

# Resultados del Ejercicio de la Corbata

- Ingresa a [www.menti.com](https://www.menti.com) con el código: 7954740



# Investigación

Modelo TDD

# Preguntas de Orientación

Es TDD un modelo de desarrollo ágil? Explique su respuesta presentando la descripción general del modelo TDD.

Cuáles son las etapas del modelo TDD? Explique cada etapa y presente un gráfico donde se vea la interacción de estas etapas.

Desde la perspectiva tecnológica qué elementos o herramientas se ven involucradas en un modelo TDD?

Compare los procesos de desarrollo en los que ha participado en sus grupos de estudio y el modelo de procesos TDD. Cuáles son las diferencias o similitudes que encuentra?

Considera que TDD puede aportarle elementos para mejorar sus procesos de desarrollo? Porqué?

# Modelos Ágiles vs. TDD

- Marco ágil de procesos:
  - Es un marco de trabajo para procesos de desarrollo.
  - Surge como una evolución de los modelos tradicionales.
  - Centrado en 4 valores: Individuos e interacciones, Software funcionando, Colaboración con el cliente y Respuesta ante el cambio.
  - De los valores se derivan 12 principios de agilísimo.
  - Incluye iteraciones en el proceso de desarrollo.
- TDD
  - Es una práctica iterativa de diseño de software OO. Presentada como el núcleo de XP.
  - Contiene 3 prácticas: Test-First, Automatización y Refactorizar.
  - Axiomas:
    - “Nunca escribas una nueva funcionalidad sin escribir primero una prueba que falle”.
    - “Si no se puede escribir una prueba para el nuevo código, entonces no se debería estar pensando en incorporar el nuevo código”

# Principios del Agilísimo

La prioridad es **satisfacer al cliente a través de entregas tempranas y continuas de software que aporte valor.**

Aceptación y **bienvenida a los cambios.** Los procesos ágiles aprovechan el cambio para proporcionar una ventaja competitiva al cliente.

**Entregar frecuentemente software que funcione**, desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.

Los **desarrolladores y** las personas de **negocio deben trabajar juntos** diariamente en todo el proyecto.

Los proyectos se desarrollan en torno a **individuos motivados.** Hay que darles el entorno y el apoyo que necesitan, y confiar en ellos para conseguir finalizar el trabajo.

La **conversación cara a cara** es el **método más eficiente y efectivo de comunicar información** al equipo y entre sus miembros.

El **software que funciona** es la **medida fundamental de progreso.**

Los **proceso Ágiles promueven el desarrollo sostenible.** Los promotores, desarrolladores y usuarios deberían ser capaces de mantener un ritmo constante de forma sostenible.

**La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.**

**La simplicidad**, o el arte de maximizar el trabajo no realizado, **es esencial.**

Las mejores arquitecturas, requisitos y diseños emergen de **equipos auto-organizados.**

En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta y perfecciona su comportamiento.

# Preguntas de Orientación

Es TDD un modelo de desarrollo ágil? Explique su respuesta presentando la descripción general del modelo TDD.

Cuáles son las etapas del modelo TDD? Explique cada etapa y presente un gráfico donde se vea la interacción de estas etapas.

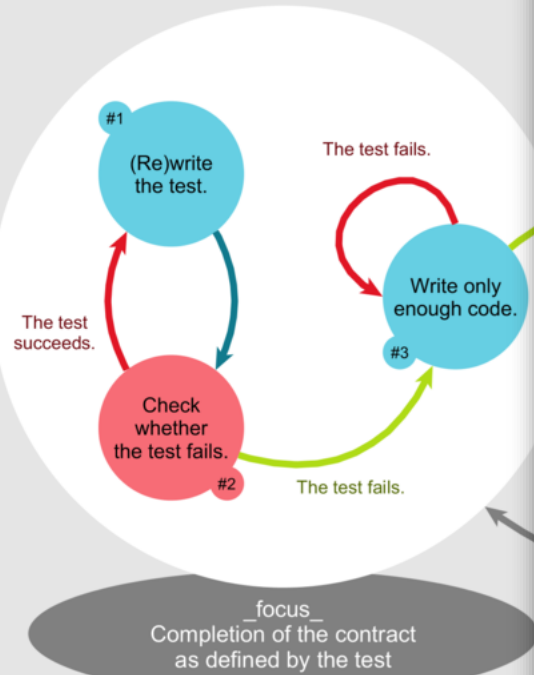
Desde la perspectiva tecnológica qué elementos o herramientas se ven involucradas en un modelo TDD?

Compare los procesos de desarrollo en los que ha participado en sus grupos de estudio y el modelo de procesos TDD. Cuáles son las diferencias o similitudes que encuentra?

Considera que TDD puede aportarle elementos para mejorar sus procesos de desarrollo? Porqué?

# Proceso de TDD y BDD

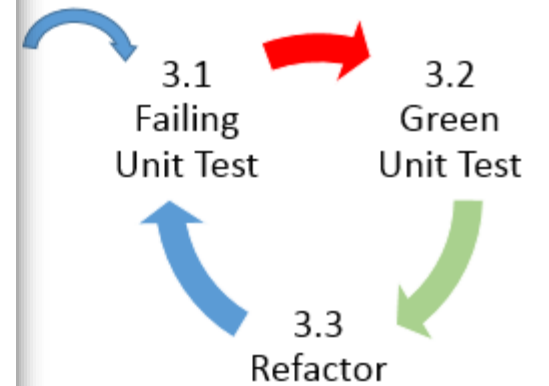
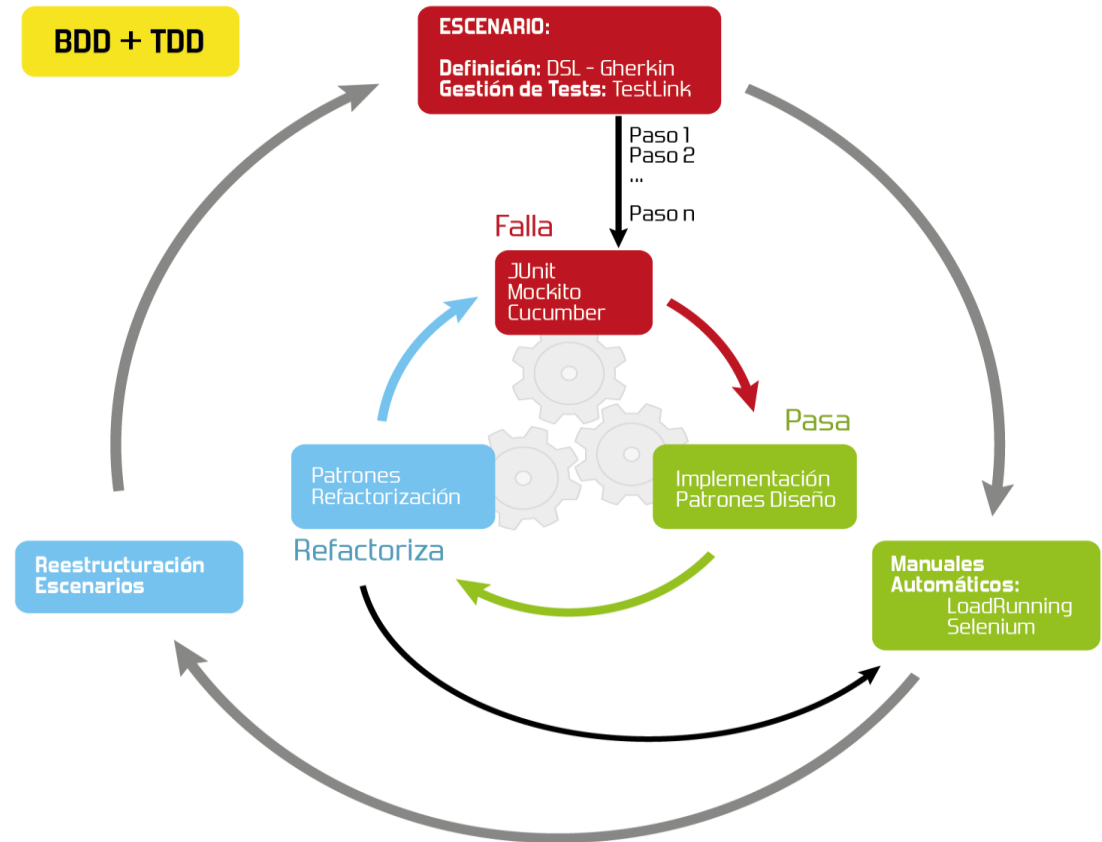
## CODE-DRIVEN TESTING



## COMPORTAMIENTO

Powered by  
autentia

### BDD + TDD





# Preguntas de Orientación

Es TDD un modelo de desarrollo ágil? Explique su respuesta presentando la descripción general del modelo TDD.

Cuáles son las etapas del modelo TDD? Explique cada etapa y presente un gráfico donde se vea la interacción de estas etapas.

Desde la perspectiva tecnológica qué elementos o herramientas se ven involucradas en un modelo TDD?

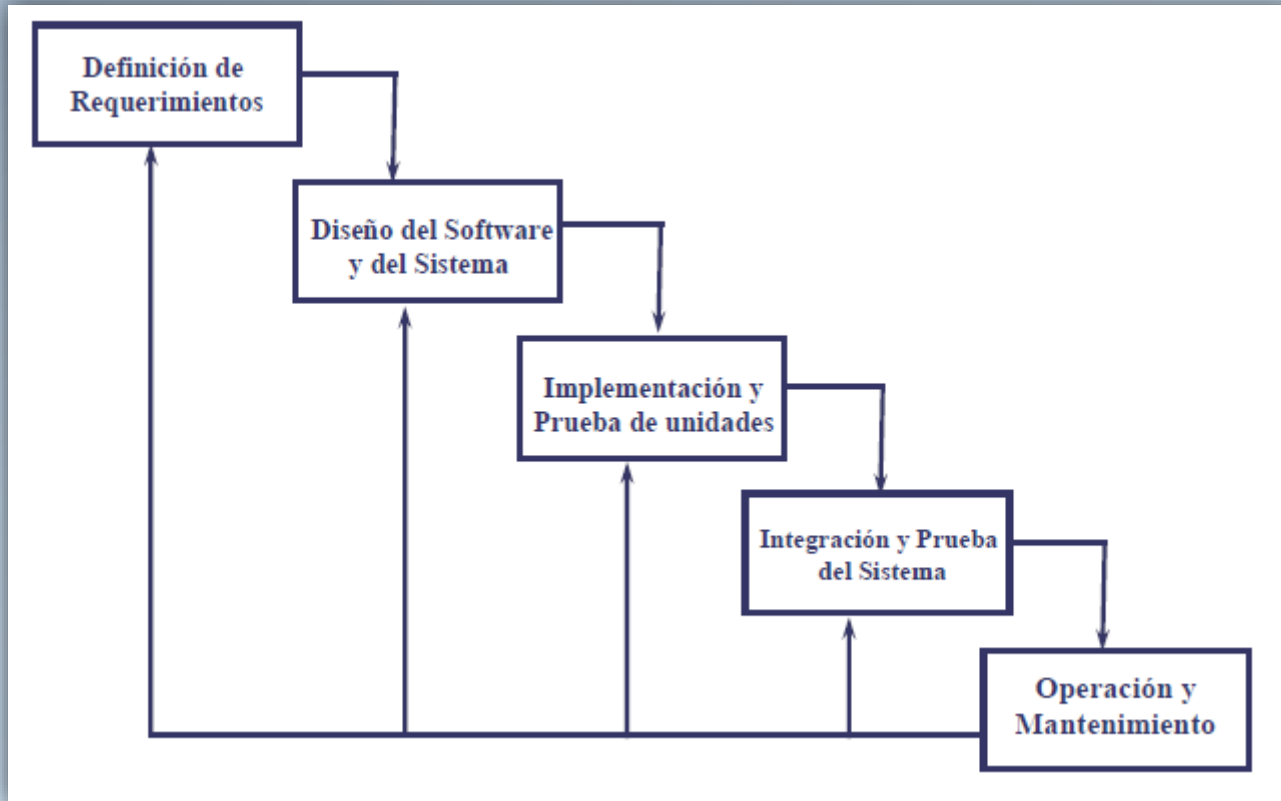
Compare los procesos de desarrollo en los que ha participado en sus grupos de estudio y el modelo de procesos TDD. Cuáles son las diferencias o similitudes que encuentra?

Considera que TDD puede aportarle elementos para mejorar sus procesos de desarrollo? Porqué?

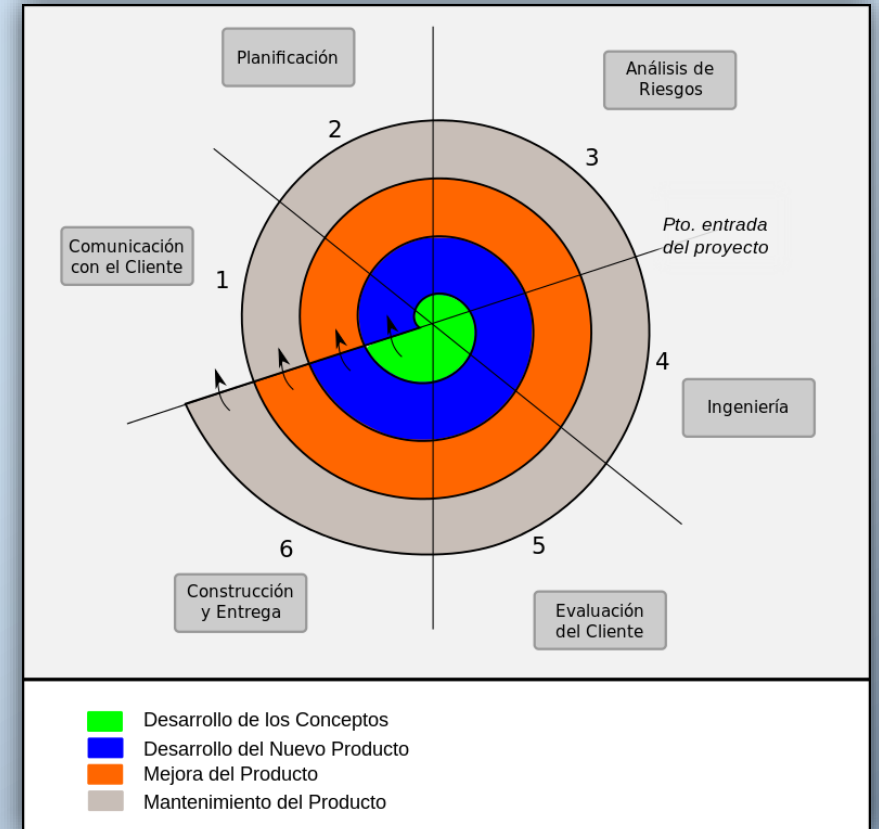


# Procesos Tradicionales

## Modelo en Cascada



## Modelo en Espiral



# Preguntas de Orientación

Es TDD un modelo de desarrollo ágil? Explique su respuesta presentando la descripción general del modelo TDD.

Cuáles son las etapas del modelo TDD? Explique cada etapa y presente un gráfico donde se vea la interacción de estas etapas.

Desde la perspectiva tecnológica qué elementos o herramientas se ven involucradas en un modelo TDD?

Compare los procesos de desarrollo en los que ha participado en sus grupos de estudio y el modelo de procesos TDD. Cuáles son las diferencias o similitudes que encuentra?

Considera que TDD puede aportarle elementos para mejorar sus procesos de desarrollo? Porqué?

# Posibles Mejoras para el Desarrollo

- Ingresa a [www.menti.com](https://www.menti.com) con el código: 7954740

# Definición de Pruebas

- A qué hace referencia en este contexto TDD una prueba?
  - Una prueba para el modelo TDD, puede ser Prueba Unitaria y Prueba de Integración.
  - Prueba Unitaria: Son las pruebas que se hacen a componentes pequeños o clases. Son las pruebas que debe realizar el desarrollador.
  - Pruebas de Integración: Son las pruebas que se realizan para evaluar la comunicación entre dos componentes, módulos o sistemas.
  - Las pruebas son el inicio del proceso de desarrollo.

# Definición de Pruebas

Una prueba contiene los siguientes componentes:

- Entradas a la funcionalidad
- Salidas esperadas de la funcionalidad
- Funcionalidad a probar

Al ejecutar una prueba (manual o automáticamente) se pueden dar los siguientes resultados:

- Éxito: cuando la funcionalidad con las entradas dadas retorna el resultado esperado.
- Fallo: cuando la funcionalidad con las entradas dadas retorna resultados errados o diferentes a los esperados.

Para reportar un fallo se debe incluir las siguientes características:

- Cómo sucedió el fallo?
- Qué sucedió en la prueba?
- Donde se encontró el fallo?
- Cuándo sucede el fallo?

Los desarrolladores deben encontrar el origen del fallo determinado por:

- La pieza de software que está fallando
- Porqué sucede el fallo?
- Qué generó el fallo? Una mala definición del requerimiento, mal diseño o error de programación.

# Ejercicio

## Historia de Usuario: HU3

Yo como estudiante deseo jugar a adivinar los conceptos. Para medir mi conocimiento. Si acierto el nombre del concepto, se debe presentar una nueva definición. Si no acierto pierdo el juego. Deseo también poder saltar una definición que no conozca para acertar las que conozco, pero quiero que al momento de haber visto todas las definiciones, si hay conceptos pendientes de adivinar, se vuelvan a presentar mientras el tiempo no termine.

### Criterios de Aceptación:

- Si el estudiante aprueba las 10 definiciones el programa termina.
- Si el estudiante no ha adivinado todos los conceptos y el tiempo termina, el juego debe terminar.
- Los puntos serán ganados con los conceptos adivinados.

## Historia de Usuario: HU2

Yo como profesor deseo utilizar un juego que contenga conceptos acompañados de definiciones. Para enseñar a mis estudiantes las definiciones de ciertos conceptos de mi materia. El juego debe presentar al estudiante, de manera secuencial, las definiciones y esperar que el estudiante identifique el concepto asociado.

### Criterios de Aceptación:

- Cada partida debe presentar un número de definiciones, especificada por el profesor.
- Cada partida debe tener un tiempo limitado, especificado por el profesor.

## Historia de Usuario: HU1

Yo como profesor deseo que el juego me permita guardar un concepto y su definición para tener una lista de conceptos y definiciones.

### Criterios de Aceptación:

- Los conceptos no se pueden repetir.
- Un concepto se puede eliminar de la lista.

# Cómo definir una prueba?

1. Identifique las funcionalidades de las que hablan las HU.
2. Identifique por cada funcionalidad las entradas para su correcto funcionamiento.
3. Identifique por cada funcionalidad las salidas o resultados del proceso que realiza.
4. Escriba la prueba.



# Primera HU1

- Funcionalidades:
  1. Guardar conceptos y definiciones
  2. Eliminar un concepto
  3. Buscar un concepto – se requiere que el concepto sea único.
- Entradas:
  1. Concepto y Definición
  2. Concepto
  3. Concepto
- Salidas:
  1. Éxito o Mensaje: El concepto ya existe.
  2. Mensaje: El concepto ha sido eliminado o El concepto no Existe
  3. Mensaje: El concepto existe o El concepto no Existe

## Historia de Usuario: HU1

- Yo como profesor deseo que el juego me permita guardar un concepto y su definición para tener una lista de conceptos y definiciones.

## Criterios de Aceptación:

- Los conceptos no se pueden repetir.
- Un concepto se puede eliminar de la lista.

# Escritura de pruebas

Caso No.	Funcionalidad	Prueba	Entrada	Salida
Test1	Guardar	Almacenar un concepto y su descripción.	Concepto: "Prueba" Descripción: "Una prueba es la definición de entradas, salidas y proceso a probar".	Éxito

Escriba el resto de pruebas que considere necesarias dadas las Funcionalidades, entradas y salidas identificados...

# Cómo asegurar que las pruebas son suficientes?

- Aplicación de técnicas de pruebas para tener una cobertura apropiada de la funcionalidad.
- Técnica 1: Particiones de Equivalencia
  - Cuando ha identificado las variables de entrada determine si:
    - La variable recibe un valor específico.
    - La variable puede recibir un valor dentro de un conjunto de valores.
    - La variable puede recibir un valor dentro de un rango de valores.
    - La variable puede recibir un valor de verdad (true o false).
  - Para cada variable se definen los casos de éxito de la funcionalidad o control de error del sistema.

# Analizando las variables del caso de estudio HU1

- Entradas:
  1. Concepto y Definición
  2. Concepto
  3. Concepto
- Clasifique estas variables de entrada en los casos anteriores utilizando la siguiente tabla:

Funcionalidad	Entrada	Condición	Valores Válidos	Valores NO Válidos
Guardar	Concepto	Genera un valor de verdad	No existe el concepto	Existe el concepto
	Descripción	Valor en un rango de valores	Mínimo 10 y máximo 250 caracteres	Menor 10 caracteres y Mayor a 250

# Escritura de pruebas

Caso No.	Funcionalidad	Prueba	Entrada	Salida
Test1	Guardar	Almacenar un concepto y su descripción.	Concepto: "Prueba" Descripción: "Una prueba es la definición de entradas, salidas y proceso a probar".	Éxito

Verifique si se requiere definir otras pruebas dado el análisis anterior.

# Cómo asegurar que las pruebas son suficientes?

- Técnica 2: Análisis de valor límite.
  - Cuando ha identificado las variables de entrada determine si son variables que reciben valores dentro de un rango.
  - Entonces para establecer valores que aseguren el rango debe tener en cuenta como valores válidos:
    - El valor límite inferior
    - El valor límite inferior + 1
    - El valor límite superior – 1
    - El valor límite superior
  - Para valores NO válidos:
    - El valor límite inferior – 1
    - El valor límite superior + 1

# Ejercicio: aplique la definición de pruebas a la HU2 y HU3

## Historia de Usuario: HU3

Yo como estudiante deseo jugar a adivinar los conceptos. Para medir mi conocimiento. Si acierto el nombre del concepto, se debe presentar una nueva definición. Si no acierto pierdo el juego. Deseo también poder saltar una definición que no conozca para acertar las que conozco, pero quiero que al momento de haber visto todas las definiciones, si hay conceptos pendientes de adivinar, se vuelvan a presentar mientras el tiempo no termine.

### Criterios de Aceptación:

- Si el estudiante aprueba las 10 definiciones el programa termina.
- Si el estudiante no ha adivinado todos los conceptos y el tiempo termina, el juego debe terminar.
- Los puntos serán ganados con los conceptos adivinados.

## Historia de Usuario: HU2

- Yo como profesor deseo utilizar un juego que contenga conceptos acompañados de definiciones. Para enseñar a mis estudiantes las definiciones de ciertos conceptos de mi materia. El juego debe presentar al estudiante, de manera secuencial, las definiciones y esperar que el estudiante identifique el concepto asociado.

### Criterios de Aceptación:

- Cada partida debe presentar un número de definiciones, especificada por el profesor.
- Cada partida debe tener un tiempo limitado, especificado por el profesor.