7s 408ms

Característica: Obtener el registro de llamadas de un usuario

Como: usuario registrado

Quiero: consultar el registro de llamadas de un usuario

Para: conocer la cantidad de llamadas o AutoOn realizados


❌ ⌚ 21s 984ms 📄 24 | ✅ 16 ❌ 8

@ID-get_apiV1CallregistryParticipant-1 @Fermex @CallManager.API @call-registry-controller

Escenario: Solicitar el registro de llamadas: Todas

✅ ⌚ 4s 936ms 📄 7 | ✅ 7

Resultado detallado (html)

Característica: Obtener el registro de llamadas de un usuario 

Como: usuario registrado

Quiero: consultar el registro de llamadas de un usuario

Para: conocer la cantidad de llamadas o AutoOn realizados

  21s 984ms  24 |  16  8



Escenario: Solicitar el registro de llamadas: Todas

  4s 936ms  7 |  7

Antecedentes:

 Dado  7ms 



 Y que  8ms 



 Dado los siguientes parámetros de búsqueda: 12ms 

name	value
appToken	`\${blue.userToken}`
callRegistryType	all

 Cuando se realiza la búsqueda del registro de llamada 4s 687ms 

 Entonces el código de respuesta HTTP es 200 58ms 

 Y el entero del fragmento de la respuesta 'findAll{ fex -> fex.isAutoon == false }.size()' es mayor o igual que 0 111ms 

 Y el entero del fragmento de la respuesta 'findAll{ fex -> fex.isAutoon == true }.size()' es igual a 0 50ms 

Evolución temporal

Wakamiti 

EDIT DASHBOARD

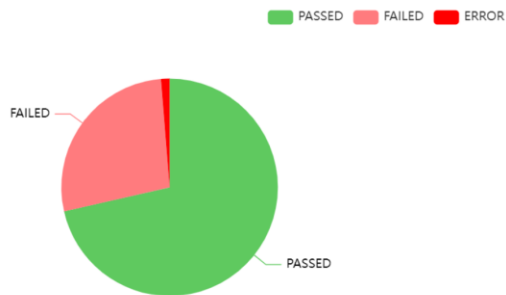
Nº Servicios web

54

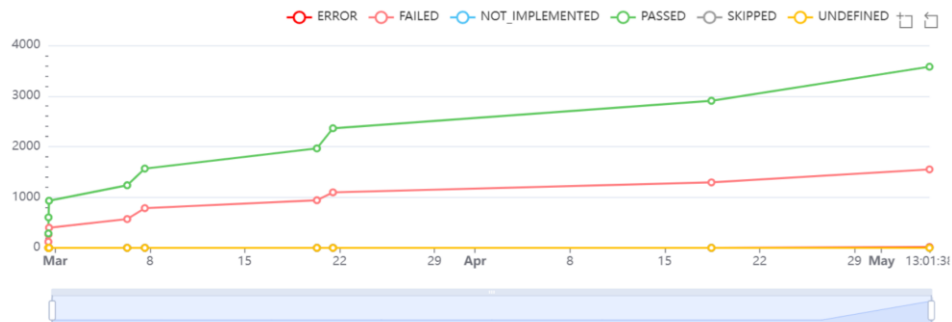
Nº Casos de casos de test

942

Resultados última ejecución



Evolución de los casos de test



Rendimiento

Estado por servicio web de la última ejecución

Search 54 records...

lastExecution	project	feature	Scenarios	Passed%	Not Implemented%	Skipped%	Undefined%	Failed%	Error%
2023-05-04	PHP	Obtener un listado de clientes por país (getClientesPorPaís)	558	92	0	0	0	7	0
2023-05-04			289	39	0	0	0	60	0
2023-05-04			256	67	0	0	0	32	0
2023-05-04			216	71	0	0	0	28	0
2023-05-04			192	65	0	0	0	34	0
2023-05-04			189	68	0	0	0	31	0
2023-05-04			189	58	0	0	0	41	0
2023-05-04			189	57	0	0	0	42	0
2023-05-04			153	77	0	0	0	22	0

Tiempos de respuesta



**Todo muy lindo peeeero ...
¿Qué hay de Odoo?**

Testing en Odoo

Automatización

1. ¡Si se puede!
2. Existe un **framework** / guías
3. Se pueden usar **herramientas externas** para automatizar
4. ¿PoC con **Wakamiti**? (APIs / Backend)

Testing Odoo: referencias

1. Odoo's Documentation

<https://www.odoo.com/documentation/16.0/es/developer/reference/backend/testing.html>

Python unit tests | JS unit tests | Tours (see Integration Testing)

2. Odoo's Web Framework Tutorial

https://www.odoo.com/documentation/16.0/developer/tutorials/master_odoo_web_framework/05_testing.html#

Integration testing | Unit testing a Component | Unit testing our gallery view | Examples

3. Odoo's Test Framework Learn Best Practices

<https://www.youtube.com/live/JEIscps000Q?feature=share>

Odoo's Test Framework to learn some best practices for testing Odoo modules, such as how to patch a model for a test, and how to test cases that make the current transaction fail

https://www.odoo.com/documentation/16.0/es/developer/tutorials/unit_tests.html

Recap

Mensaje final

1. Condiciones **muy buenas** para automatizar pruebas
2. **No es fácil**, pero es necesario
3. PoC de **herramientas**, siempre
4. **Estrategia** antes que la técnica
5. Cuidado con las **expectativas**

ORGANIZADORES:



VALENCIA

XVI JORNADAS ODOO

PATROCINADORES **ORO**



PATROCINADORES **PLATA**



COLABORADORES:

