

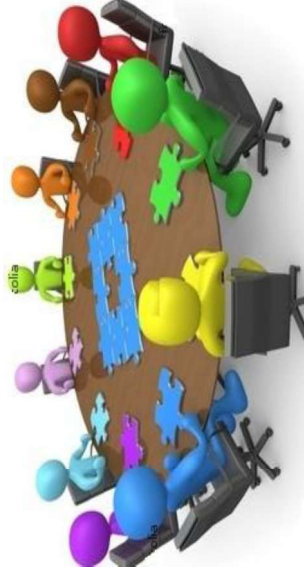
# BDD - GHERKIN

## Prácticas de Automatización de Pruebas

# BDD (Behavioral-Driven Development )

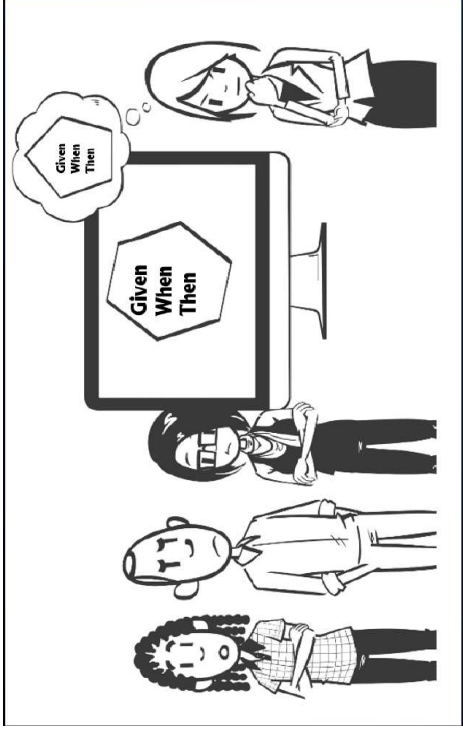
---

“Se le llama BDD a la construcción de software basado en ejemplos.”  
Esos ejemplos son especificaciones del qué debería hacer el sistema, con el propósito de disponer de un **lenguaje común** para unir la parte técnica y la de negocio, siendo este lenguaje un punto de partida para el Testing, la validación, y el desarrollo de software.



# Conexión personas técnicas - negocio

- Para hacer BDD se requiere un lenguaje y para que especifique el comportamiento claro de una aplicación se hace necesario que esa notación sea muy similar al lenguaje natural. Una de las notaciones más populares es **GHERKIN**.
- **Gherkin** establece una cercanía a nivel de lenguaje entre personas técnicas y no técnicas.



# Negocio - Historias de Usuario

Las **historias de usuario**, son pequeñas descripciones de los requerimientos de un cliente y deben incluir los criterios de aceptación. Para escribir una historia de usuario se debe tener en cuenta lo siguiente:

**Definir quién utilizará la funcionalidad a desarrollar:** Es útil imaginarnos qué características tienen las personas que usarán el producto, y detallar su necesidad y problemas actuales para dar lugar al entendimiento sobre sus expectativas reales.

## Historias de Usuario

- **Especificar qué producto quiere el usuario:** Las historias de usuario deben describir qué se espera como salida de la implementación, y cómo se ve beneficiado el usuario final.
- **Para qué utilizará el producto:** establecer el objetivo de construcción, y aporta la posibilidad de explorar otras alternativas para llegar al mismo fin.
- **Los criterios de aceptación:** Está relacionada con las pruebas que se realizarán para verificar el cumplimiento de la expectativa de diseño, usabilidad, rendimiento, y la satisfacción del usuario.

## *Características que deben cumplir las historias de usuario*



**Independientes:** Deben ser atómicas en su definición. Es decir, se debe intentar que no dependa de otras historias para poder completarla.

**Valoradas:** Deben ser valoradas por el cliente. Para poder saber cuanto aporta al Valor de la aplicación y junto con la estimación convertirse en un criterio de prioridad.

**Estimables:** Tener su alcance lo suficientemente definido como para poder suponer una medida de trabajo en la que pueda ser completarla.

Características que  
deben cumplir las  
historias de usuario

- **Pequeñas:** Para poder realizar una estimación con cierta validez y no perder la visión de la Historia de Usuario, se recomienda que sean mayores de dos días y menores de dos semanas.
- **Verificables:** Este es el gran avance de las Historias de Usuario. Que, junto con el cliente, se acuerdan unos Criterios de Aceptación que verifican si se ha cumplido con las funcionalidades descritas y esperadas.

## Historias de Usuario

- Una historia de usuario sigue el siguiente formato:
  - Como <quién> Quiero <qué> Para <objetivo>.
- Ejemplo:
  - Como **Vendedor**, quiero **registrar los productos y cantidades** que me solicita un **cliente** para **crear un pedido de venta**.

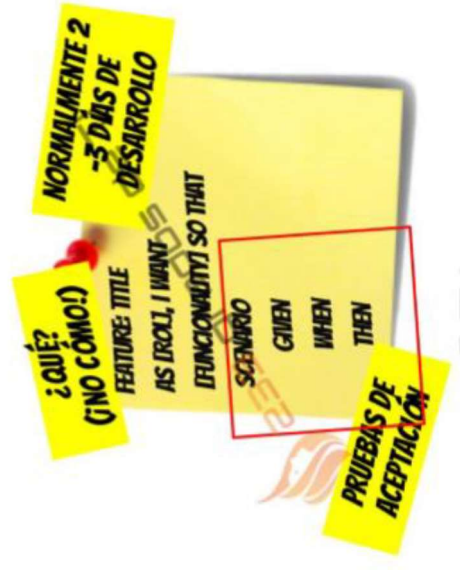
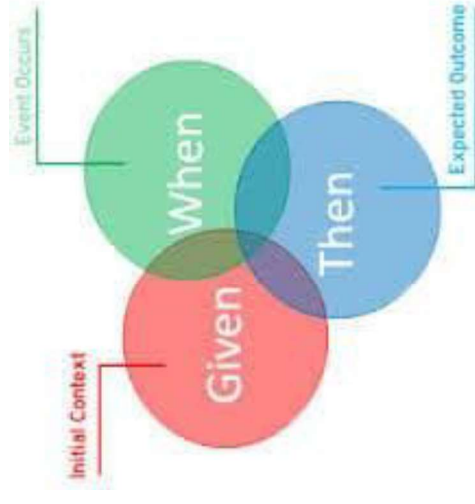


# Taller 1 – Creación Historias de Usuario

- Redactar la historia de usuario para cada uno de los siguientes casos.
  - Compra de productos para el hogar, desde una aplicación móvil. – Rol ama de casa
  - Matriculas online desde una aplicación Web – Rol estudiante
  - Registro de las notas finales de los estudiantes en una aplicación stan alone – Rol profesor
  - Control de pasientes e historial clínico en una aplicación web – Rol Médico

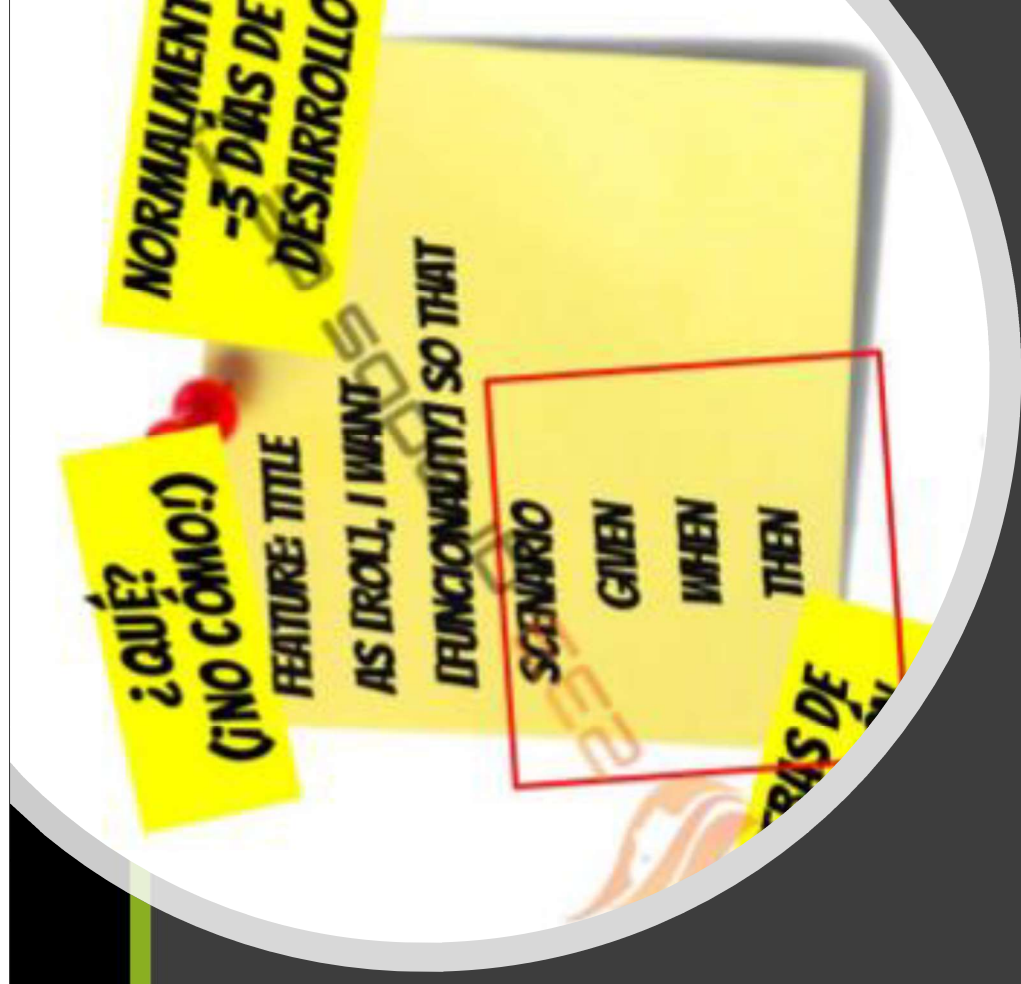
## GHERKIN

- Es un lenguaje que define una estructura y una sintaxis básica, para la descripción de las pruebas que pueden ser entendidas, tanto por los integrantes técnicos del equipo, como también por los Analistas/PO o quien este como representante del cliente.



## COMPONENTES DEL LENGUAJE COMUN

- **Feature (Característica):** Contiene la descripción de la historia de usuario y uno o más escenarios (cada escenario vendrá a ser una prueba).
- Los **Features** deben probar partes o funcionalidades de una App y no la App completa.



## COMPONENTES DEL LENGUAJE COMUN

**Scenario (Escenario):** Describe un ejemplo de comportamiento del sistema, son condiciones de aceptación de una historia de usuario y normalmente las proporcionan los product owner.

El **escenario** contiene una lista de pasos que comienza con algunas de estas palabras claves:

**Given** (| Dado | Dada | Dados | Dadas)

**When** (Cuando)

**Then** (Entonces)

**But** (Pero) o And (Y)

## Ejemplo:

- **Feature:** Buscar GenbetaDev en google
- **As a human**
- **I want** to search GenbetDev en Google
- **So I** can find GenvetaDev website
- **Scenario:** Search for Genbeta Dev
- **Given** I have visited Google
- **When** I search for "GenbetaDev"
- **Then** I see "Genbeta Dev"

# Conceptos - Features

<i>Especifican funcionalmente una necesidad que soluciona la aplicación (historia de usuario)</i>	<i>Feature: [Descripción del feature]</i> <b>Como</b> [rol] <b>Quiero</b> [alguna cosa] <b>Para</b> [que me va aportar]
<i>Puede contener varias historias de usuario</i>	<i>Feature: Envío de entradas por correo electrónico</i> <b>Como</b> socio de la selección colombia <b>Quiero</b> recibir un email con mis entradas <b>Para</b> poder imprimirlas y asistir al partido

# Conceptos - Escenarios

<i>Permite definir una situación concreta. Es un criterio de aceptación redactado como ejemplo.</i>	<i>Feature: [Descripción del escenario]</i> <b>Given</b> [una situación] <b>When</b> [alguna cosa pasa] <b>Then</b> [como responde el sistema]
<i>Contenidos en los features, pueden ser uno o más dentro de un mismo feature.</i>	<i>Feature: El cliente compra un bono familiar</i> <b>Given</b> cliente con bono familiar <b>When</b> llega la fecha del partido <b>Then</b> se le envían todas las entradas de su familia

# Conceptos - Background

<i>Permite añadir un contexto a todos los escenarios de un feature</i>	<p><i>Background:</i> <i>Given el premio de la lotería toca en la oficina</i></p> <p><i>Scenariio:</i> <i>Given un trabajador que compró la décima boleta</i> <i>When se entera de la noticia</i> <i>Then se pone muy contento</i></p> <p><i>Scenariio:</i> <i>Given un trabajador que no compró la décima boleta</i> <i>When se entera de la noticia</i> <i>Then odio a sus compañeros de trabajo</i></p>
<i>Solo incluye un given</i>	



# Conceptos - Given

<i>Es usado para presentarnos los actores (y su entorno) de un escenario</i>	<b>Given</b> un cliente de Bancolombia <b>And</b> tiene su nómina domiciliada  <b>Given</b> usuario de la web <b>And</b> tiene su nómina domiciliada
<i>Siempre va en el comienzo de un escenario y/o background</i>	<b>Given</b> tengo 5 latas de coca cola en la nevera  <b>Given</b> un piloto de fórmula 1 <b>And</b> tiene un coche averiado

# Conceptos - When

<p>Representa la acción que ejecuta el usuario</p> <p><b>When</b> es un utilizado <b>después</b> de presentar los actores (<b>Given</b>)</p>	
<p><i>Aunque se pueden concatenar varios When no es recomendable.</i></p>	<p><b>When</b> solicita un préstamo hipotecario</p> <p><b>When</b> el boleto comprado tiene premio</p> <p><b>When</b> es autenticado correctamente</p> <p><b>When</b> enviar sus datos a ser procesados</p> <p><b>When</b> confirma que los datos son correctos <b>And</b> declara que acepta los términos y condiciones</p>

# Conceptos - Then

<i>Es usado para verificar el resultado esperado después de ejecutarse la acción (when)</i>	<p><b>Then</b> el saldo del cliente es incrementado</p> <p><b>Then</b> quedan 4 latas de cerveza en la nevera</p> <p><b>Then</b> se solicita el código de seguridad</p> <p><b>Then</b> aumenta el saldo de la cuenta 'AAA' And disminuye el saldo de la cuenta 'BBB'</p>
<i>Siempre va en el final de un escenario y para concatenarlos se usa And</i>	

# Conceptos - But


<i>Es usado igual que Then pero su uso es negativo</i>	<p><b>But</b> no se muestra la posición global del cliente</p> <p><b>But</b> el cliente no es autenticado correctamente</p> <p><b>Then</b> aumenta el saldo de la cuenta 'AAA' But no aumenta el saldo de la cuenta 'BBB'</p>
<i>Siempre va en el final de un escenario y no tiene un uso muy extendido</i>	

# Conceptos - Scenario outline, examples

<p>Se trata de un tipo de Scenario concreto donde definimos datos de entrada particulares</p>	<p>Scenario outline: Alquiler de peluches</p> <p>Given Estoy en un centro comercial</p> <p>And mi hijo quiere montar en un peluche con ruedas</p> <p>When alquilo el peluche con ruedas durante &lt;tiempo&gt;</p> <p>Then tengo que pagar &lt;precio&gt;</p> <p>And se me queda cara de &lt;cara_de&gt;</p> <p>Example:</p> <table><tr><td> Tiempo </td><td>Precio</td><td> </td><td>cara de</td><td> </td></tr><tr><td> 15'</td><td> 25€</td><td> </td><td>Me has timado</td><td> </td></tr><tr><td> 30'</td><td> 60€</td><td> </td><td>Me has timado mucho</td><td> </td></tr></table>	Tiempo	Precio		cara de		15'	25€		Me has timado		30'	60€		Me has timado mucho	
Tiempo	Precio		cara de													
15'	25€		Me has timado													
30'	60€		Me has timado mucho													
<p>Permite probar el mismo escenario para distintos datos de entrada. Se recomienda no abusar de los ejemplos</p>																


# Buenas y malas prácticas

“La buena práctica es especificar qué debe hacer el sistema y no cómo debe hacerlo”



Scenario: Add two numbers  
Given I have entered 10 into the calculator  
And I have entered 5 into the calculator  
When I press add  
Then the result should be 15 on the screen

**MALA PRÁCTICA**

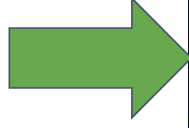


Scenario: sum of two numbers  
When I add 10 and 5  
Then I should see the answer 15

**BUENA PRÁCTICA**

# Ejercicio - Solución

Scenario: Search for the cucumber homepage  
Given I am at <http://google.com>  
When I enter "cucumber" into the search field  
And click "Search"  
Then the top result should say "Cucumber -  
Making BDD fun"



Scenario: Find the Cucumber home page  
When I search for "Cucumber"  
Then the top result should say "Cucumber - Making  
BDD fun"

# Taller 2 – Gherkin

- Redactar la historia de usuario creadas en el taller 1 en lenguaje Gherkin
  - Compra de productos para el hogar, desde una aplicación móvil. – Rol ama de casa
  - Matriculas online desde una aplicación Web – Rol estudiante
  - Registro de las notas finales de los estudiantes en una aplicación web – Rol profesor
  - Control de pacientes e historial clínico en una aplicación web – Rol Médico



# Referencias

---

- [Gherkin... Una guía de supervivencia ágil](#)
- [Tidy App – Gherkin](#)
- <http://www.javiergarzas.com/2014/12/bdd-behavior-driven-development-2.html>

# GLOSARIO

---

---

Feature (Característica).

---

Scenario (Escenario).

---

Step (Paso).

---

Given, When, Then, And y But (Dado, Cuando, Entonces, Y, Pero).

---

Background (Antecedentes).

---

Scenario Outline with Examples (Esquema del escenario con Ejemplos).

---

Step Definition (Definición del Paso o esqueleto del caso de prueba).

---

TDD ó Test Driven Development (Desarrollo guiado por Pruebas)

---

BDD ó Behaviour Driven Development (Desarrollo guiado por Comportamiento).

---