

## **Caso práctico QA evolutivo zara.com**

### **Producto**

Formas parte de un equipo de producto que desarrolla nuevas funcionalidades para un sistema de e-commerce del sector textil.

Dicha plataforma de venta online permite la venta en 10 mercados diferentes, repartidos entre EEUU,

Europa, Asia y América del Sur.

### **Equipo**

Formas parte de un equipo de uno de los equipos de producto, a los que llamamos verticales de negocio, como qa evolutivo, el cuál trabaja con metodología Scrum con sprint de una semana.

Esta vertical está compuesta por

- 1 product owner
- 1 proxy product owner: project manager
- 1 delivery manager
- 5 desarrolladores IOS
- 5 desarrolladores Android
- 4 desarrolladores backend
- 4 desarrolladores android
- 5 QAs evolutivos, entre los que estás tú

### **Tus funciones como QA evolutivo**

- Definición de los criterios de aceptación de esa HU
- Escritura de esos criterios de aceptación en BDD
- Ejecución manual

### **HU**

Se ha definido una nueva HU cuyo enunciado es el siguiente:

“Modificar el proceso de login para que además de usuario y contraseña sea obligatorio aceptar un

## **check de política de privacidad”**

**En base a la información aportada, se pide:**

### **1. Definición de los criterios de aceptación de esa HU**

1. El formulario de login debe incluir un nuevo campo de verificación de política de privacidad.
2. El campo de verificación de política de privacidad debe estar marcado por defecto.
3. El usuario debe poder desmarcar el campo de verificación de política de privacidad.
4. Al intentar realizar el login sin marcar el campo de verificación de política de privacidad, se debe mostrar un mensaje de error indicando que la aceptación de la política es obligatoria.
5. Al marcar el campo de verificación de política de privacidad y proporcionar el usuario y contraseña correctos, el usuario debe poder iniciar sesión correctamente.
6. Al iniciar sesión correctamente, el usuario debe ser redirigido a la página principal o a la última página visitada antes de iniciar sesión.
7. Al marcar el campo de verificación de política de privacidad y proporcionar un usuario o contraseña incorrectos, se debe mostrar un mensaje de error indicando que las credenciales son inválidas.
8. La funcionalidad de aceptar la política de privacidad debe ser compatible con los navegadores y dispositivos principales utilizados por los usuarios.
9. La funcionalidad de aceptar la política de privacidad debe ser probada en los diferentes mercados (EEUU, Europa, Asia y América del Sur) para asegurar su correcto funcionamiento en cada uno de ellos.

### **2. Escritura en BDD Gherkin de los casos de test para validar esa HU**

Adjunto los casos en inglés porque es como los defino yo en mi puesto de trabajo actual:

Casos de prueba para validar la HU "Modify the login process to make acceptance of a privacy policy check mandatory":

Feature: Modify login process with mandatory privacy policy acceptance

Scenario: Display error message if privacy policy is not accepted

Given I am on the login page

When I enter my valid username and password

And I do not check the privacy policy acceptance checkbox

And I click the login button

Then I should see an error message stating that I must accept the privacy policy

Scenario: Successfully log in with privacy policy acceptance

Given I am on the login page

When I enter my valid username and password

And I check the privacy policy acceptance checkbox

And I click the login button

Then I should successfully log in and be redirected to the main page

Scenario: Display error message if credentials are invalid

Given I am on the login page

When I enter an invalid username or password

And I check the privacy policy acceptance checkbox

And I click the login button

Then I should see an error message stating that the credentials are invalid

Scenario: Verify compatibility across different browsers and devices

Given I am on the login page

When I enter my valid username and password

And I check the privacy policy acceptance checkbox

And I click the login button

Then I should see the main page or the last visited page before logging in

And the privacy policy acceptance functionality works correctly across different browsers and devices

Scenario Outline: Verify functionality in different markets

Given I am on the login page

When I enter my valid username and password

And I check the privacy policy acceptance checkbox

And I click the login button

Then I should see the main page or the last visited page before logging in

And the privacy policy acceptance functionality works correctly in the "<market>" market

Examples:

| market |

| USA |

| Europe |

| Asia |

| South America |

**3. Teniendo en cuenta que esta HU tiene implementación tanto de Web, como de iOS, Android y backend, comenta qué herramientas utilizarías para validar esta HU en cada disciplina.**

Para validar la HU de forma manual en cada disciplina, utilizaría las siguientes herramientas:

**1. Validación Web:**

- Navegadores web: Chrome, Firefox, Safari y Edge, para probar la funcionalidad en diferentes entornos de navegación.

- Herramientas de desarrollo integradas en los navegadores, como las "DevTools" de Chrome y las herramientas de desarrollador de Firefox, para inspeccionar elementos, verificar la consola de errores y realizar pruebas de rendimiento.

**2. Validación iOS:**

- Dispositivo iOS: realizar pruebas directamente en un dispositivo físico, utilizando Xcode, o algún simulador.

**3. Validación Android:**

- Dispositivo Android: Realizar pruebas directamente en un dispositivo físico o mediante algún emulador.

#### 4. Validación Backend:

- API testing tools: Utilizaría herramientas como Postman para enviar solicitudes HTTP directamente a la API backend y verificar las respuestas.
- Pruebas de integración: Realizaría pruebas de integración donde se envían solicitudes al backend y se validan las respuestas, asegurándome de que los cambios en la funcionalidad del login no hayan afectado otras partes del sistema.

#### **4. Como QA engineer dentro de un equipo, compártenos que ceremonias y reuniones crees que existirán durante la semana que dura el sprint.**

1. Sprint Planning: Al comienzo del sprint, el equipo se reunirá para definir y acordar los objetivos y las historias de usuario que se abordarán durante el sprint. Se discutirán las tareas necesarias, los criterios de aceptación y se estimará el esfuerzo requerido para completarlas.
2. Daily Scrum: Todos los días, el equipo se reunirá brevemente para compartir actualizaciones rápidas sobre el progreso individual, cualquier impedimento o bloqueo que se esté enfrentando y coordinar las actividades del día.
3. Sprint Review: Al final del sprint, se llevará a cabo una reunión de revisión del sprint para demostrar y revisar las funcionalidades completadas durante el sprint. El equipo presentará el trabajo realizado y se recopilarán los comentarios y sugerencias de los stakeholders, incluido el Product Owner.
4. Sprint Retrospective: Después de la revisión del sprint, el equipo realizará una retrospectiva para reflexionar sobre el sprint que acaba de finalizar. Se discutirán los aspectos positivos y negativos, se identificarán oportunidades de mejora y se establecerán acciones para implementar en futuros sprints.

Además de estas ceremonias principales, también es posible que se realicen otras reuniones y actividades adicionales según las necesidades del equipo y del proyecto, como:

- Reuniones de refinamiento de historias de usuario: Estas reuniones se realizan para revisar y refinar las historias de usuario futuras antes de que se agreguen a un sprint.
- Reuniones ad hoc con el Product Owner: Para aclarar dudas sobre los requisitos, discutir cambios o prioridades, y recibir retroalimentación sobre la funcionalidad desarrollada.

- Reuniones de revisión de calidad: Para analizar métricas de calidad, revisar los resultados de pruebas y evaluar el estado general de la calidad del producto.
- Sesiones de colaboración y coordinación con otros equipos: Para garantizar una integración fluida y colaborar en la resolución de problemas o dependencias entre equipos.

**5. Como último paso, nos gustaría incorporar la validación de este nuevo caso de uso a nuestro plan de regresión automática.**

• **¿Qué herramientas usarías para automatizar la parte de Web, iOS, Android y la parte backend?**

1. Automatización de pruebas web:

- Selenium WebDriver

2. Automatización de pruebas en iOS:

- XCTest o Appium.

3. Automatización de pruebas en Android:

- Appium

4. Automatización de pruebas en el backend:

- Postman

• **¿Qué herramientas de CI/CD crees que se podrían usar para programar el lanzamiento de la regresión de forma planificada y manual?**

- Jenkins o GitLab.

**6. El equipo de Desarrollo nos entrega un día tarde una HU prioritaria para el Producto. ¿Que estrategia tomarías en caso de que veas que no puedes llegar a probarla?**

Realizaría las siguientes acciones:

1. Comunicación con el equipo: Lo primero que haría es comunicar de inmediato al equipo de Desarrollo y al Product Owner sobre la situación y la falta de tiempo para realizar las pruebas. Es importante mantener una comunicación abierta y transparente para que todos estén al tanto de la situación y puedan tomar decisiones informadas.

2. Priorización de las pruebas: Evaluaría la importancia y el impacto de la Historia de Usuario en cuestión. Si es crítica para el funcionamiento del producto o tiene un alto riesgo asociado, buscaría la manera de ajustar las pruebas existentes o reasignar recursos para asegurarme de que se realice al menos una prueba básica para verificar su funcionamiento mínimo.

3. Análisis de riesgos: Realizaría un análisis de riesgos para evaluar las consecuencias potenciales de no probar completamente la Historia de Usuario en cuestión. Si el riesgo es aceptable y el impacto potencial es bajo, podría decidir posponer las pruebas formales hasta el siguiente sprint, priorizando otras tareas críticas que ya se hayan planificado y garantizando la calidad general del producto.

4. Discusión con el Product Owner: Sería importante tener una conversación con el Product Owner para discutir las opciones y encontrar una solución conjunta. Podríamos explorar la posibilidad de incluir la Historia de Usuario en un sprint futuro o buscar alternativas, como realizar pruebas de validación rápida fuera del sprint o aprovechar el feedback de usuarios o pruebas en entornos de preproducción.