

Arquitecturas de Software para Aplicaciones Empresariales

Introducción a la arquitectura de software



PROGRAMA DE INGENIERIA DE SISTEMAS

Ing. Daniel Eduardo Paz Perafán (danielp@Unicauca.edu.co)

Ing. Pablo A. Magé (pmage@Unicauca.edu.co)

Agenda

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura



La Arquitectura es un nivel de diseño que hace foco en aspectos "más allá de los algoritmos y estructuras de datos de la computación".

La Arquitectura de Software se refiere a una descripción de cómo está organizado un sistema en subsistemas, módulos, componentes y cómo interactúan entre ellos.

Los elementos pueden ser:

- **Entidades lógicas que existen en tiempo de desarrollo** (clases, paquetes, módulos)
- **Entidades físicas** (nodos donde se van a desplegar los sistemas software).
- **Entidades presentes en tiempo de ejecución** (objetos, hilos)



Definición de los sistemas en los cuales se organiza una arquitectura

Un subsistema es esencialmente un límite dentro del cual se ejecuta algún código o se almacenan algunos datos. Un subsistema normalmente corresponderá a un proceso en ejecución. **Un subsistema encapsula varios módulos.**

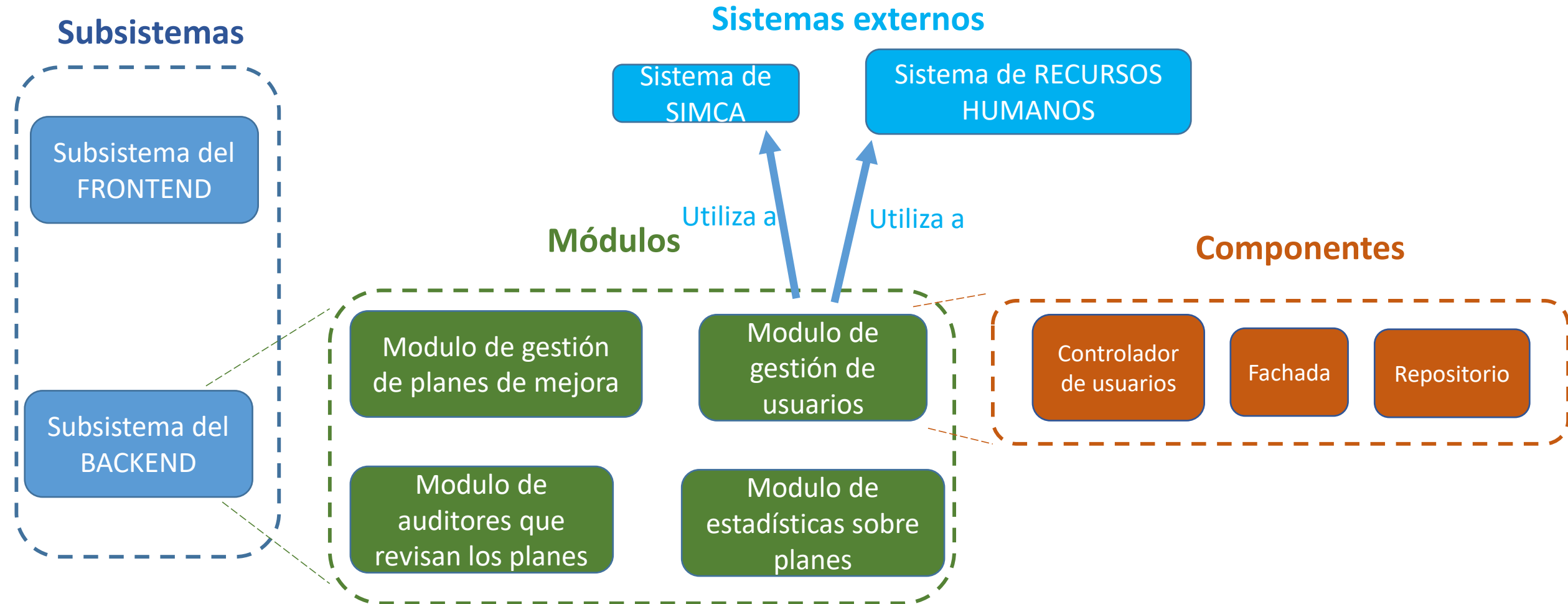
Un módulo es una pieza de software que encapsula un conjunto de responsabilidades correlacionadas y cohesivas. Un modulo implementa varias funcionalidades. **Un modulo encapsula varios componentes.**

Un componente es una agrupación de funciones relacionadas encapsuladas detrás de una interfaz bien definida. **Un componente puede estar constituido por varias clases.**



Definición de arquitectura

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS



Propósito de utilizar una arquitectura representada en un modelo

1. **Como una forma de facilitar la discusión sobre el diseño del sistema.** Una vista arquitectónica de alto nivel de un sistema es útil para la comunicación con las partes interesadas con el fin de que comprendan una visión abstracta del sistema.
2. **Como una forma de documentar una arquitectura que ha sido diseñada.** Una descripción arquitectónica tan detallada facilita la comprensión y la evolución del sistema.



Definición de arquitectura

La arquitectura debe representarse mediante varios diagramas

Diagrama de clases

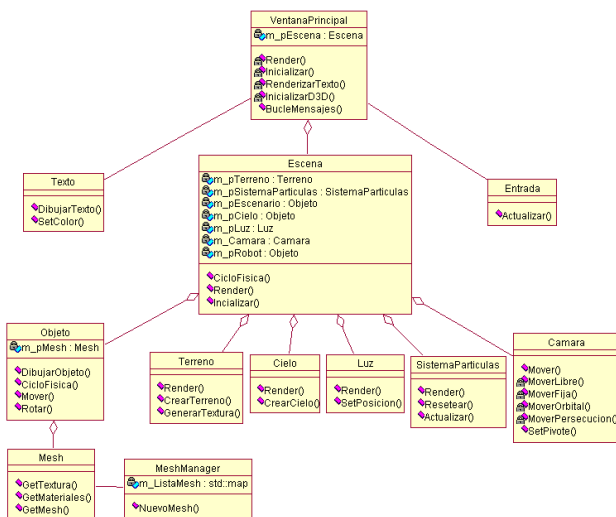


Diagrama de paquetes

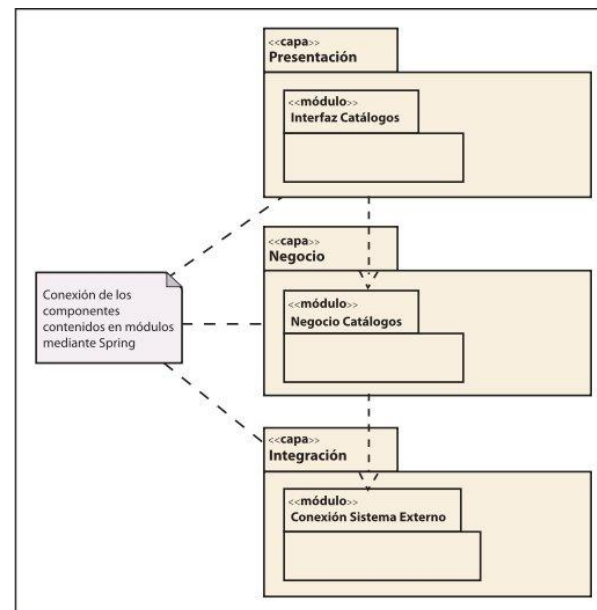
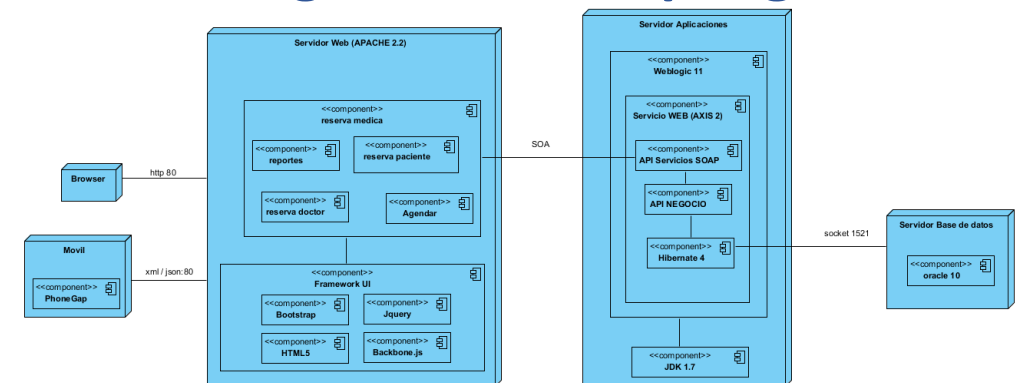


Diagrama de despliegue



¿Que no es la arquitectura de software?

El sistema informático que estamos desarrollando es una aplicación web que permite que un cliente pueda obtener información de todos sus créditos en entidades financieras legalmente matriculadas con el estado.

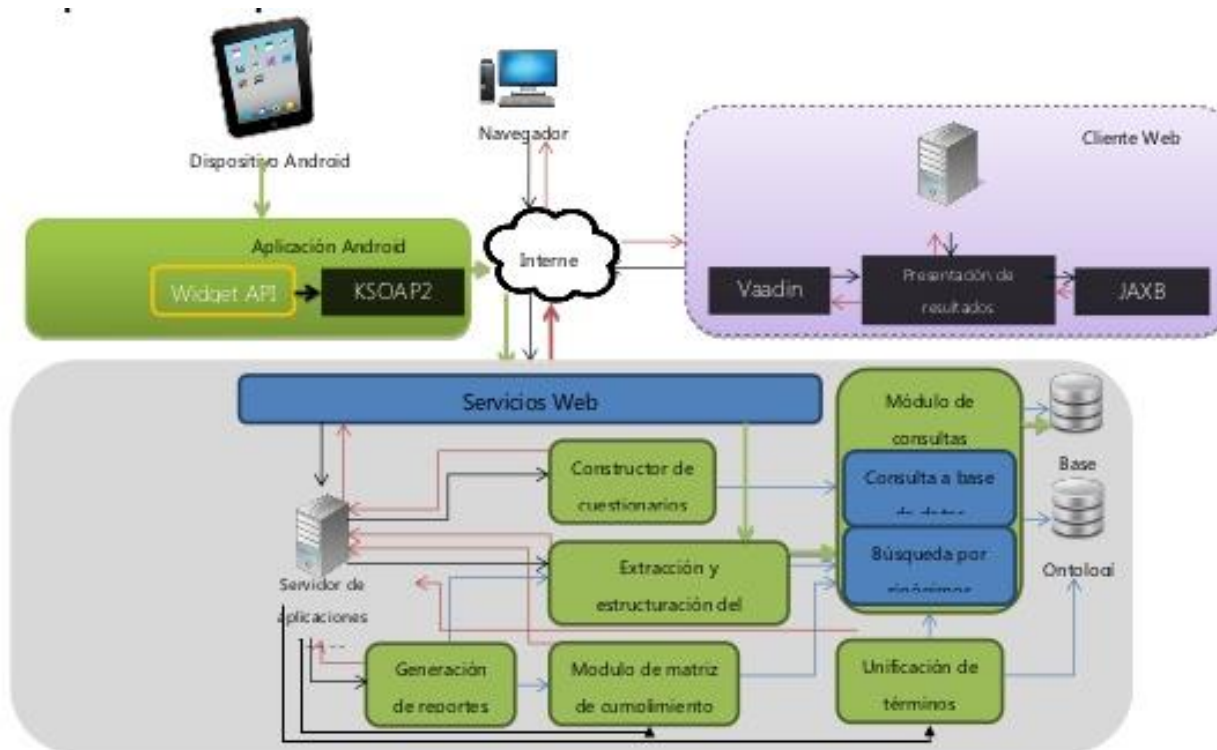
- Esta implementado en java
- Utiliza un framework css bootstrap
- La base de datos es en Oracle
- Utiliza un servidor web apache



Listado de tecnologías y requisitos de la aplicación.

¿Que no es la arquitectura de software?

No es un conjunto de imágenes de una única vista del sistema.



Agenda

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura



Idea inicial y estudio
de factibilidad

Todo software se crea a partir de una
idea inicial y se puede realizar un
estudio de factibilidad

Definición
de requisitos

Diseño del
sistema

Desarrollo de los
subsistemas y
pruebas unitarias

Integración de
los subsistemas

Pruebas del
sistema

Despliegue del
sistema

Mantenimiento
Del sistema

Desmantelamiento y
retirada del sistema



Idea inicial y estudio
de factibilidad

Definición
de requisitos

Se capturan y especificaban los requisitos
funcionales (Servicios) y no funcionales
(Restricciones o condiciones de operación)

Diseño del
sistema

Desarrollo de los
subsistemas y
pruebas unitarias

Integración de
los subsistemas

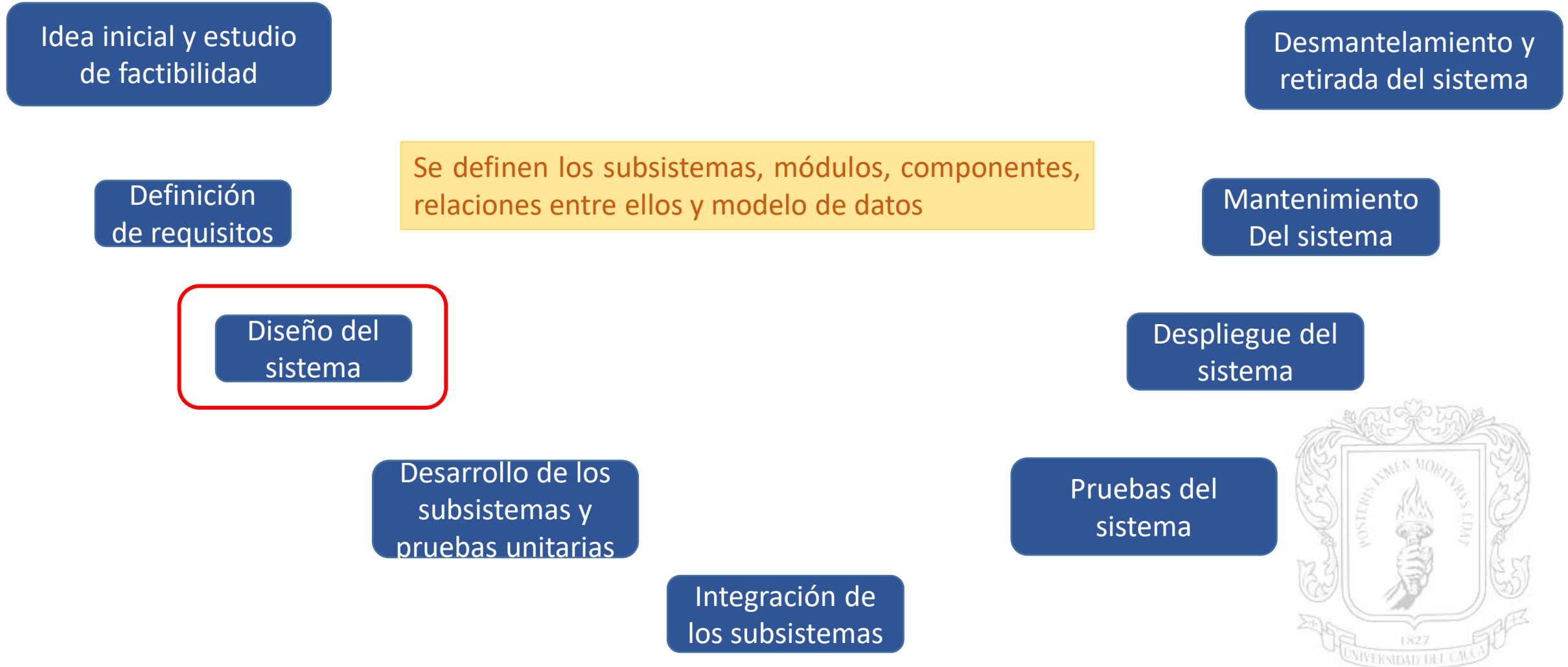
Pruebas del
sistema

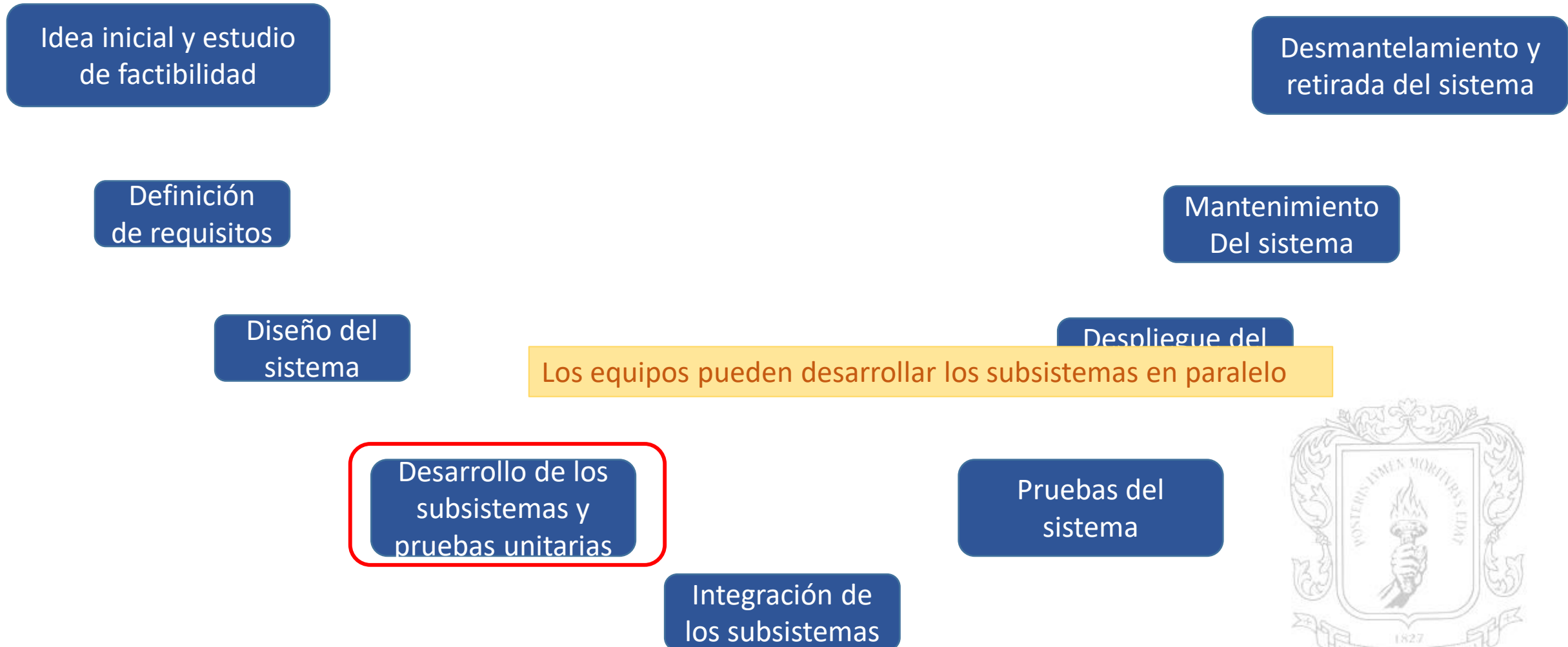
Despliegue del
sistema

Mantenimiento
Del sistema

Desmantelamiento y
retirada del sistema







Idea inicial y estudio
de factibilidad

Desmantelamiento y
retirada del sistema

Definición
de requisitos

Mantenimiento
Del sistema

Diseño del
sistema

Integración big bang», que consiste en integrar todos los componentes al mismo tiempo.

Integración creciente , que consiste en integrar los componentes uno a uno.

Desarrollo de los
subsistemas y
pruebas unitarias

Pruebas del
sistema

Integración de
los subsistemas



Idea inicial y estudio
de factibilidad

Desmantelamiento y
retirada del sistema

Definición
de requisitos

Mantenimiento
Del sistema

Diseño del
sistema

Despliegue del
sistema

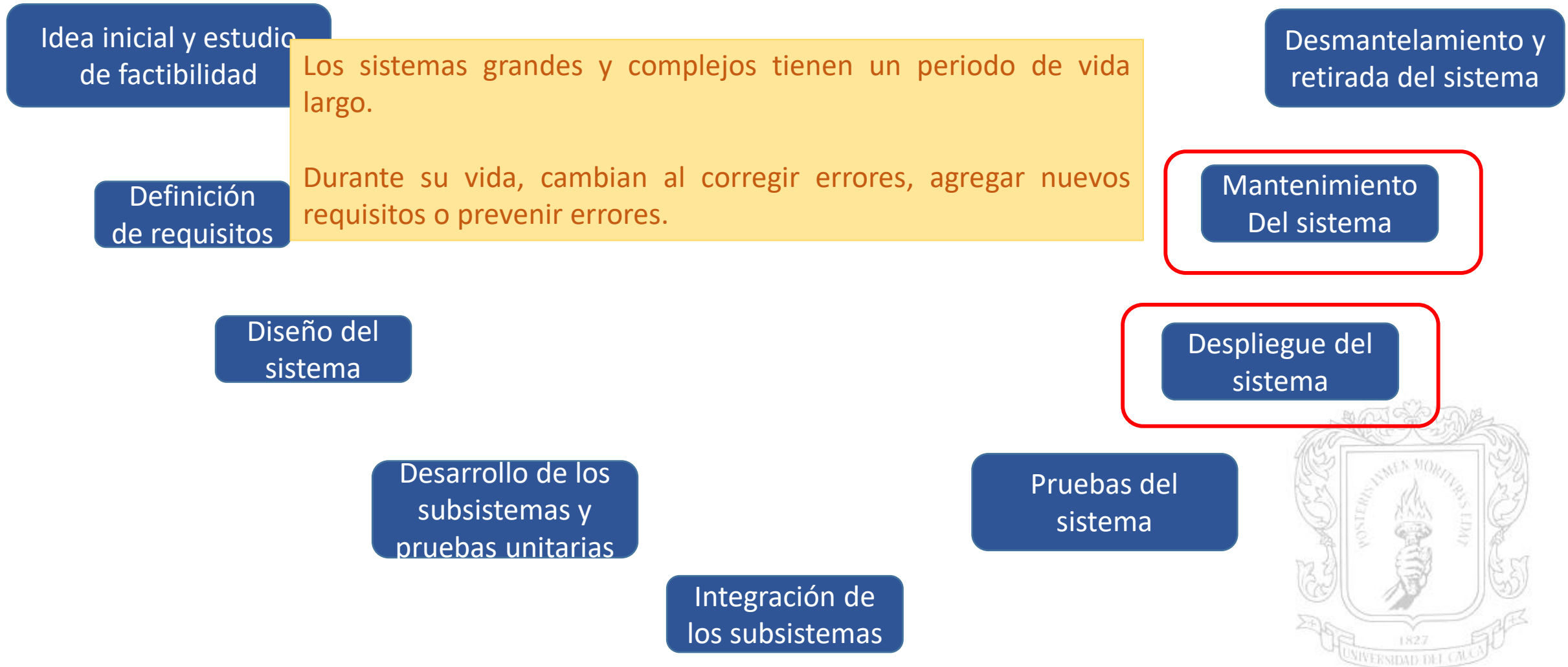
Estas pruebas pretenden probar las interfaces entre los componentes y el comportamiento del sistema en su totalidad.

pruebas unitarias

Pruebas del
sistema

Integración de
los subsistemas





Idea inicial y estudio
de factibilidad

El desmantelamiento del sistema significa poner fuera de servicio a dicho sistema después de que termina su periodo de utilidad operativa.

Desmantelamiento
y retirada del
sistema

Definición
de requisitos

Mantenimiento
Del sistema

Diseño del
sistema

Despliegue del
sistema

Desarrollo de los
subsistemas y
pruebas unitarias

Pruebas del
sistema

Integración de
los subsistemas

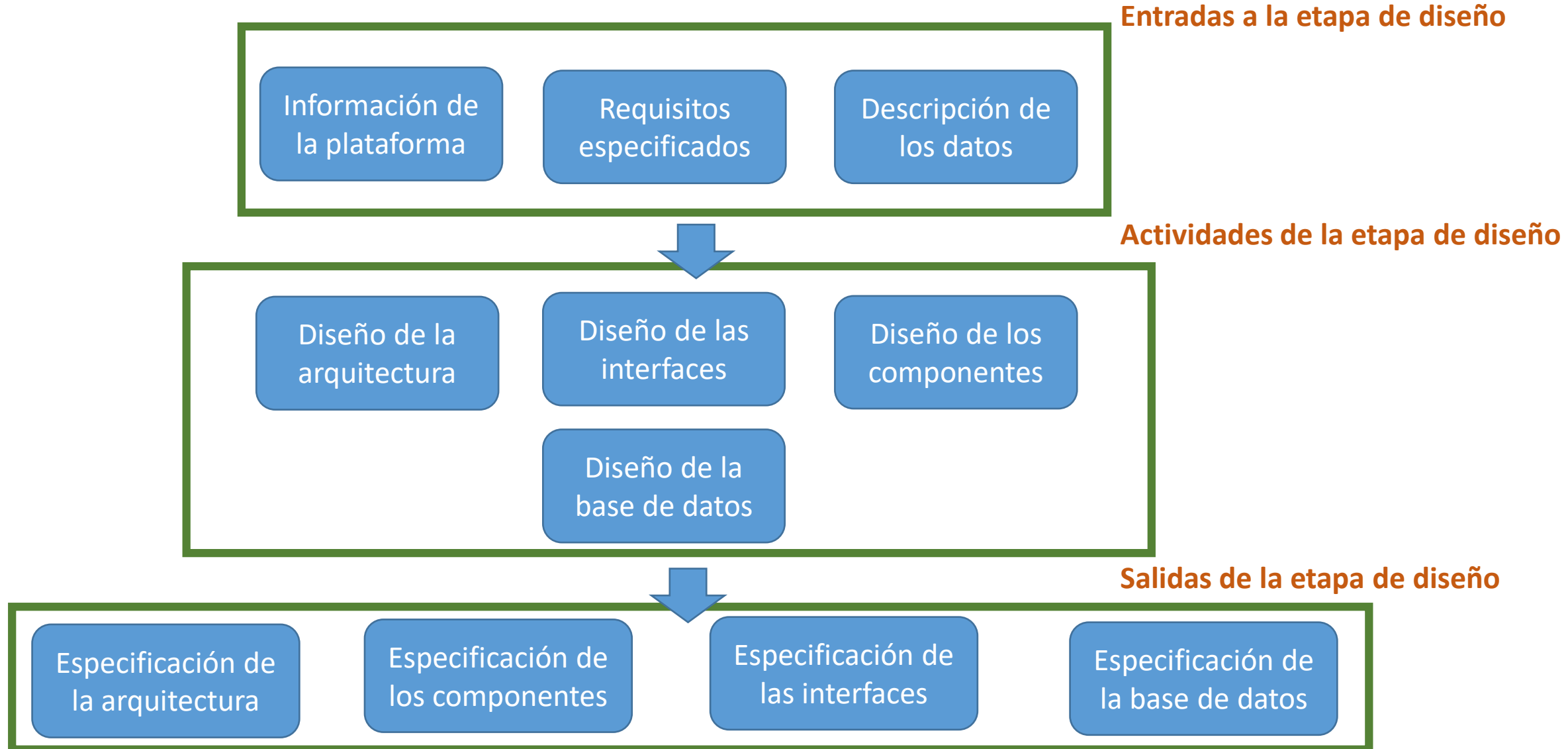


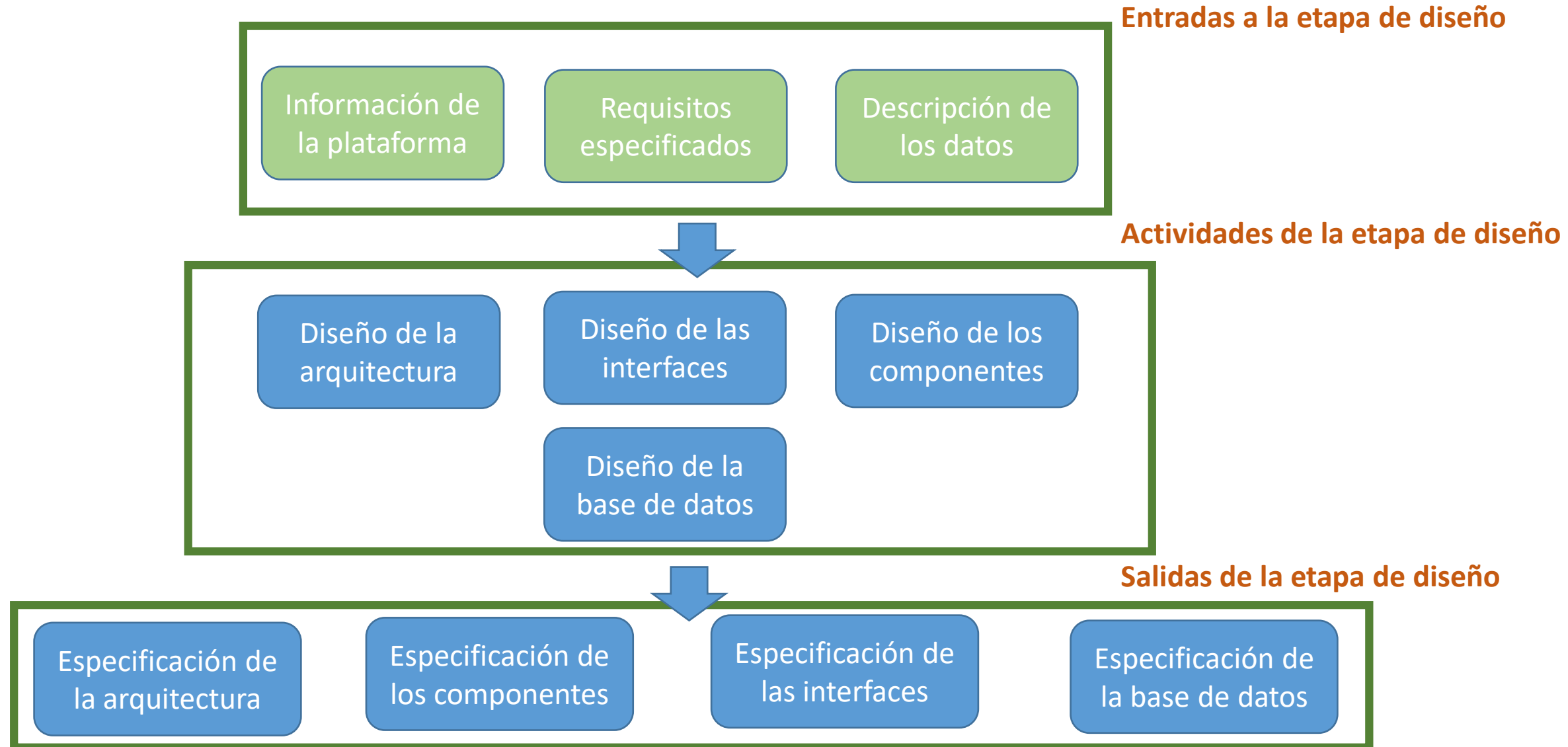
Agenda

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura

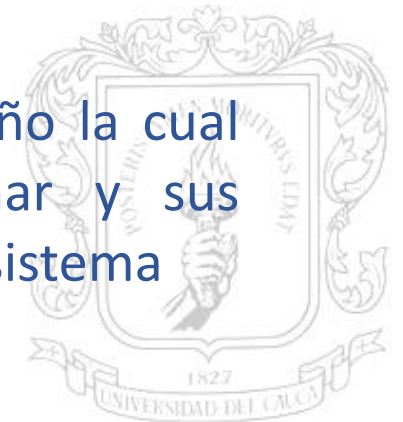


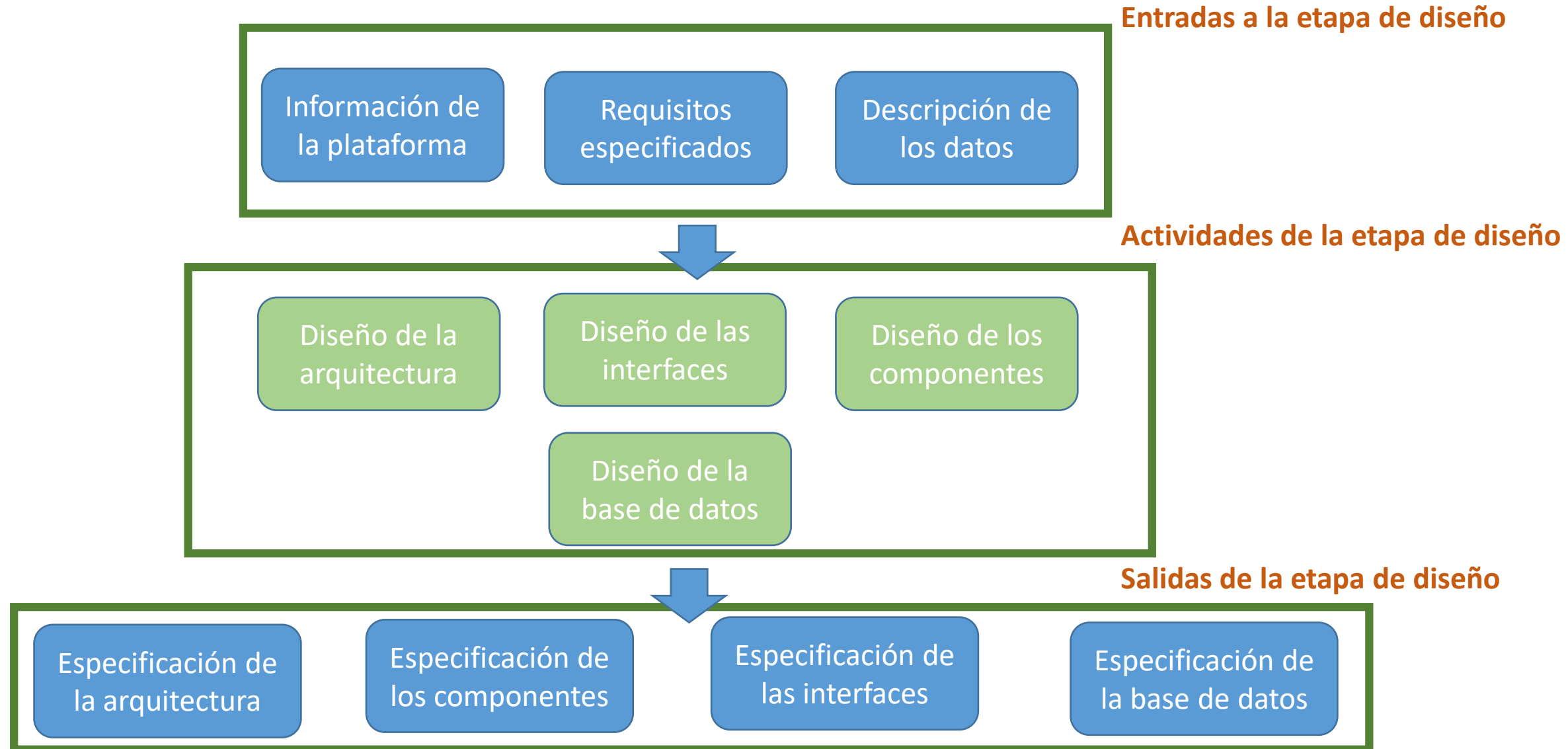




Entradas a la etapa de diseño

- ❖ La información de la plataforma corresponde a información acerca del entorno donde se ejecutará el software, tal como **sistema operativo, base de datos, servidor de aplicaciones, formatos para el intercambio de datos (JSON, XML)**
- ❖ La especificación de requisitos es una descripción de los servicios **(RF)** Y restricciones o condiciones de operación del sistema **(RNF)** tales como mantenibilidad, escalabilidad, confiabilidad.
- ❖ La descripción de los datos puede ser una entrada al proceso de diseño la cual consiste en un conjunto de **definiciones de los datos** a gestionar y sus características, con el propósito de que se defina el modelo de datos del sistema





Actividades de la etapa de diseño

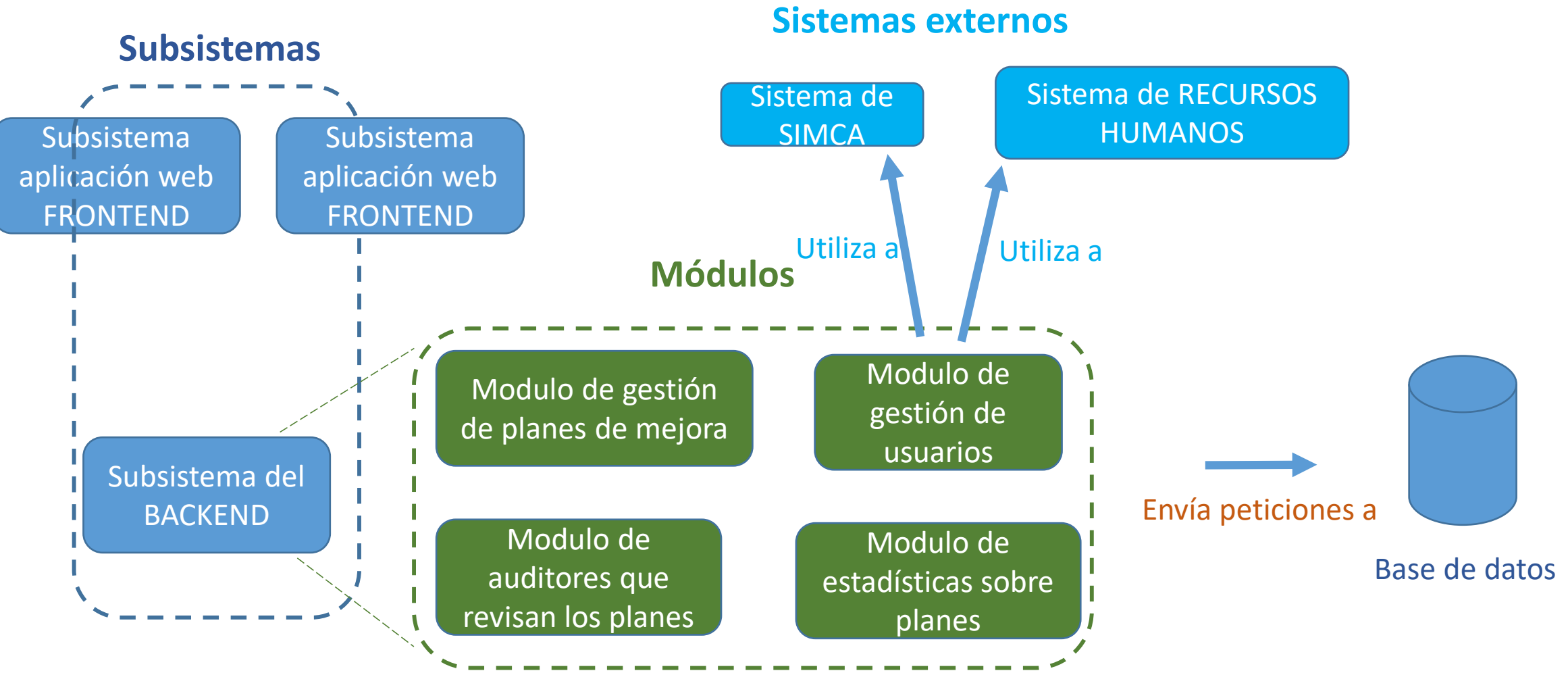
- ❖ **Diseño arquitectónico.** Se identifican los subsistemas, módulos sus relaciones y cómo se distribuyen.

Los subsistemas **inicialmente** pueden ser representados mediante un diagrama de bloques que muestra los principales subsistemas y la interconexión entre ellos.

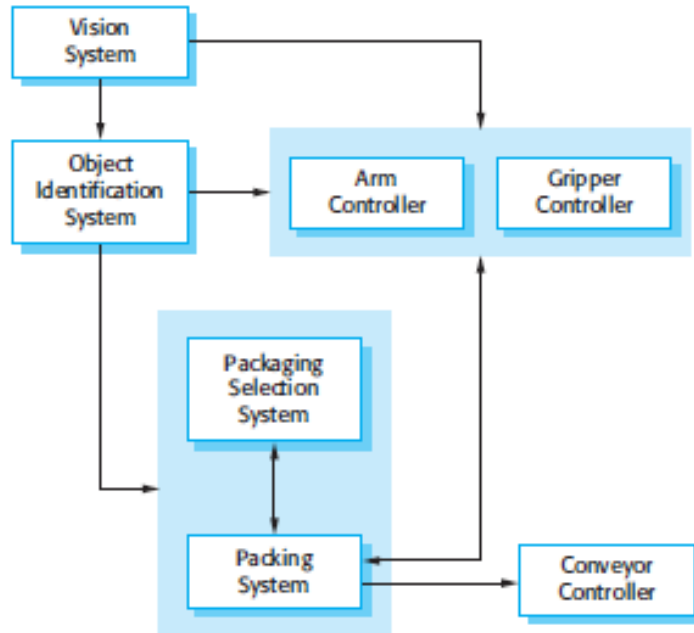
El diagrama proporciona al lector una visión general de la organización del sistema.

Definición de arquitectura

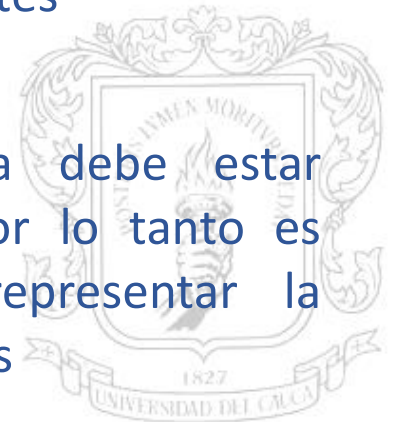
UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS



Las arquitecturas del sistema a menudo se modelan utilizando diagramas de bloques simples.



- ❖ Alto nivel de la estructura del sistema
- ❖ En una etapa temprana son validas
- ❖ Pero son representaciones deficientes
- ❖ La arquitectura de un sistema debe estar completamente documentada, por lo tanto es mejor usar una notación y representar la arquitectura desde diferentes vistas



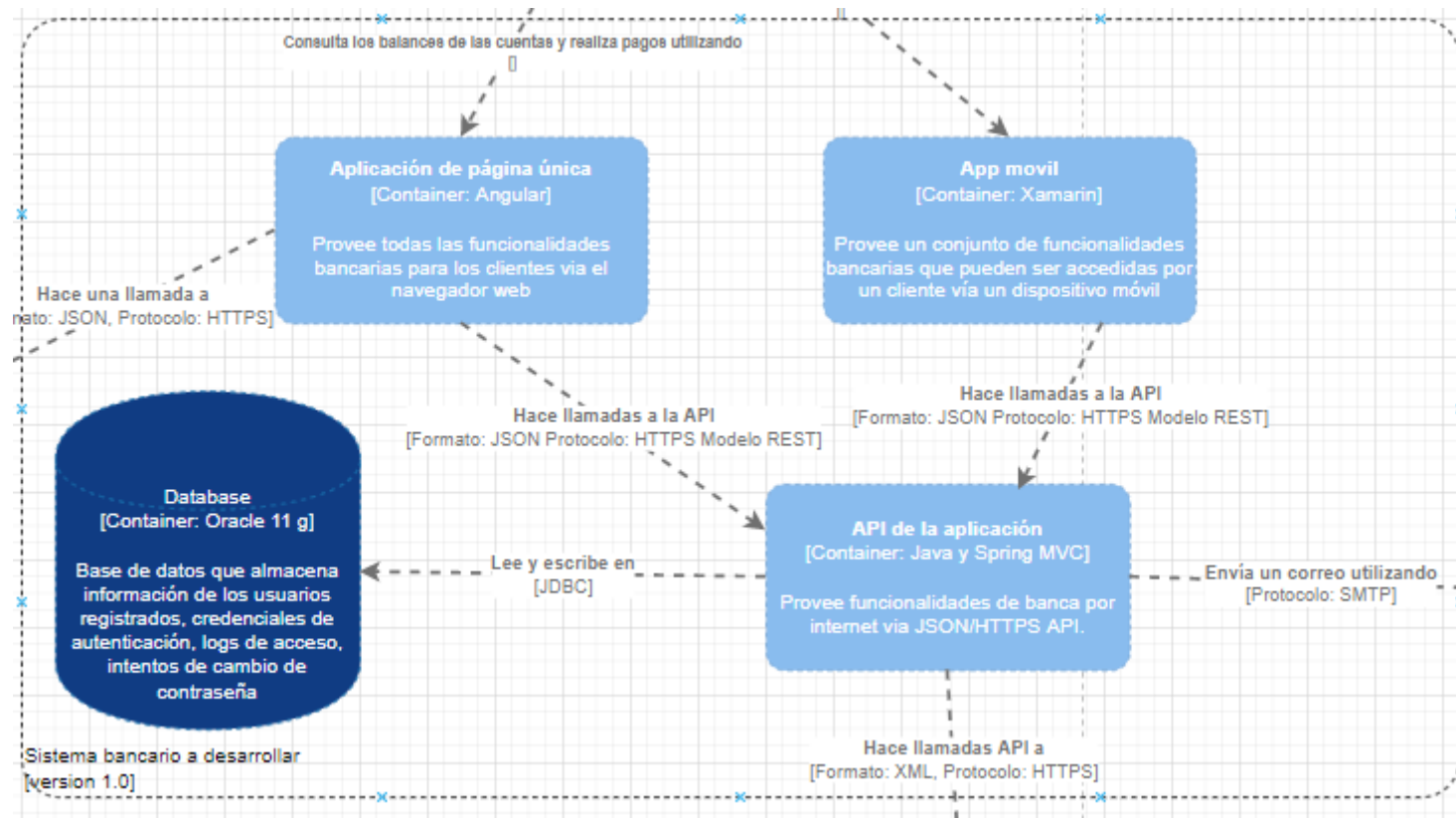
Actividades de la etapa de diseño

- ❖ **Diseño de interfaz.** Se definen las interfaces entre los componentes del sistema. Con una interfaz precisa, un componente se puede utilizar sin que otros componentes tengan que saber cómo se implementa.
- ❖ **Diseño de componentes.** Se aborda cada componente del sistema y se diseña cómo funcionará. Esta puede ser una simple declaración de la funcionalidad que se espera implementar, dejando el diseño específico al programador.
- ❖ **Diseño de base de datos.** Se diseñan las estructuras de datos del sistema y cómo se van a representar en una base de datos. Por ej: modelo entidad relación, modelo relacional



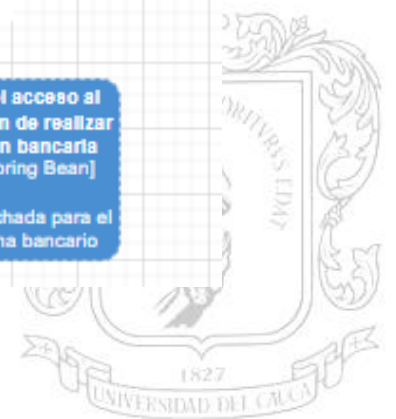
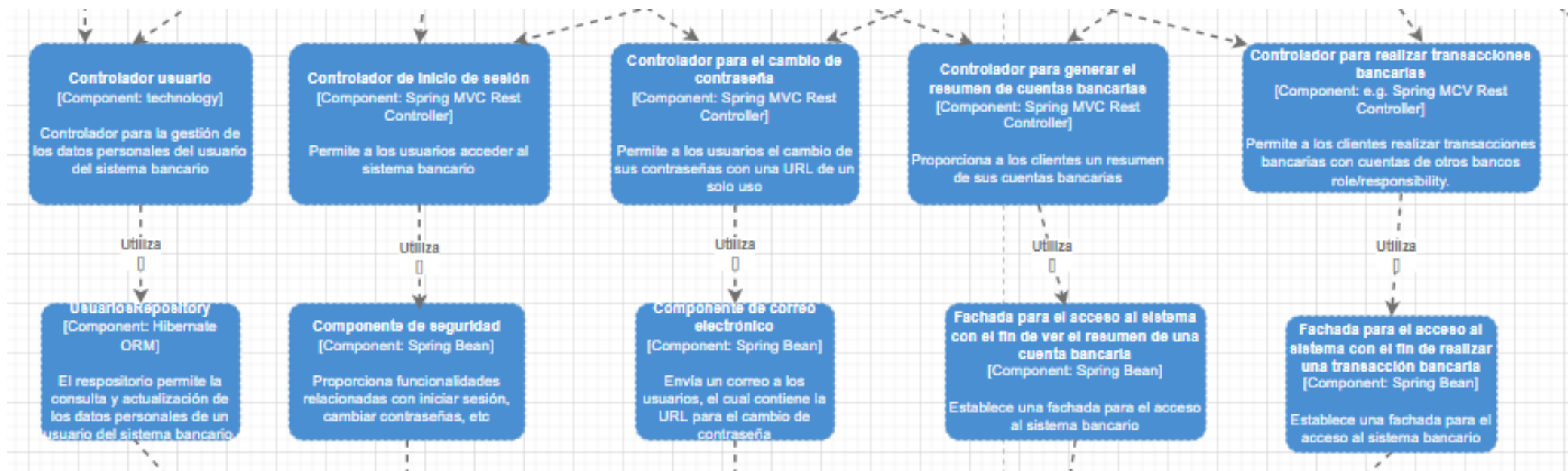
Actividades de la etapa de diseño

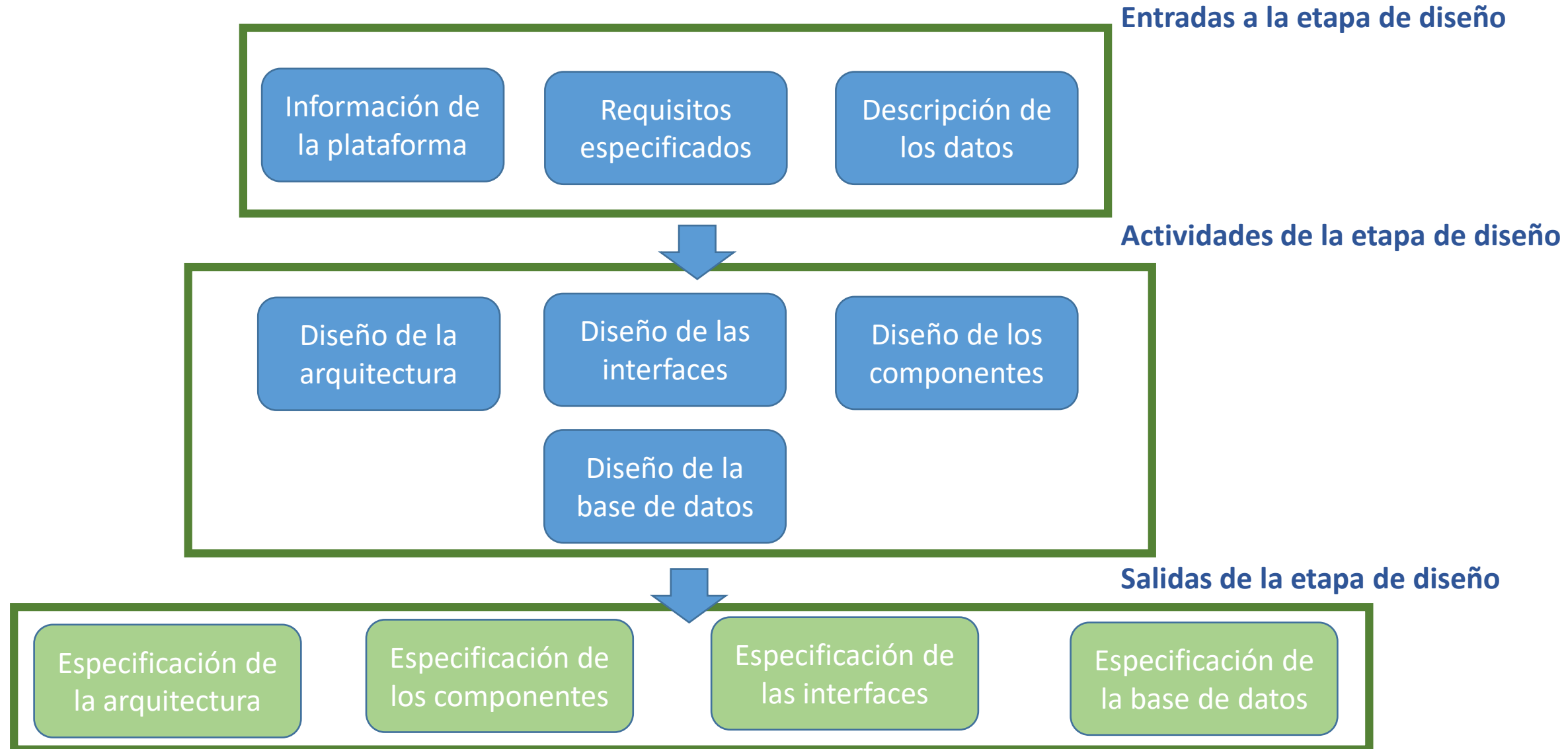
❖ Ejemplo de diseño de interfaz.



Actividades de la etapa de diseño

❖ Ejemplo de diseño detallado





Salidas de la etapa de diseño

- ❖ El detalle y la representación de los resultados varían considerablemente.
- ❖ La especificación (Descripción) se presenta mediante un conjunto de modelos.
- ❖ Para los sistemas críticos, los documentos de diseño detallados que establecen descripciones precisas y exactas del sistema deben ser producido.



Agenda

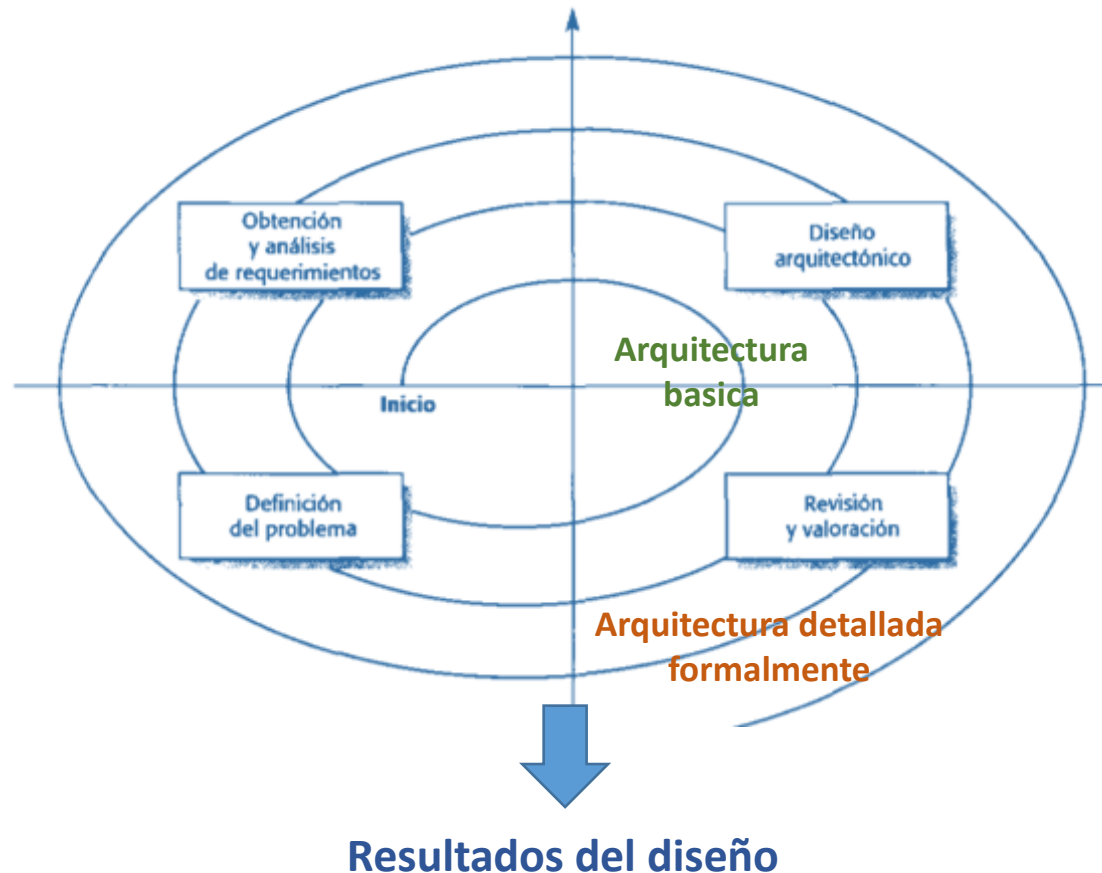
UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura



Relación entre la ingeniería de requisitos y la etapa de diseño

- ❖ Aunque se han separado los procesos de ingeniería de requerimientos y de diseño, en la práctica están **inextricablemente relacionados**.



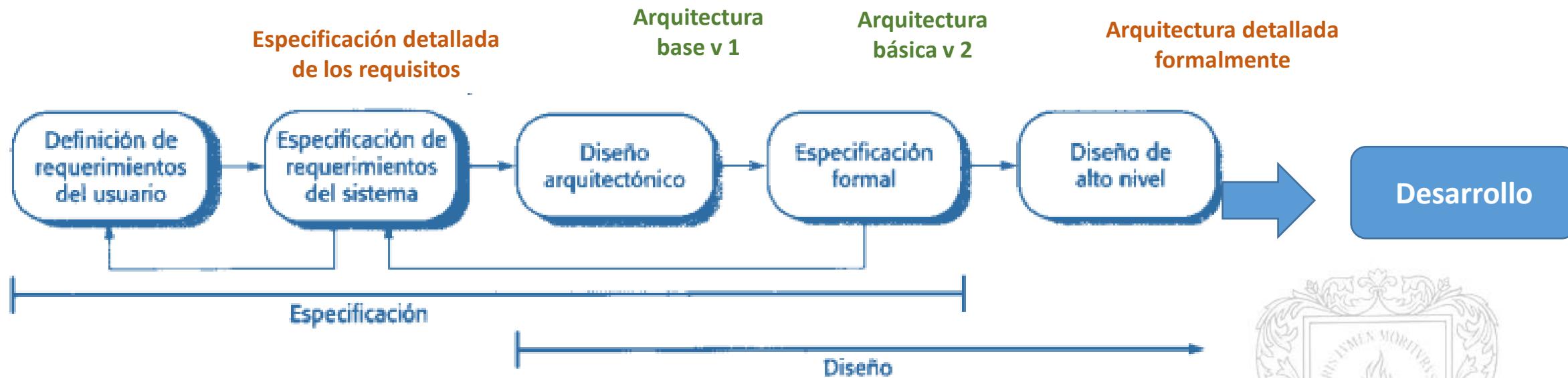
Comenzando en el centro, cada vuelta de la espiral añade algún detalle a los requisitos y al diseño. Algunas vueltas se centran en los requisitos; otras en el diseño.

Los diseñadores no llegan a un diseño terminado de inmediato, sino que desarrollan el diseño de manera iterativa.



Relación entre la ingeniería de requisitos y la etapa de diseño

Tanto los requisitos del sistema como el diseño se expresan con detalle y son analizados cuidadosamente antes de que comience la implementación



Puede usar esta descomposición para discutir los requisitos y características del sistema con las partes interesadas, puede descubrir problemas con los requisitos y pueden surgir nuevos requisitos.



Agenda

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura

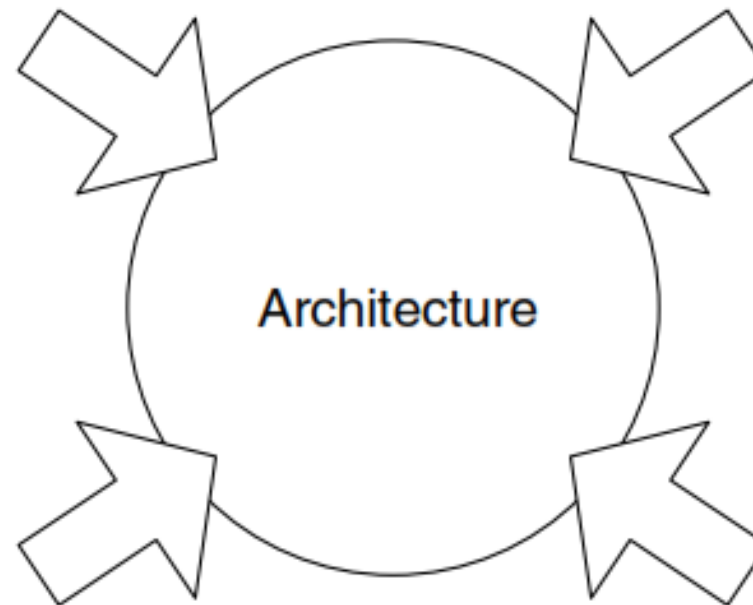


Requisitos funcionales

- Gerentes
- Usuarios finales
- Otros involucrados

Requisitos no funcionales

- Rendimiento
- Escalabilidad
- Reusabilidad



Experiencia en

- Arquitecturas existentes
- Patrones

Aspectos técnicos

- Sistemas legados
- Middleware
- Otros

(Jacobson et al. 1999)

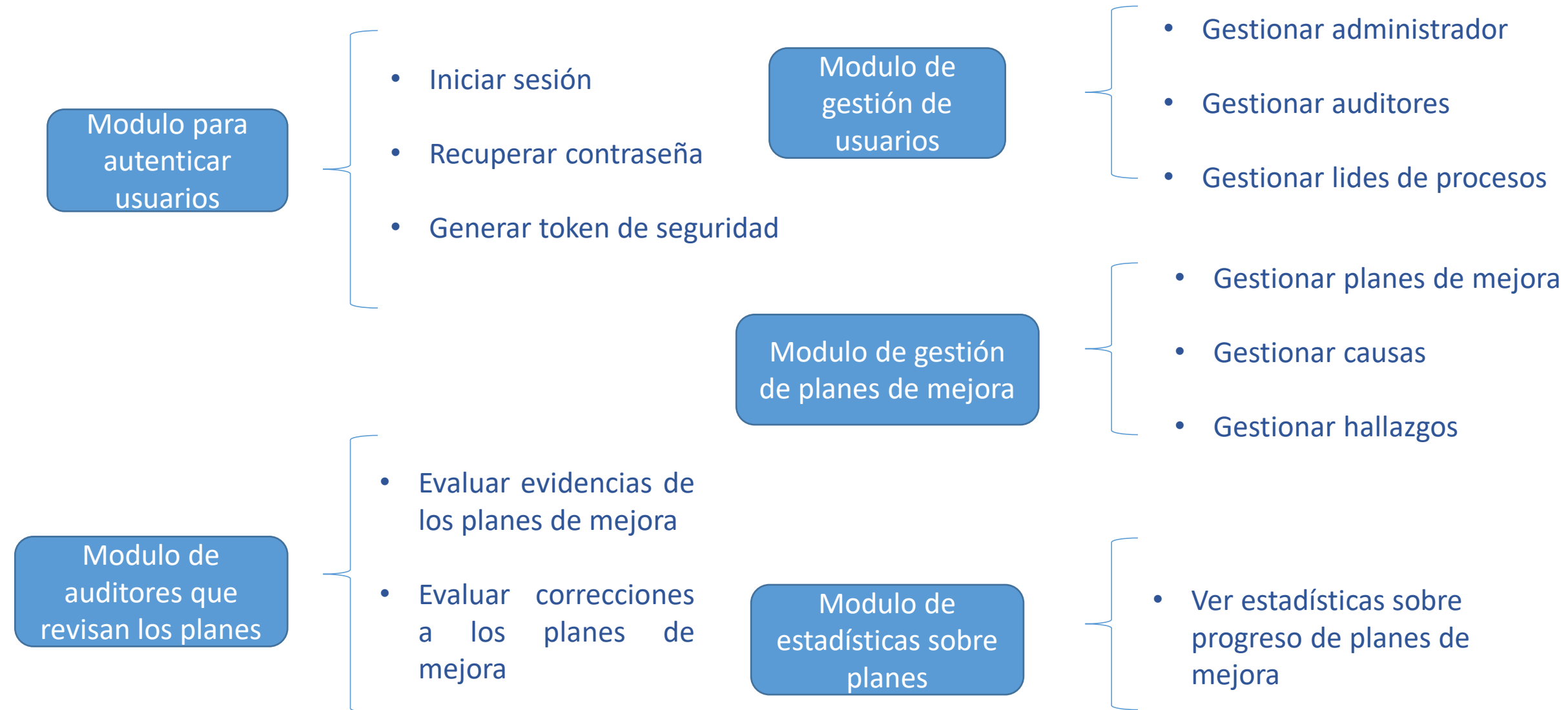
Considerar los **requisitos funcionales** para identificar los componentes. Debemos realizar las siguientes actividades

- a) Identificar los diferentes subsistemas que pueden. individual o colectivamente, cumplir los requisitos
- b) Asignar los requisitos a los subsistemas de tal manera que compartan características comunes
- d) Especificar las relaciones entre los subsistemas
- e) Definir las interfaces necesarias y requeridas por cada subsistema. Una vez que estas interfaces se han acordado, es posible desarrollar estos subsistemas en paralelo.



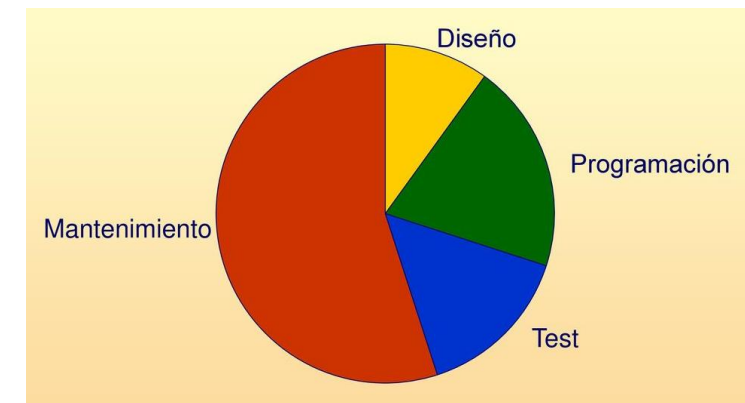
Factores que influyen el desarrollo de la arquitectura

UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS



La arquitectura debe depender de los **requisitos no funcionales** del sistema.

- ❖ **Seguridad.** Si la seguridad es un requisito crítico, se debe utilizar una estructura en capas para la arquitectura, con los activos más críticos protegidos en las capas más internas, con un alto nivel de validación de seguridad aplicada a estas capas.
- ❖ **Capacidad de mantenimiento.** Si la capacidad de mantenimiento es un requisito crítico, la arquitectura del sistema debe diseñarse utilizando componentes independientes y de grano fino que puedan cambiarse fácilmente.
- ❖ **Disponibilidad.** Si la disponibilidad es un requisito crítico, la arquitectura debe diseñarse para incluir componentes redundantes para que sea posible reemplazar y actualizar componentes sin detener el sistema.



La arquitectura debe depender de los **requisitos no funcionales** del sistema. Ha que considerar los diversos tipos de requisitos no funcionales

ISO/IEC 25010



La arquitectura de un sistema de software puede basarse en un conjunto de patrones en particular.

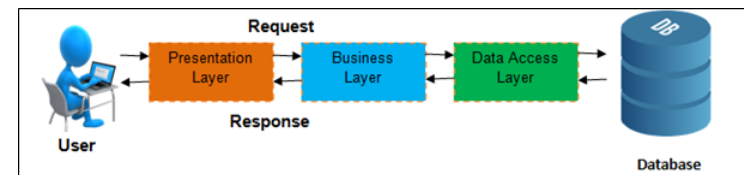
Un patrón arquitectónico es una descripción de la organización de un sistema (Garlan y Shaw, 1993).

Patrón capas

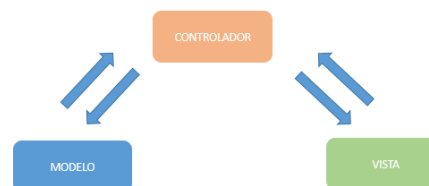


Patrón niveles

Patrón cliente - servidor



Patrón MVC



Agenda

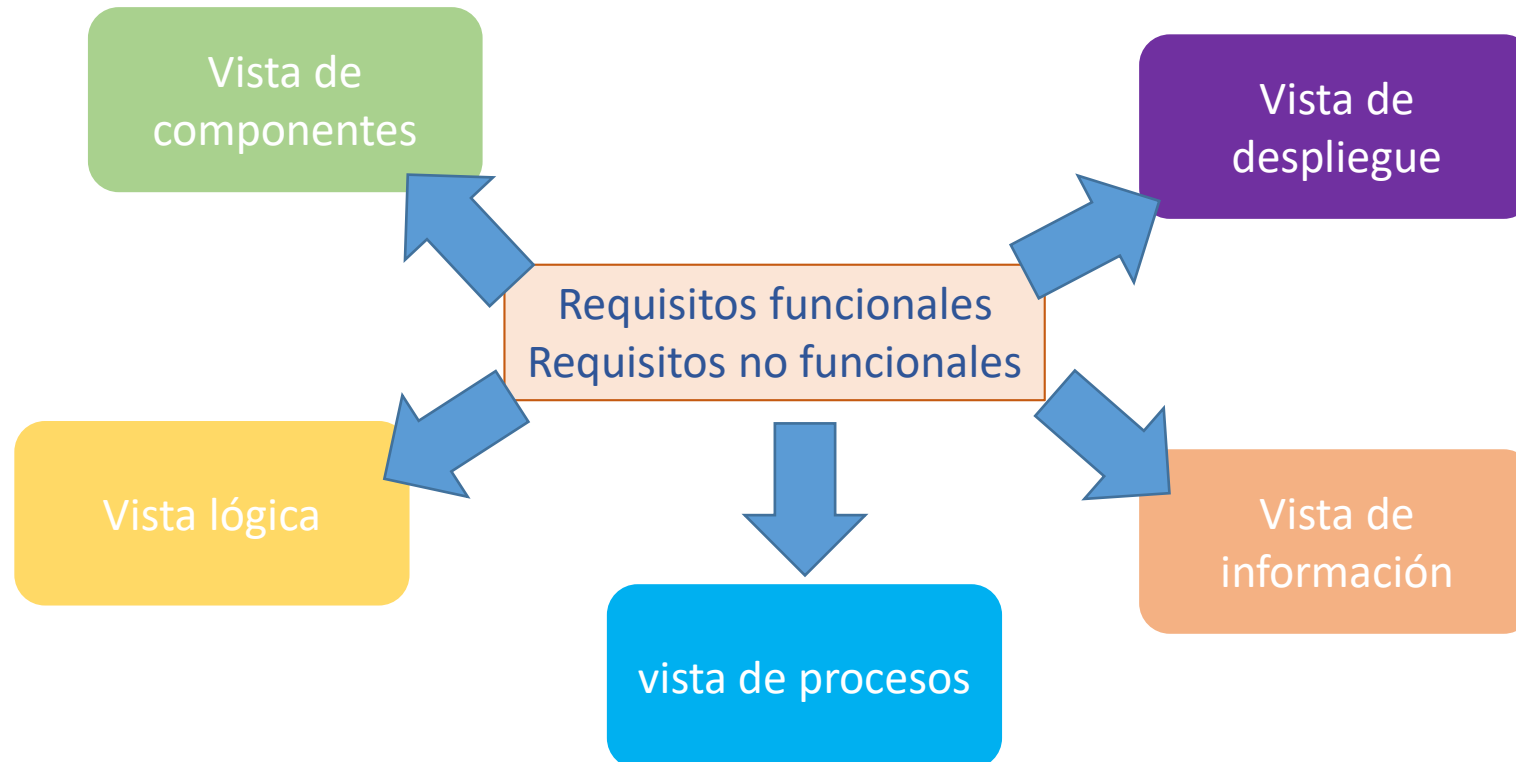
UNIVERSIDAD DEL CAUCA – FIET
DEPARTAMENTO DE SISTEMAS

- 1) Definición de la arquitectura
- 2) Ciclo de vida del software
- 3) Etapa de diseño de software
- 4) Relación entre la ingeniería de requisitos y la etapa de diseño
- 5) Factores que influyen en el desarrollo de la arquitectura
- 6) Vistas de la arquitectura



Es imposible representar toda la información relevante sobre la arquitectura de un sistema en un solo modelo arquitectónico, ya que cada modelo solo muestra una vista o perspectiva del sistema

Normalmente es necesario presentar varias vistas de la arquitectura del software.



Vista de componentes. Muestra cómo se descompone un sistema en subsistemas, módulos y componentes y cómo interactúan entre sí. La vista es útil para administradores y programadores de software.

Vista lógica. Muestra las abstracciones clave en el sistema como clases, procedimientos, funciones, eventos etc.

Vista de despliegue. Muestra como se distribuyen los componentes software en diferentes computadores en la red. Esta vista es útil para los ingenieros de sistemas que planean la implementación de un sistema.

Vista de información. Muestra la estructura del medio de persistencia. Por ejemplo el modelo relacional en una base de datos.

vista de procesos. Muestra en tiempo de ejecución, como los componentes interactúan entre sí. Pueden utilizarse diagramas de secuencia



Ventajas de diseñar y documentar explícitamente las vistas de la arquitectura:

- **Comunicación con las partes interesadas.** La arquitectura en un modelo de alto nivel puede ser utilizada como foco de discusión por una variedad de diferentes partes interesadas.
- **Realización de análisis de sistemas.** la arquitectura del sistema explícita en una etapa temprana en el desarrollo del sistema requiere algún análisis. Las decisiones de diseño arquitectónico tienen un efecto profundo sobre si el sistema puede cumplir o no con requisitos críticos como rendimiento, confiabilidad y facilidad de mantenimiento.
- **Reutilización a gran escala.** La arquitectura del sistema suele ser la misma para sistemas con requisitos similares y, por lo tanto, puede admitir la reutilización de software a gran escala.



Bibliografía

Murugesan, S., Deshpande, Y., Hansen, S., & Ginige, A. (2001). Web engineering: A new discipline for development of web-based systems. In Web Engineering (pp. 3-13). Springer, Berlin, Heidelberg.

Aliaga, C. J., & Quintero, A. I. L. (2018). Arquitecturas de microservicios para aplicaciones desplegadas en contenedores.

Sommerville, I. (2011). Software engineering 9th Edition. ISBN-10, 137035152, 18.



Muchas gracias

Preguntas

