

# SC-202 Introducción a la Programación

## Semana 1

## Características de Java

```
for object to m...
mirror_mod.mirror_object
operation == "MIRROR_X":
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False
operation == "MIRROR_Y":
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
operation == "MIRROR_Z":
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True
```

```
selection at the end -add
mirror_ob.select= 1
mirror_ob.select=1
context.scene.objects.active
("Selected" + str(modifier...
mirror_ob.select = 0
= bpy.context.selected_object
data.objects[one.name].select
print("please select exactly
```

--- OPERATOR CLASSES ---

```
types.Operator):
X mirror to the selected
object.mirror_mirror_x"
mirror X"
```

```
context):
context.active_object is not
```

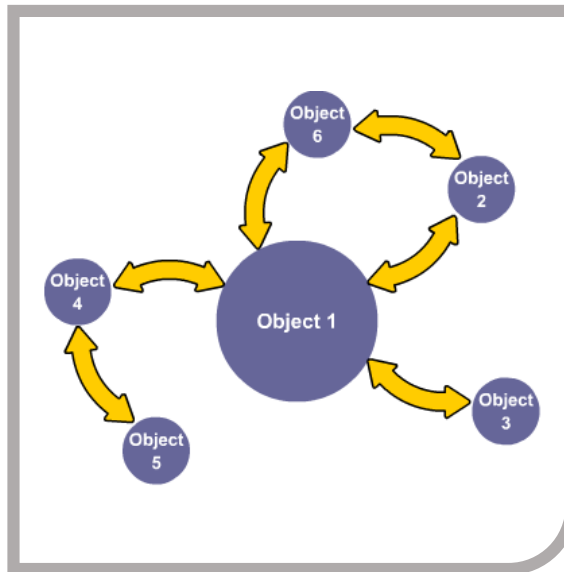
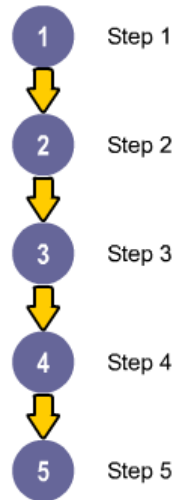
# Principales Características de Java



- El Lenguaje JAVA fue diseñado para ser:
  - 1.- Orientado a Objetos
  - 2.- Distribuido
  - 3.- Simple
  - 4.- Multi-hilo
  - 5.- Seguro
  - 6.- Independiente de plataforma

# 1. Orientado a Objetos

- La programación orientada a objetos nació cerca de 1967 con el lenguaje simula-67.
- Se diferencia de la programación procedimental (estructurada) porque la programación estructurada resuelve problemas en forma secuencial, mientras que en la orientada a objetos se da una interacción entre objetos (procesos).
- Ventajas de la Orientación a Objetos:
- Maximiza la modularidad y encapsulación
  - El sistema se descompone en objetos con unas responsabilidades claramente definidas.
- Extensibilidad
  - Posibilidad de ampliar la funcionalidad de la aplicación de manera sencilla.
- Reusabilidad
  - Posibilidad de reutilizar parte del código para el desarrollo de una aplicación similar.



# Procedimental

# OOP

## 2.- El lenguaje Java es distribuido

- Se dice distribuido porque el lenguaje Java soporta el uso de la tecnología distribuida por la red.
- Además, se pueden cargar piezas de código almacenados en otros lugares de la red.
- Java proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas.

# Ejemplo

1. Se solicita el recurso
2. Se retorna el recurso
3. Se ejecuta el recurso



## 3.- Java es simple

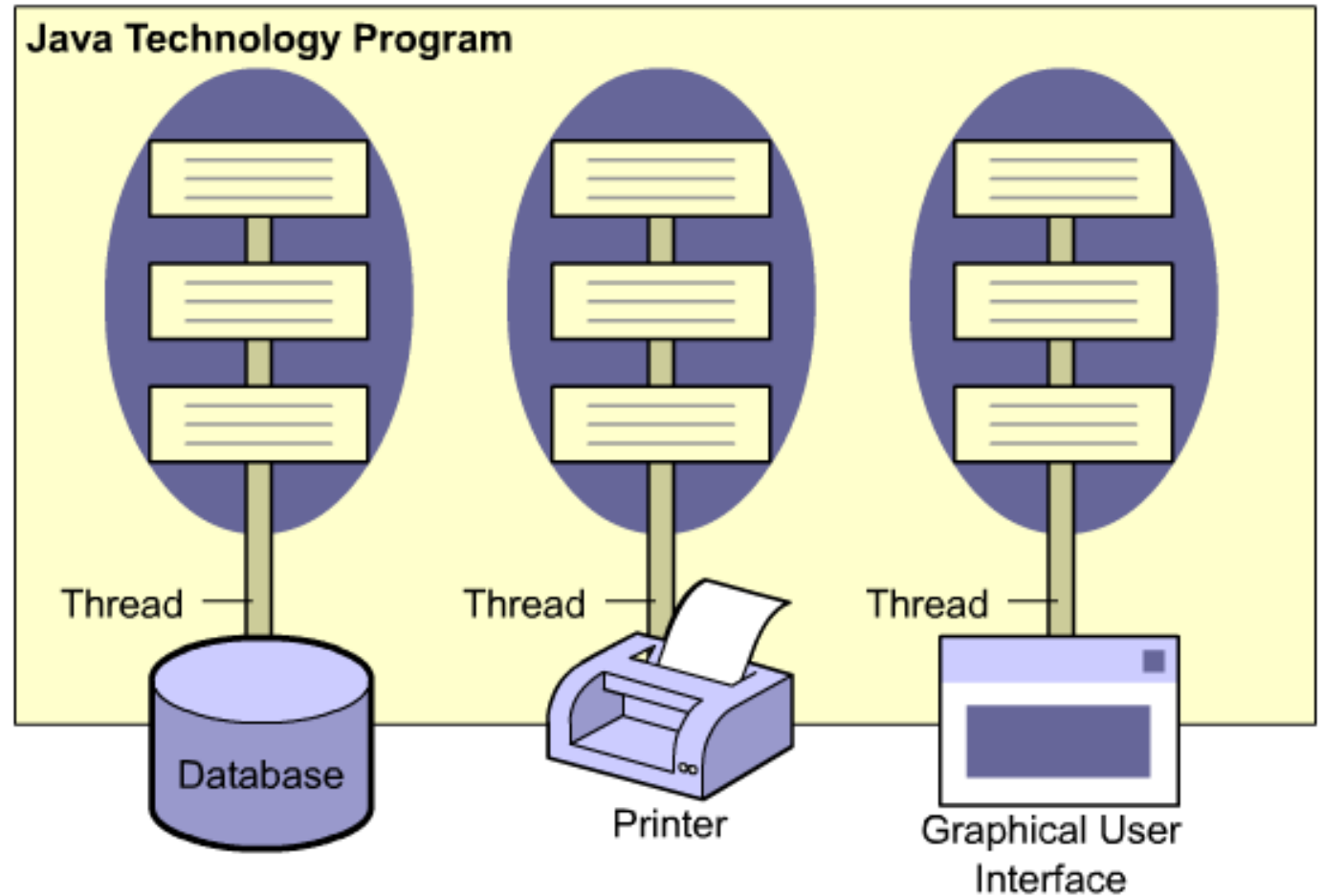
- Se han eliminado algunas posibilidades de programación compleja y oscura encontradas en otros lenguaje populares. Por ejemplo, manipular directamente la memoria o acceder a recursos del sistema.
- En Java se manipulan los objetos mediante referencias, si un objeto no es referenciado el “recolector de basura” lo eliminará.
- Otro punto es que su sintaxis es sencilla, para representar un valor lógico, solo puede tener valores de falso o verdadero, no 0 ó 1 como en otros lenguajes.

## 4.- Java es multi-hilo

- Un programa en java puede ejecutar diferentes tareas en un mismo periodo de tiempo. Acceder a una base de datos mientras se ejecuta un cálculo y se despliega información en la pantalla.
- Esta tecnología permite a Java explotar al máximo los recursos de un sistema.



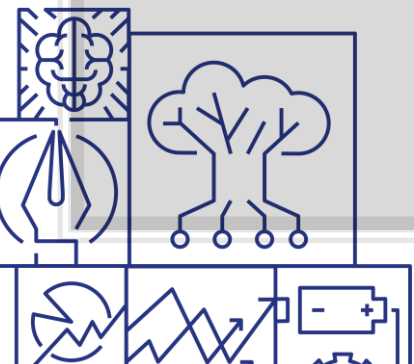
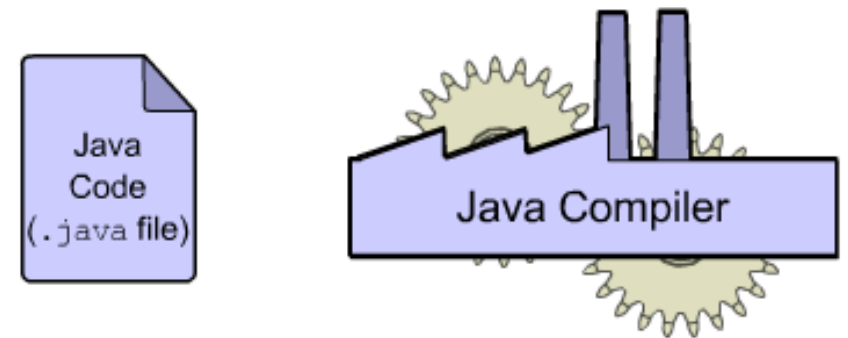
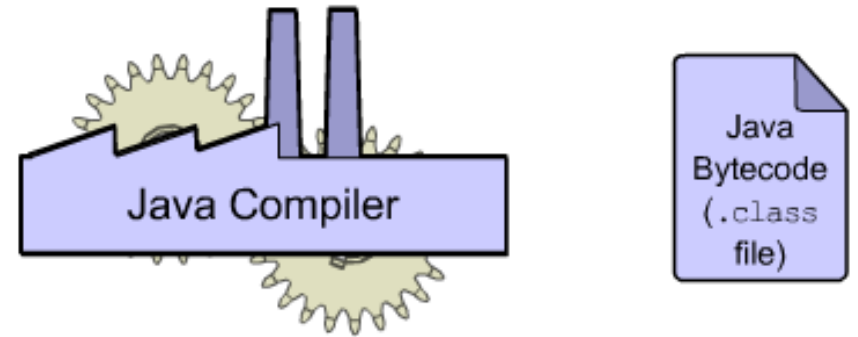
# Ejemplo multi-hilo



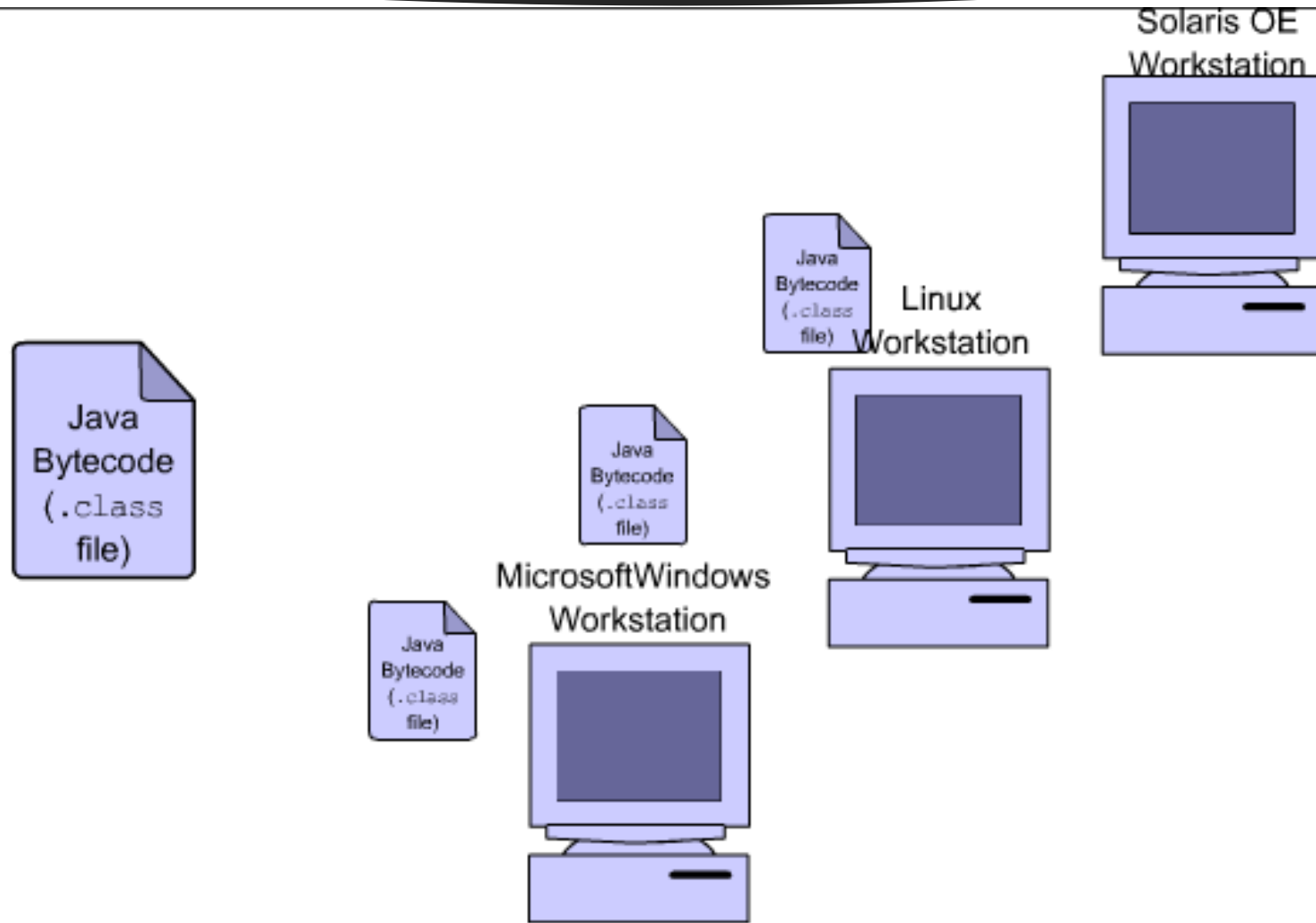
## 5.- Java independiente de plataforma

- Los programas en Java se escriben una vez y se ejecutan en cualquier lugar.
- Gracias a la JVM (Java Virtual Machine) se ejecuta en diferentes sistemas (Windows, Linux, Mac) que toma el código de tecnología java y lo ejecuta.

# Java independiente de plataforma

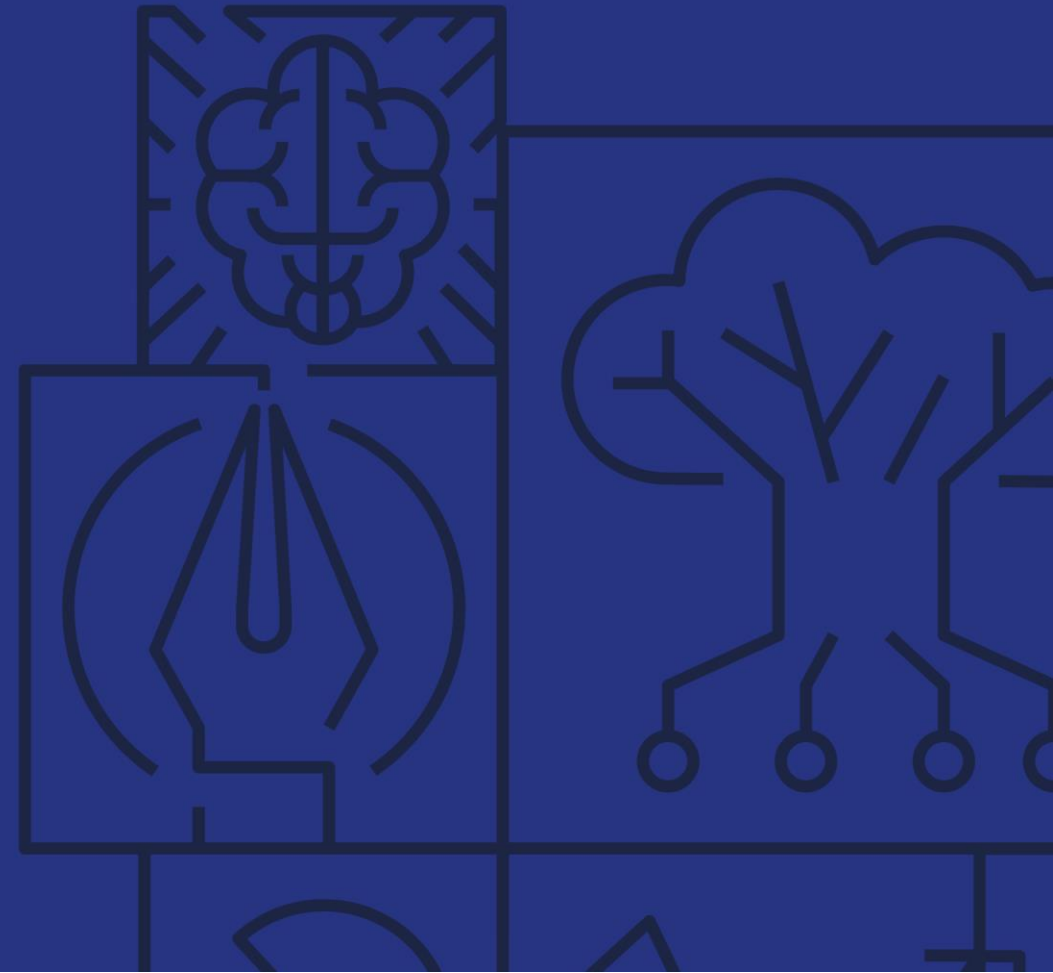


# Java independiente de plataforma





# Grupo de productos de Tecnología Java



# Tecnologías Java

- Las tecnologías Java, así como las JVM han sido incluidas y distribuidas en diferentes formas para cubrir necesidades particulares.

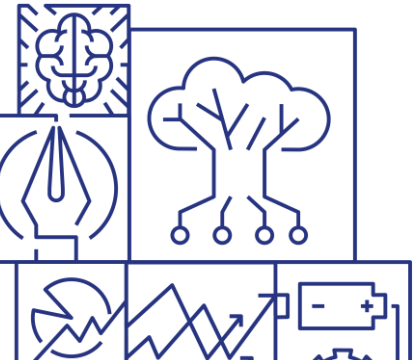
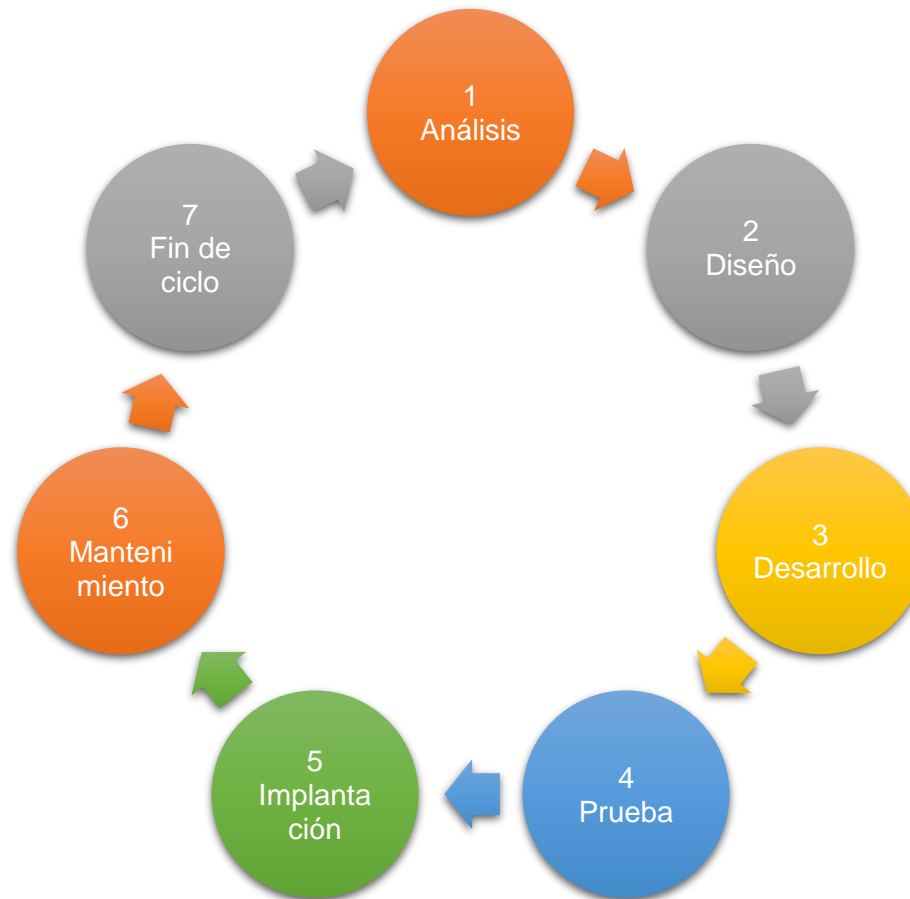


## **Ciclo de vida de productos**



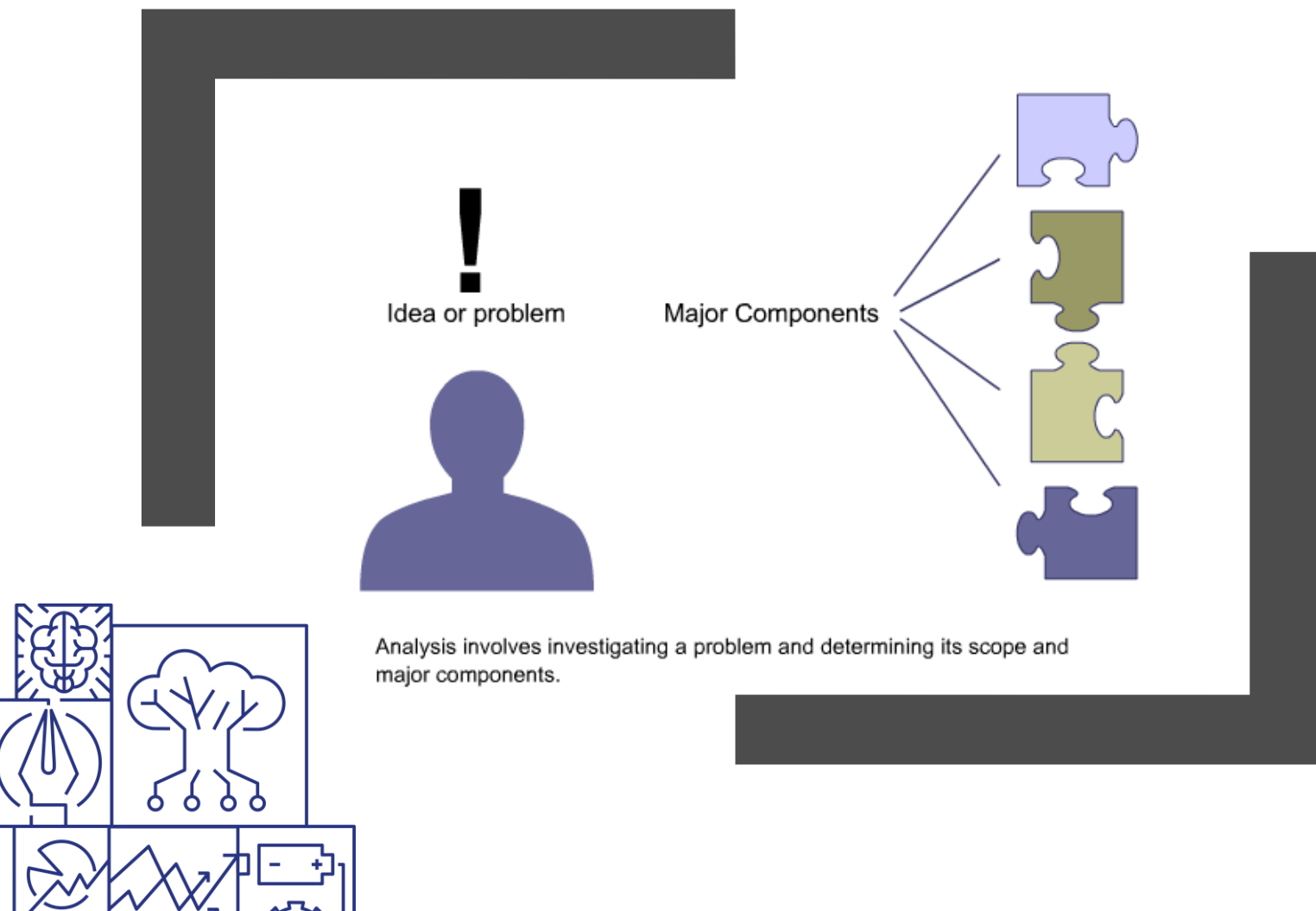
# Ciclo de vida de un producto

- Hay 7 estados consensuados por la industria de desarrollo sobre los diferentes escenarios que se deben superar para desarrollar un nuevo producto.





# Etapa de Análisis

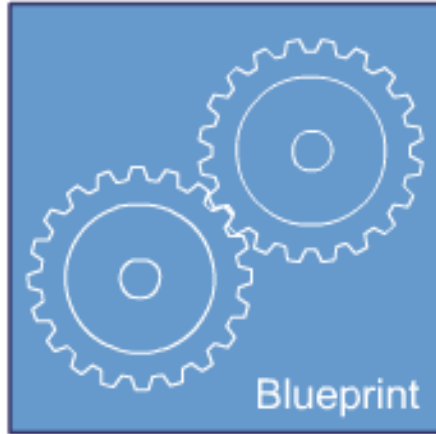


Análisis es el proceso de investigar el problema que se desea resolver, dentro de las tareas se pueden enumerar.

- Definir claramente el problema que se desea resolver.
- Determinar el nicho de mercado al que se desea llegar.
- Establecer el sistema que se desea crear.
- Delimitar el problema y consecuentemente el alcance.
- Identificar los componentes principales para desarrollar el producto.



Major Components



Master Blueprint



Component Blueprints

Design involves creating blueprints for components that you will be developing.

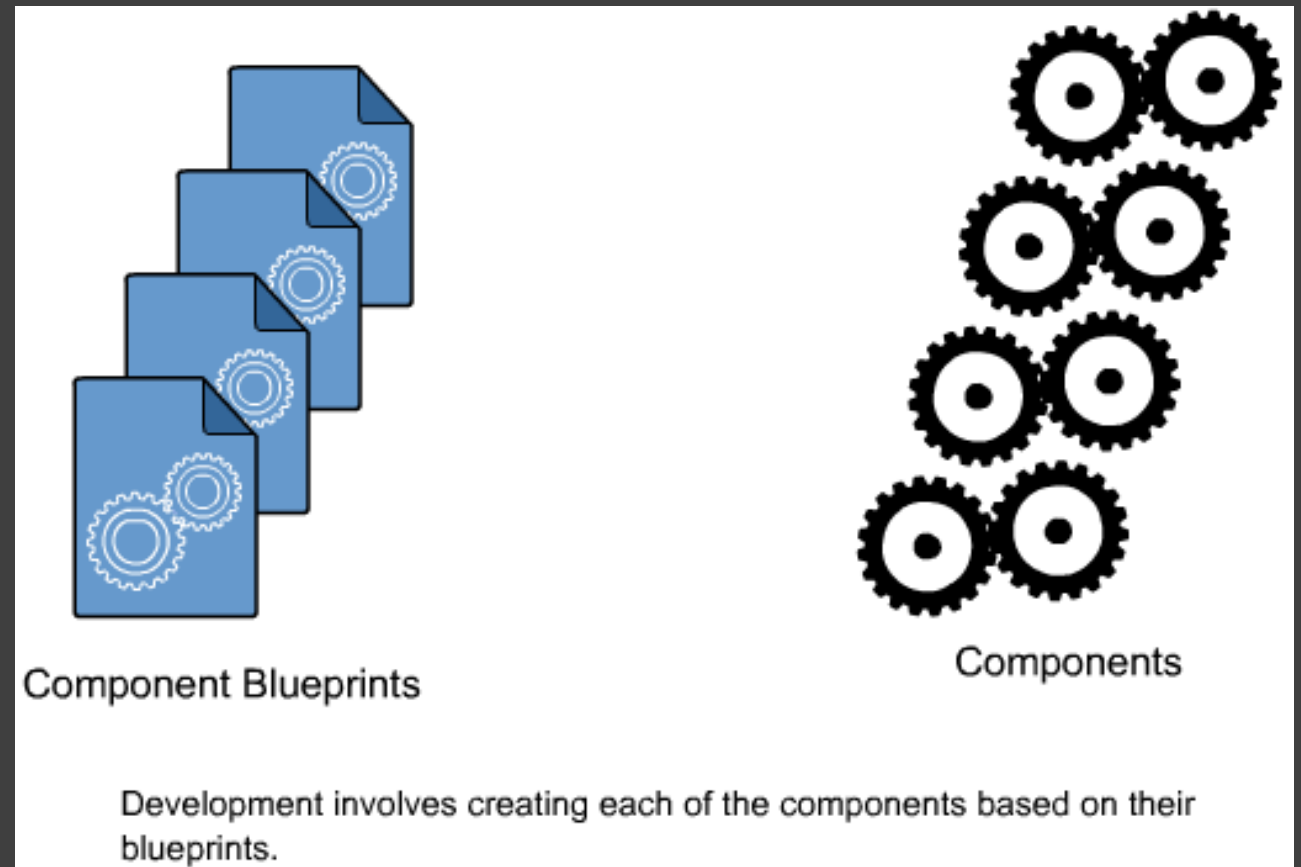
## Etapa de Diseño

El diseño es el proceso de aplicar lo encontrado durante la etapa de análisis.

- La primer tarea es realizar una diagramación o especificación de los productos o componentes del sistema.

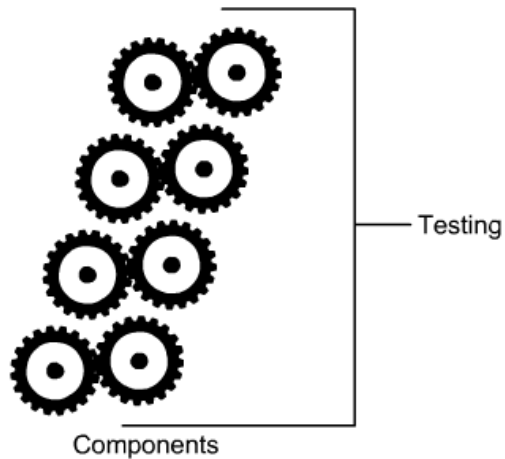
# Etapa de Desarrollo

El desarrollo consiste en utilizar lo diagramado en la etapa de diseño y crear los componentes.

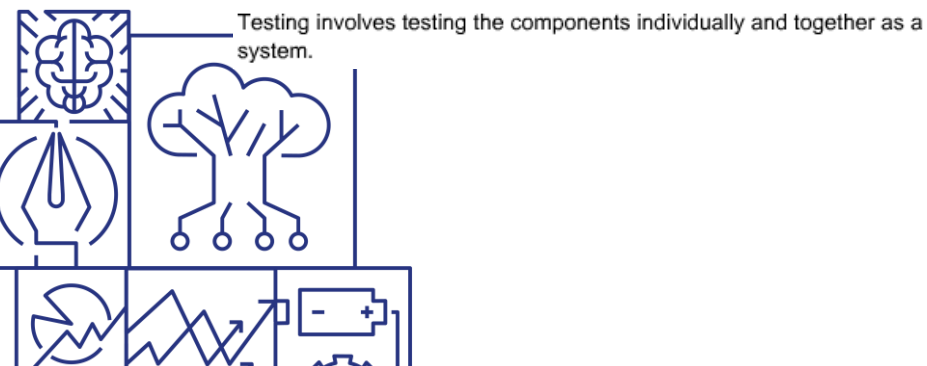


# Etapa de Prueba

---

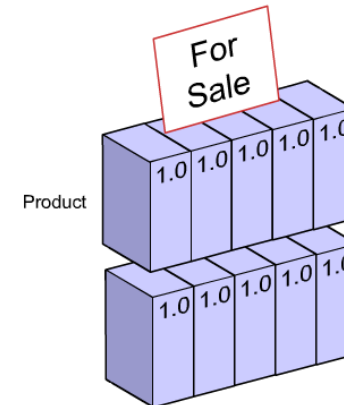


- Prueba consiste en asegurar que los componentes individuales se comportan como un todo y cumplen con las especificaciones de los requerimientos creados en la etapa de diseño
- El buen análisis de un problema facilita un buen diseño de la solución disminuyendo el tiempo en las etapas de desarrollo y prueba.
- Las pruebas son usualmente desarrolladas por un equipo que normalmente no es el mismo que participó en la etapa de desarrollo. Es así para evitar sesgos por parte de los desarrolladores.

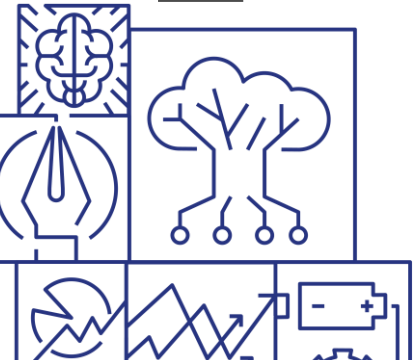


# Etapa de Implementación

- Consiste en hacer disponible el producto para los clientes.
- La etapa de implementación es conocida como la primer entrega al cliente o FCS (First Client Shipment) en la industria de computación.



Implementation refers to shipping a product, making the product available to the customers.



## Etapa de Mantenimiento



- La etapa de mantenimiento se da para corregir algún tipo de problema detectado por los clientes o bien la liberación de mejoras.

## Fin de ciclo

- El PLC (Producto Life Cycle) no tiene una etapa para iniciar un nuevo concepto o producto. Por ello se tiene el EOL (End of Life).
- Esta etapa consiste en finalizar todas las tareas necesarias para asegurar que los clientes y personal conozcan que cierto producto no estará disponible para venta y que se creará un nuevo producto.



End-of-life involves replacing the product with another product that has been developed.