



**GUSTAVO CORONEL**  
DESARROLLA SOFTWARE



# TRABAJANDO CON POSTMAN

## INTRODUCCIÓN A LAS PRUEBAS DE SOFTWARE

**Eric Gustavo Coronel Castillo**

**I N S T R U C T O R**

[youtube.com/DesarrollaSoftware](https://youtube.com/DesarrollaSoftware)

[gcoronelc@gmail.com](mailto:gcoronelc@gmail.com)

# LOGRO ESPERADO

Al finalizar esta lección el participante estará en capacidad de reconocer los diferentes tipos de pruebas de software.



# Definiciones preliminares

- **Defecto o bug**

Es una imperfección en el código que hace que el componente o sistema falle.

- **Error o mistake**

Es una acción que introduce un defecto en el código. Los errores los producen las personas.

- **Falla o failure**

Es una desviación en el comportamiento de un sistema con respecto a lo que se esperaba de él.



# Definiciones preliminares

- **Testing**

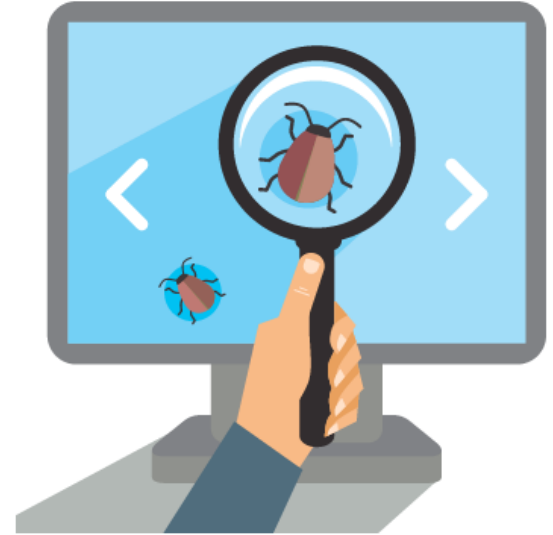
Permite comprobar si un componente se comporta como se esperaba, permite encontrar fallas.

- **Depuración**

Es la investigación de una falla. Permite encontrar el motivo de la falla.

- **Calidad**

El grado en que un sistema o componente cumple con los requerimientos especificados, y las necesidades y/o expectativas del cliente o usuario.



# Objetivos del testing

- Encontrar defectos
- Ganar confianza sobre el nivel de calidad
- Proveer información para la toma de decisiones
- Prevenir defectos

Es mucho mas barato para el desarrollo de un proyecto encontrar un defecto lo antes posible o si es posible evitarlo.



# Niveles de pruebas

- Pruebas de componente (unitarias)
- Pruebas de integración
  - ✓ Integración de componentes
  - ✓ Integración de sistemas
- Pruebas de sistema
- Pruebas de aceptación
  - ✓ Pruebas de aceptación de usuario
  - ✓ Pruebas operacionales
  - ✓ Pruebas de regulación y contrato
  - ✓ Alpha and beta testing



# Tipos de pruebas

- Pruebas estructurales

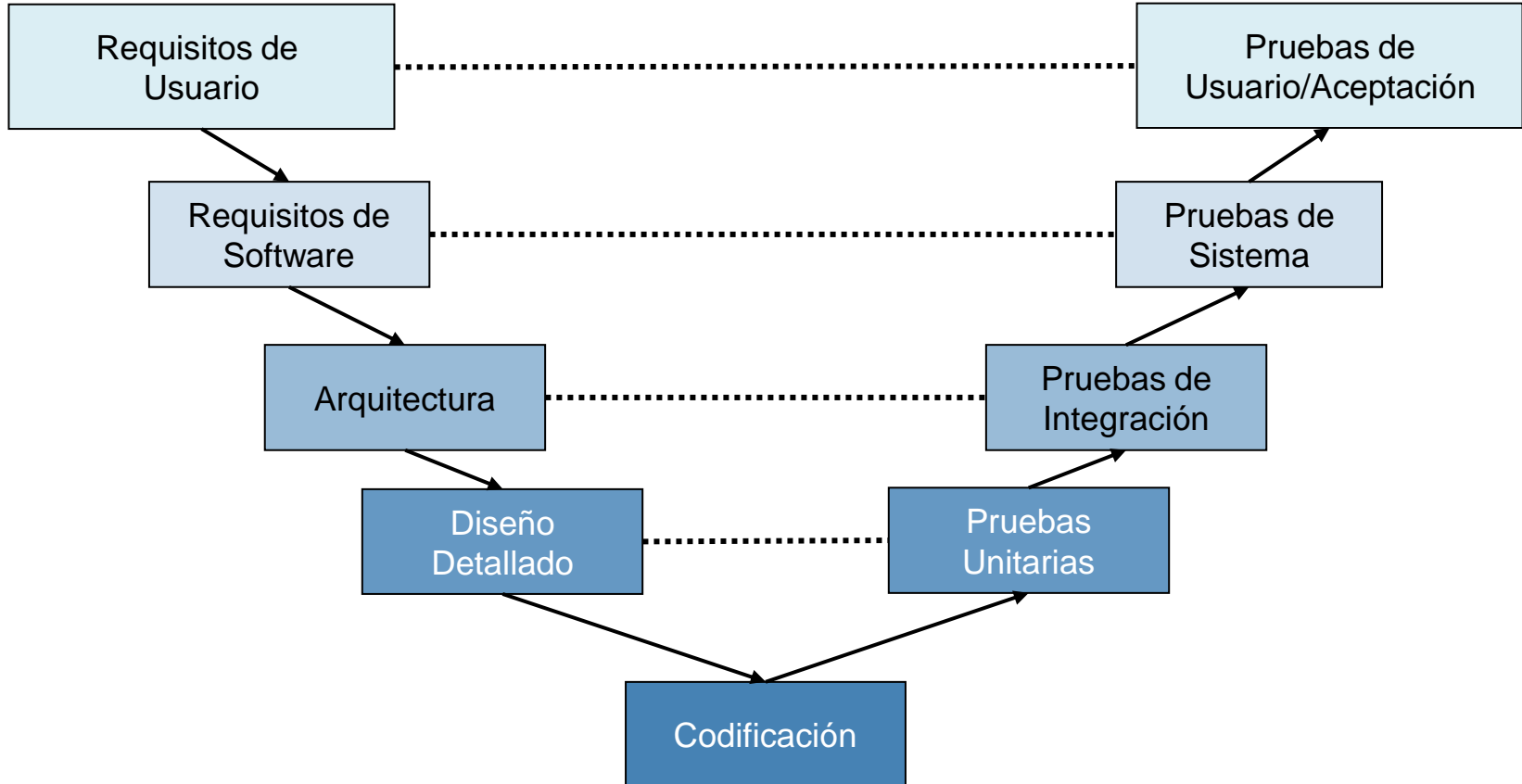
Pruebas de caja blanca, se tiene acceso al código, se comprueba la calidad del código.

- Pruebas relacionadas con cambios

- ✓ Retesting
- ✓ Pruebas de regresión (Automatización)



# Modelo de ciclo de vida en V





# Pruebas unitarias

- Propósito: Encontrar defectos en un módulo o función.
- Normalmente las pruebas unitarias son hechas por el programador del módulo.



# Diseño de pruebas unitarias

- Las pruebas de cada función/método se especifican junto a la especificación de cada función.
- Debemos asegurarnos de incluir pruebas para:
  - Casos habituales y casos extremos de los parámetros.
  - Todas las excepciones.
  - Todos los tipos de efectos laterales.
- Un mismo caso de prueba puede mostrar una excepción y un efecto lateral.



unit testing

# Demostración de pruebas unitarias



unit testing

GRACIAS  
**TOTALES**



**Gustavo Coronel**  
gcoronelc.github.io



[gcoronelc.github.io](https://gcoronelc.github.io)

[youtube.com/DesarrollaSoftware](https://youtube.com/DesarrollaSoftware)

[facebook.com/groups/desarrollasoftware](https://facebook.com/groups/desarrollasoftware)



## FUNDAMENTOS DE PROGRAMACIÓN CON JAVA

Inicia tu aprendizaje, utilizando las mejores prácticas de programación



## CURSO PROFESIONAL DE JAVA ORIENTADA A OBJETOS

Aprende programación en capas, patrones y buenas prácticas



## PROGRAMACIÓN DE BASE DE DATOS ORACLE CON PL/SQL

Aprende a obtener el mejor rendimiento de tú base de datos

JDBC



## PROGRAMACIÓN DE BASE DE DATOS ORACLE CON JDBC

Aprende a programar correctamente con JDBC