

23/08

User story vs. U

reg → funcional
↳ no funcional.

U → describe de forma técnica lo q tiene q hacer un software para a pto.
User story → describe la necesidad del cliente.

↳ No enfocamos en una necesidad de negocio.

3 'C' de la User story.

- Conversación → (+) imp.
- Card - Tarjeta
- Confirmación.

parte de
una tarjeta

Identificación frase escrita
→ frase verbal
Descripción
criterios de aceptación
RA - quien
QUE
PARA QUE → valor de negocio
prueba de usuario. del un CP.

- para evitar software q NO se usa.

Diferencia entre el usuario y el cliente

puede o no
ser el usuario,
el cliente.

↳ El q paga

→ El usuario debería tomar las
decisiones

NOTA

User story → valor de negocio x sí misma.

ej: El usuario nabel inicia sesión, nabel selecciona opc. hacer cliente. HA 28/08

nabel ingresa monto a cobrar 5000.000, nabel selecciona opc.

continuar. → He del software

frase verbal → ej: Cobrar al cliente

Descripción → ej: Como [ROL] quiero [QUE] para [valor de negocio]

Como casero quiero cobrar al cliente para finalizar la compra.

• cuando tenemos una user story dividir → o No tiene VALOR el q es parte de otra cosa.

• cada user story tiene 1 valor de negocio → y si queremos o tenemos 2 valores de negocio deben ser 2 user story.

criterios de aceptación → ej: quiero que la pantalla sea azul.

• criterios para que el cliente valide que es una buena user story.

• que el log. la US debe hacer.

L para q le sirva el cliente.

• detalla una parte de la conversación.

pruebas de usuario → ej: probar cobrar al cliente con mas de 500k (Fallar)

" " " " " menu " " (pausa) → software dice si

• después de que asumimos q no hay errores se muestra el software al cliente.

• pruebo que funcione como quiero que funcione.

• el cliente puede aceptar o NO el código que entregamos.

• solo se prueba si el algo q puedo controlar.

• se necesita lo suficiente para q el cliente se sienta cómodo.

• tiene q pensarse en el lugar del usuario y como se me sirve el

software program

→ Todos los req. pueden cambiar.

Criterio de ready

(Cada equipo lo define x sí mismo.

Cuando una us. esta lista para ponerse a trabajar.

Criterio posible:

Independiente. {evitar

Negociable → se pueden negociar No estan cerrados.

Valuable → todas las us. tienen q tener un valor de negocio.

si no es se puede dividir
en varios sprints.

Estimable → poder decir cuanto lleva, granitica...

Small → entra en una iteración.

Testable

(TP) → bastos familiares

User story candidatas:

- Registrar gastos

- Registrar responsable de gasto.

- Visualizar planilla de gastos

- Filtrar gastos

- Iniciar sesión

- Registrar usuario

- Modificar gastos.

- Registrar un tipo de gasto.

- Eliminar gastos.

- Cerrar sesión.

preservar q no

verbal
inintuitivo

van en el lenguaje del cliente.

guardar gasto.

Agregar responsable de gastos.

sesión con mail

Registrar con google

iniciar sesión

No x q 2 US estan en la
misma pantalla van a
ver las mismas.

tema - E-pica

L Aplican enteramente del dominio q usamos

L reg. (+) gastos q k (US)

guardar gasto

como usuario ^{membro familiar y / o de una persona en un momento dado.} ^{gastador} quiero guardar un gasto para llevar control de mis finanzas.

C.A:

puede topanar

debe tipo

- te debe mostrar la fecha actual del gasto.
- se puede modificar la fecha actual del gasto. y deberia
- te debe tener la resion iniciada para guardar un gasto.
- debe puede guardar un gasto con un tipo y responsable de gasto asociado.
- te debe poder registrar un monto y una descripcion.

PU:

Probar guardar un gasto con una fecha diferente a la actual (pasar) ^{validar y pasada}

Probar guardar un monto negativo (falla). → agregar

probar guardar un gasto sin responsable tipo asociado (falla)

probar. guardar una fecha futura (falla).

en ca tipo

tiene setter.

Agregar responsable de gastos.