

TP Integrador

Pruebas de aceptación

Deberá diseñar las pruebas de aceptación a realizar sobre el trabajo que haya realizado otra persona, el objetivo es encontrar la mayor cantidad de errores posibles pensando antes que es lo que se pretende probar. Cada prueba debe estar documentada, de modo que sea posible reproducir la prueba en cualquier momento.

A continuación se presenta un ejemplo de como podría diseñar las pruebas a realizar:

Estacionamiento - Usuario

#	Nombre (función que se prueba)	Valores ingresados	Resultado esperado	Resultado observado	Resultado de la prueba (ok ó fallo)	Tipo error / Categoría	Prioridad
1	Loguearse con usuario con usuario inexistente/ incorrecto en base de datos	email: example@example.com contraseña:123	Debería mostrar error de usuario inválido y no dejar continuar hacia al programa	Da un mensaje de usuario inválido pero de igual forma accede al programa	Fallo	Integridad de datos	Urgente
2	Validar que el usuario ingrese un correo con formato válido.	email: asdadas	Debería mostrar error de correo inválido y no dejar continuar hacia al programa	No valida el formato del correo.	Fallo	Integridad de datos	Alta
3	Login sin ingresar datos en ningún campo	email: contraseña:	Debería mostrar una advertencia de error y no ingresar al usuario al sistema	Se muestra un mensaje de error, y se permanece en esa vista	Ok	Integridad de datos	
4	Login sin ingresar contraseña	email: example@example.com contraseña:	Debería pedir una contraseña y no avanzar al sistema	El sistema avisa que el usuario no es válido, pero igualmente ingresa al usuario.	Falló	Integridad de datos	Urgente
5	Registro de usuario	email: example@example.com contraseña: 123	Debería registrar el usuario en la base de datos e ingresar al sistema	El botón no reacciona a la interacción del usuario	Fallo	Funcionalidad	Alta
6	Login con usuario registrado en base de datos	email: a@example.com Contraseña: 123456	Debería acceder al programa.	Efectivamente una vez ingresados los datos, da acceso al programa sin problemas	Ok	Funcionalidad	
7	Login con usuario y contraseña	email: a@example.com Contraseña: 123456	Debería ocultar los caracteres de la contraseña mientras se ingresa.	Deja los datos a la vista.	Fallo	Seguridad	Media
8	Login con email registrado y contraseña incorrecta	email: a@example.com Contraseña: password	Debería mostrar mensaje de contraseña incorrecta o dato incorrecto y no dejar entrar al programa.	Deja entrar al programa, dejando realizar todas las actividades del mismo, y muestra un pequeño mensaje de	Fallo	Seguridad	Urgente

				usuario inválido.			
9	Cerrar cuenta con auto estacionado.		Debería mantener el estado de estacionado al/los autos estacionados.	Los mantiene en estado "estacionado".	Ok	Funcionalidad	
10	Eliminar auto estacionado		Debería grabar en la base de datos el estado de salida	No hace nada.	Falla	Funcionalidad	Alta
11	Estacionar un auto sin saldo suficiente (100)	patente: 1232	Debería no dejar estacionar el auto y notifica al usuario	No deja estacionar el auto pero no notifica	Falla	Interfaz de usuario	Baja
12	Estacionar sin ingresar patente		Debería no hacer nada o mostrar un mensaje de que no se ingresó ninguna patente	Estaciona el auto y descuenta el saldo al salir	Falla	Integridad de datos	Alta
13	Estacionar con patente no valida (formato incorrecto)	patente: sdgbjd	Debería mostrar un mensaje de error de patente incorrecta	Estaciona el auto y descuenta el saldo al salir	Falla	Integridad de datos	Medio
14	Dejar estacionado por mucho tiempo	patente: sdgbjd	Debería descontar en base al tiempo que estuvo estacionado	Descuenta al salir pero un monto constante de \$30 sin importar el tiempo que esté.	Falla	Funcionalidad	Alta
15	Ingreso patente con formato correcto	patente: OPU 713	Debería aceptarla.	Estaciona el auto de manera correcta	Ok	Integridad de datos	

Estacionamiento - Inspector

#	Nombre (función que se prueba)	Valores ingresados	Resultado esperado	Resultado observado	Resultado de la prueba (ok ó fallo)	Tipo error / Categoría	Prioridad
1	Aplicar una multa a un auto no estacionado (patente válida)	patente: ab334hd	Que notifique que el auto no esta estacionado , dar la opción de aplicar la multa y aplicar la multa	Notifica que el auto no esta estacionado y permite aplicar la multa	ok	Funcionalidad	
2	Verificación de auto estacionado, patente escrita con o sin mayúsculas	patente guardada en la base: AB334ZM patente ingresada: ab334zm	Si la patente ya está en la base de datos, debería detectar el auto como estacionado, sin importar si esta tiene o no mayúsculas	Notifica que el auto no esta estacionado y permite aplicar la multa	Falla	Funcionalidad	Alta
3	Ingresar un auto no estacionado pero decidir no aplicar la multa	patente: ab334hd	Que notifique que el auto no está estacionado, dar la opción de aplicar la multa o no, y al decidir no aplicar que no la aplique	Notifica que el auto no esta estacionado y al decidir no aplicarla no la aplica	ok	Funcionalidad	

4	Aplicar multas al mismo auto varias veces	patente: ab334hd	Debe notificar que el auto ya fue multado recientemente y no multarlo	Aplica varias multas sin notificar	falla	Funcionalidad	Media
---	---	------------------	---	------------------------------------	-------	---------------	-------

Las últimas columnas de la tabla pueden ser personalizadas en base a las necesidades de lo que se pretenda documentar. Un ejemplo de los posibles valores puede ser:

Tipo de error / categoría:

Interfaz de usuario
Funcionalidad
Persistencia
Integridad de datos
Seguridad

Prioridad

Sin prioridad
Baja
Media
Alta
Urgente

Se pide:

- Diseñar y realizar por lo menos 10 pruebas.
- Opcionalmente, registrar los bug encontrados en la siguiente herramienta: <https://backlog.com>