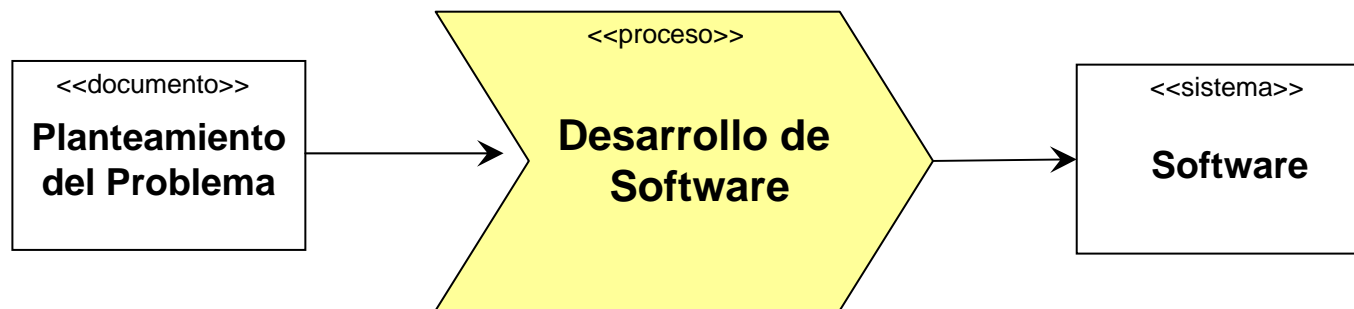


El Desarrollo de Software desde un enfoque de procesos



Proceso: conjunto de actividades interrelacionadas que permiten alcanzar un objetivo preestablecido

Prof. Gustavo Briceño / gbriceno@ula.ve
<http://webdelprofesor.ula.ve/ingenieria/gbriceno>

Para desarrollar software es necesario:

- Entender la **naturaleza del software**
- Utilizar un **proceso de desarrollo** bien definido y probado (adaptado a las características del software a desarrollar)
- **Gestionar el desarrollo** de software como un proyecto de ingeniería

El **proceso de desarrollo de software** "es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo". Concretamente "define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo" [Jacobson 1998].

Naturaleza del software

El software es un conjunto de programas, procedimientos, documentos y datos asociados a un sistema de computación para: dar solución a un problema, ejecutar una actividad en forma automática o satisfacer requerimientos a un conjunto de usuarios

El software es un objeto abstracto producido por el intelecto del hombre que tiene las siguientes propiedades: No es visible, no es tangible; No tiene propiedades físicas; No se deteriora, ni desgasta con el tiempo; Tiene una estructura modificable (esta sujeto a cambios continuos, periódicos o no); **Las modificaciones continuas ocasionan con el tiempo una pérdida de confiabilidad; Se desarrolla, no se construye** (sus componentes crecen progresivamente mediante adiciones y correcciones sucesivas de sus componentes); Su mantenimiento es complejo; **Su estructura evoluciona mediante modificaciones sucesivas.**

Los elementos fundamentales de un proyecto son:

Procesos (actividades)
Productos (intermedios, finales) } método
Recursos (humanos, financieros, tiempo, etc.)

Visto como un proyecto, el proceso de desarrollo de software debe controlar tres variables fundamentales: Costo, Tiempo y Calidad.

Estas variables se controlan mediante la aplicación de principios, procesos, prácticas y técnicas gerenciales.

Tipos de software

- *Aplicaciones* (sistemas de información, programas de aplicación, aplicaciones Web, etc.)
- *Sistemas de software* (sistemas operativos, compiladores, manejadores de bases de datos, etc.)

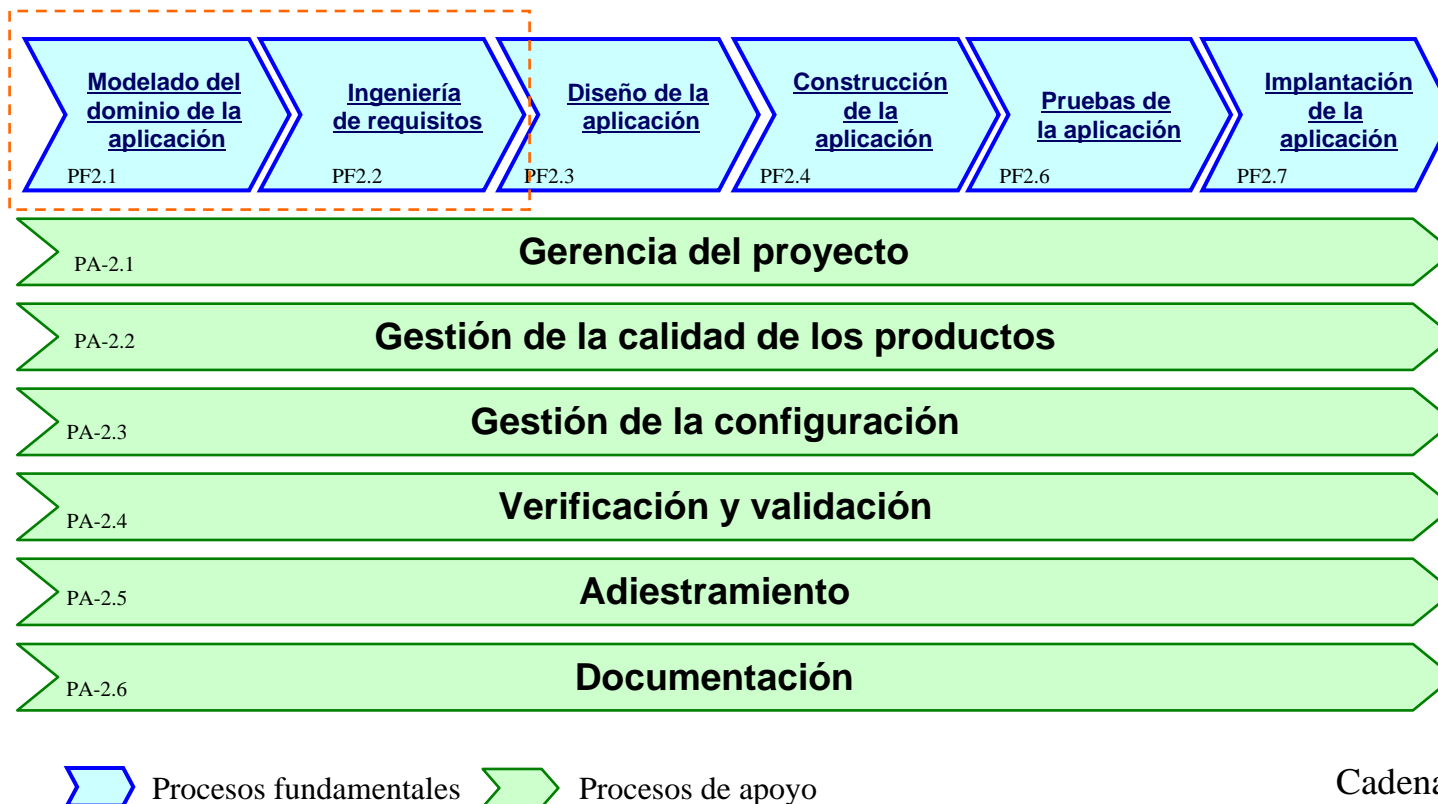
En la Ingeniería de Software, el proceso de desarrollo se realiza bajo la **modalidad de proyecto**.

Un proyecto tiene un conjunto de objetivos que deben ser alcanzados **dentro de restricciones** tales como: presupuesto, tiempo y disponibilidad de recursos.

Modelo del proceso de desarrollo de aplicaciones de la DSIA

Cadena de Valor

Versión 1.0 Mayo, 2006



Cadena de Valor
Michael Porter (2006)

[Diagrama de procesos](#)

[Proceso de desarrollo de aplicaciones](#)

[Modelado,
Ingeniería de Requisitos
y Diseño de la aplicación](#)

[Los modelos y el proceso de modelado](#)

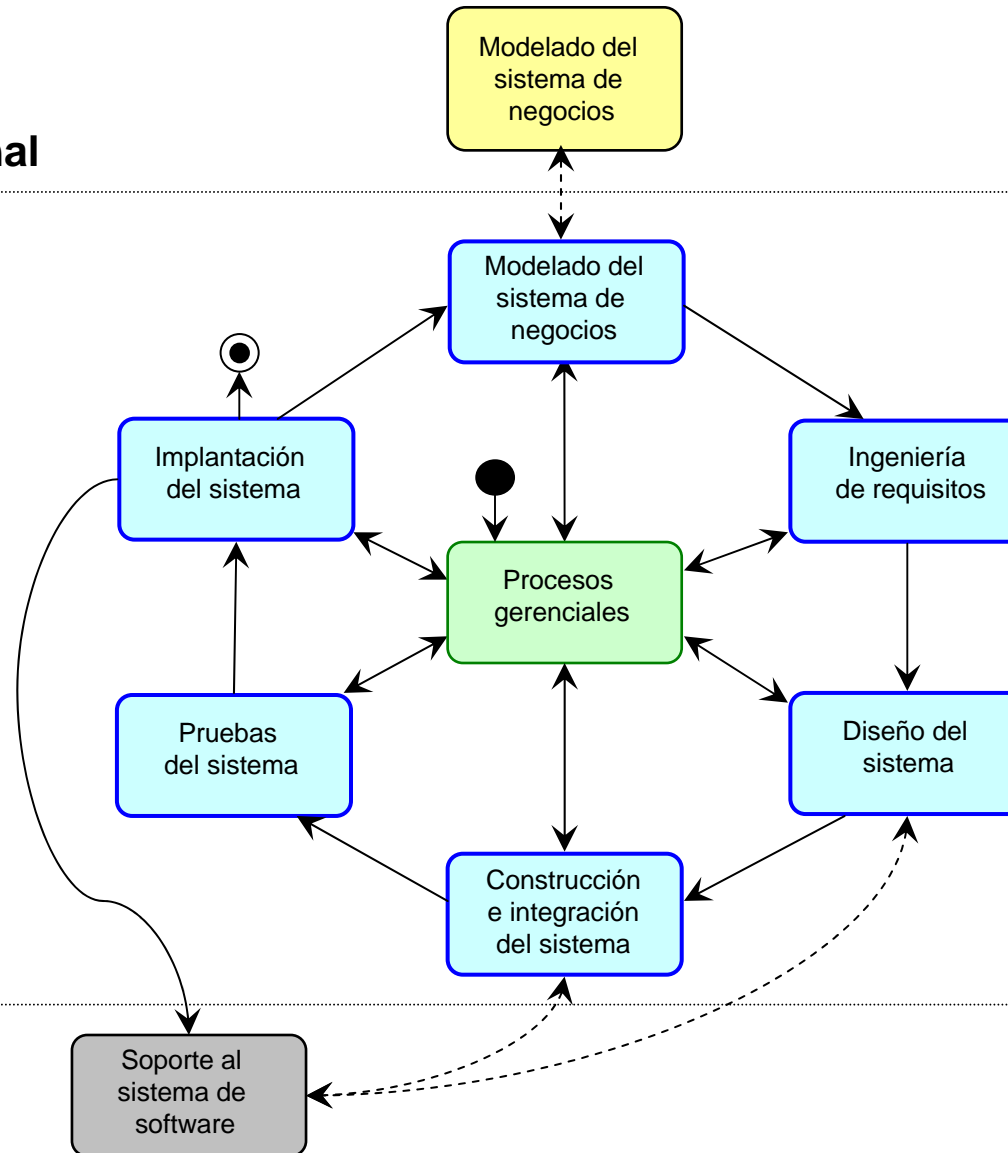
Proceso de desarrollo de software

Nivel organizacional

Nivel de sistema

Fases del proceso de desarrollo de software

Nivel de soporte

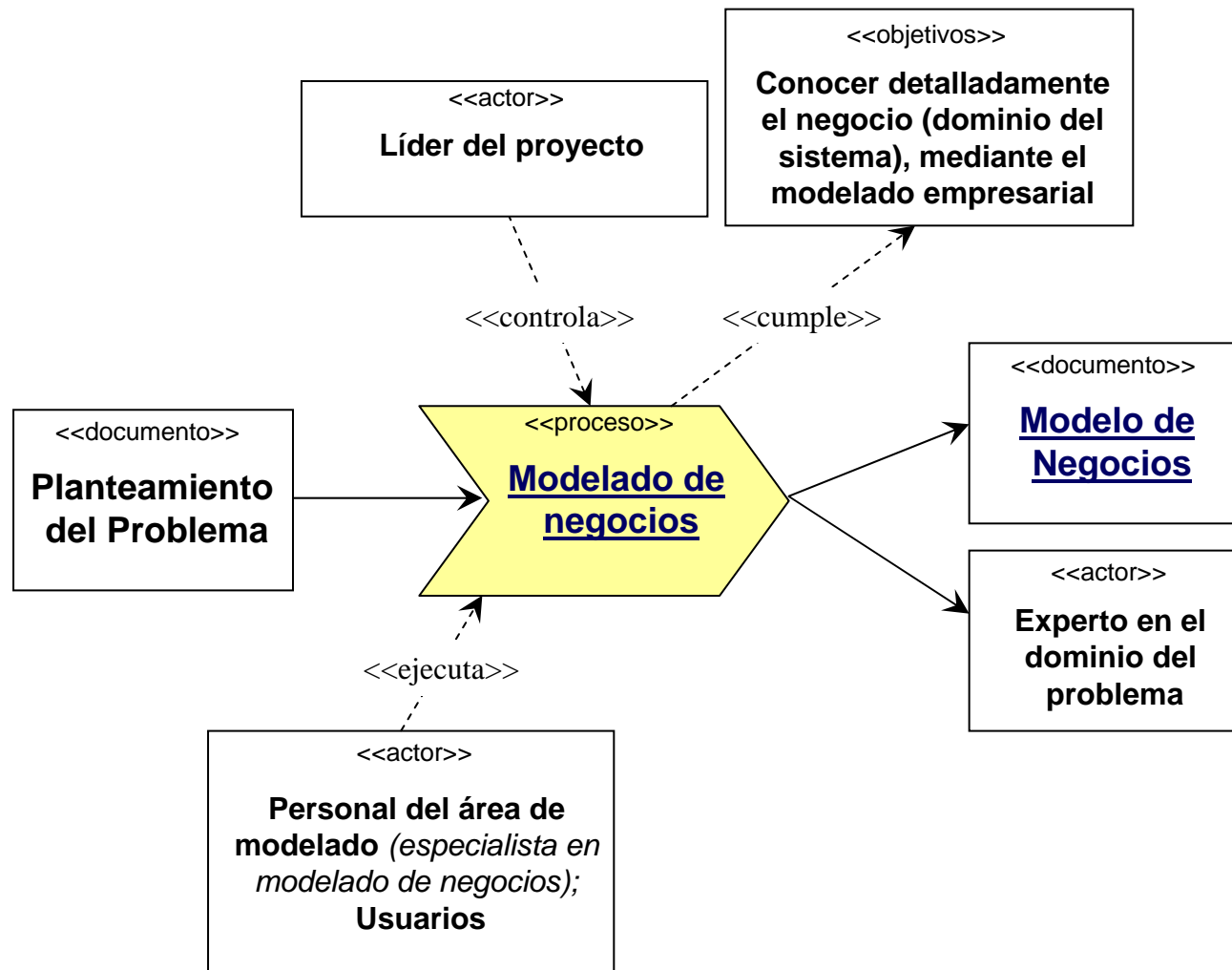


Las flechas con línea punteada se conectan con fases de soporte relacionadas con la administración de componentes de software, administración de Bases de Datos y de la seguridad y administración de Objetos del dominio.



Método de desarrollo de software

Fase: Modelado de negocios

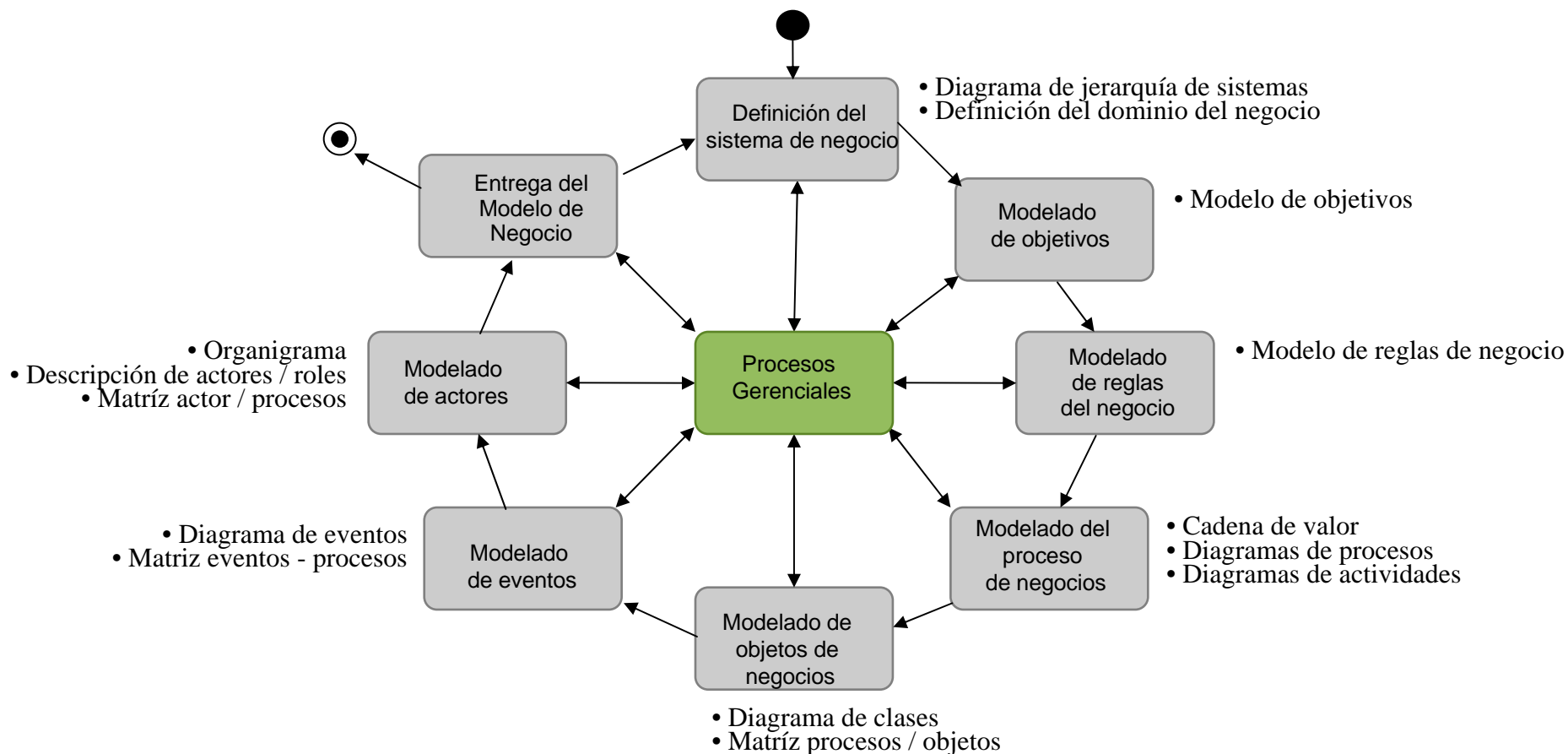


Método de desarrollo de software

Proceso: Modelado de negocios

Basado en el método BMM

Método de Modelado de Negocios



Método de desarrollo de software

Proceso: Modelado de negocios

Modelo de Negocios

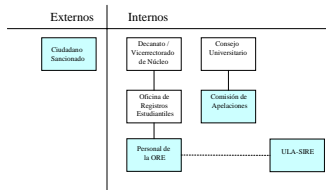
Jerarquía de Sistemas



Cadena de Valor

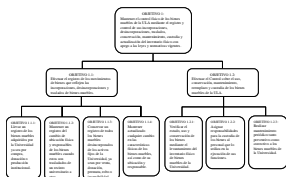


Modelo de Actores / Unidades



Modelo de producto

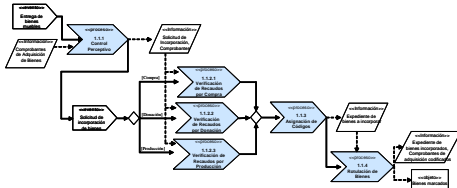
Modelo de Objetivos



Jerarquía de procesos

1. Identificación de la institución	1.1. Identificación de la institución
2. Gestión de admisiones	2.1. Gestión de admisiones
3. Gestión de trámites académicos	3.1. Gestión de trámites académicos
4. Gestión de trámites administrativos	4.1. Gestión de trámites administrativos
5. Gestión de trámites de grado	5.1. Gestión de trámites de grado
6. Gestión de trámites de grado	6.1. Gestión de trámites de grado
7. Gestión de trámites de grado	7.1. Gestión de trámites de grado
8. Gestión de trámites de grado	8.1. Gestión de trámites de grado
9. Gestión de trámites de grado	9.1. Gestión de trámites de grado
10. Gestión de trámites de grado	10.1. Gestión de trámites de grado
11. Gestión de trámites de grado	11.1. Gestión de trámites de grado
12. Gestión de trámites de grado	12.1. Gestión de trámites de grado
13. Gestión de trámites de grado	13.1. Gestión de trámites de grado
14. Gestión de trámites de grado	14.1. Gestión de trámites de grado
15. Gestión de trámites de grado	15.1. Gestión de trámites de grado
16. Gestión de trámites de grado	16.1. Gestión de trámites de grado
17. Gestión de trámites de grado	17.1. Gestión de trámites de grado
18. Gestión de trámites de grado	18.1. Gestión de trámites de grado
19. Gestión de trámites de grado	19.1. Gestión de trámites de grado
20. Gestión de trámites de grado	20.1. Gestión de trámites de grado

Modelo de eventos



Transformación de modelos

Modelo de Reglas de Negocio

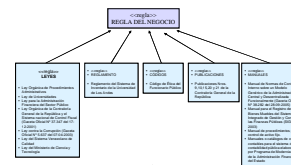


Diagrama de procesos

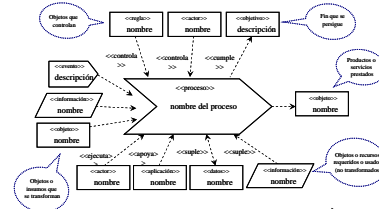


Diagrama de objetos

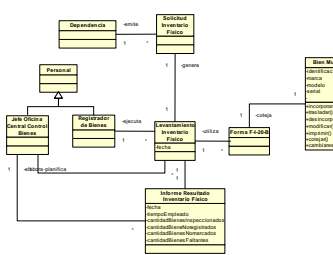
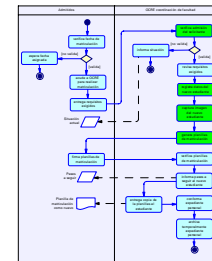
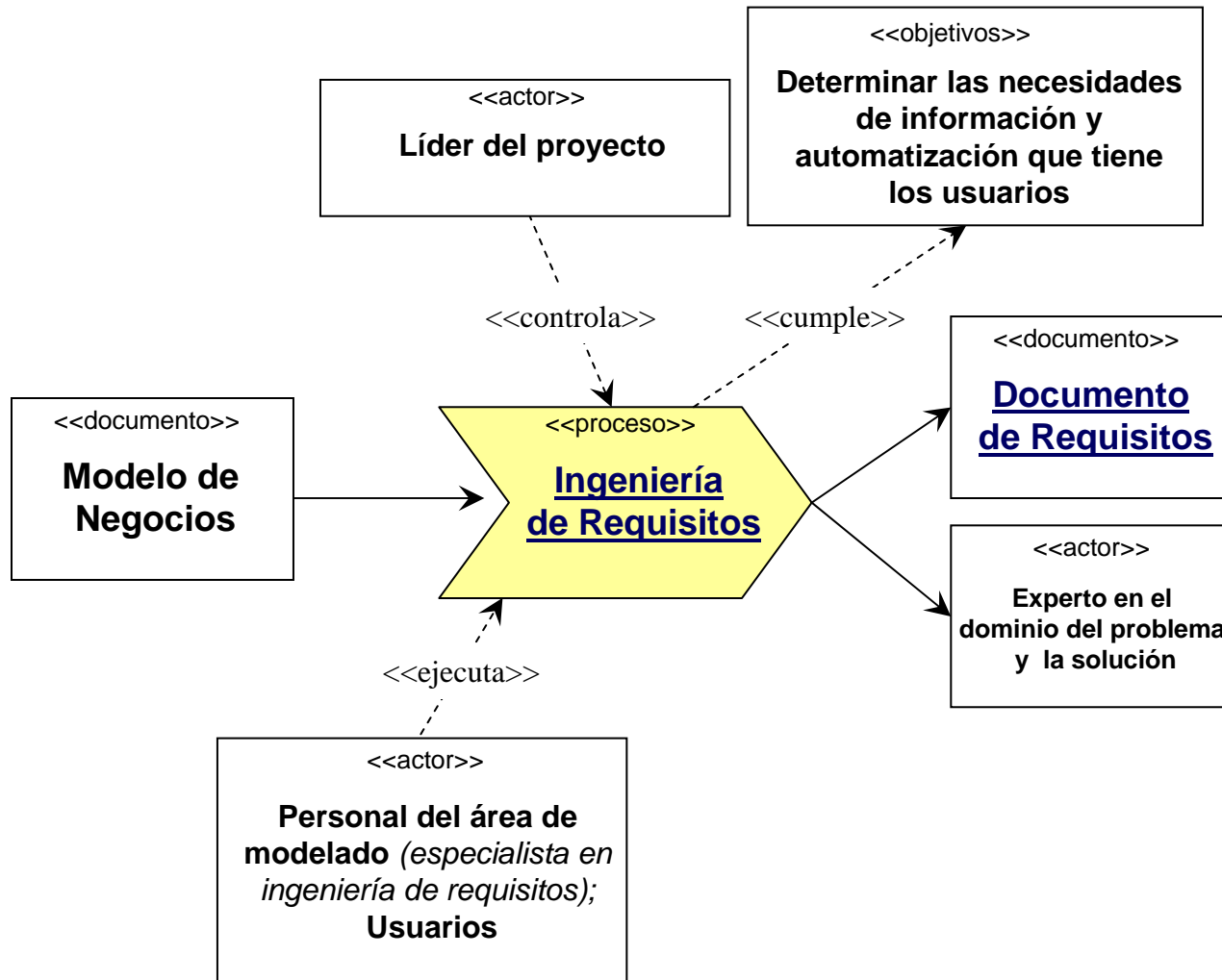


Diagrama de Actividades



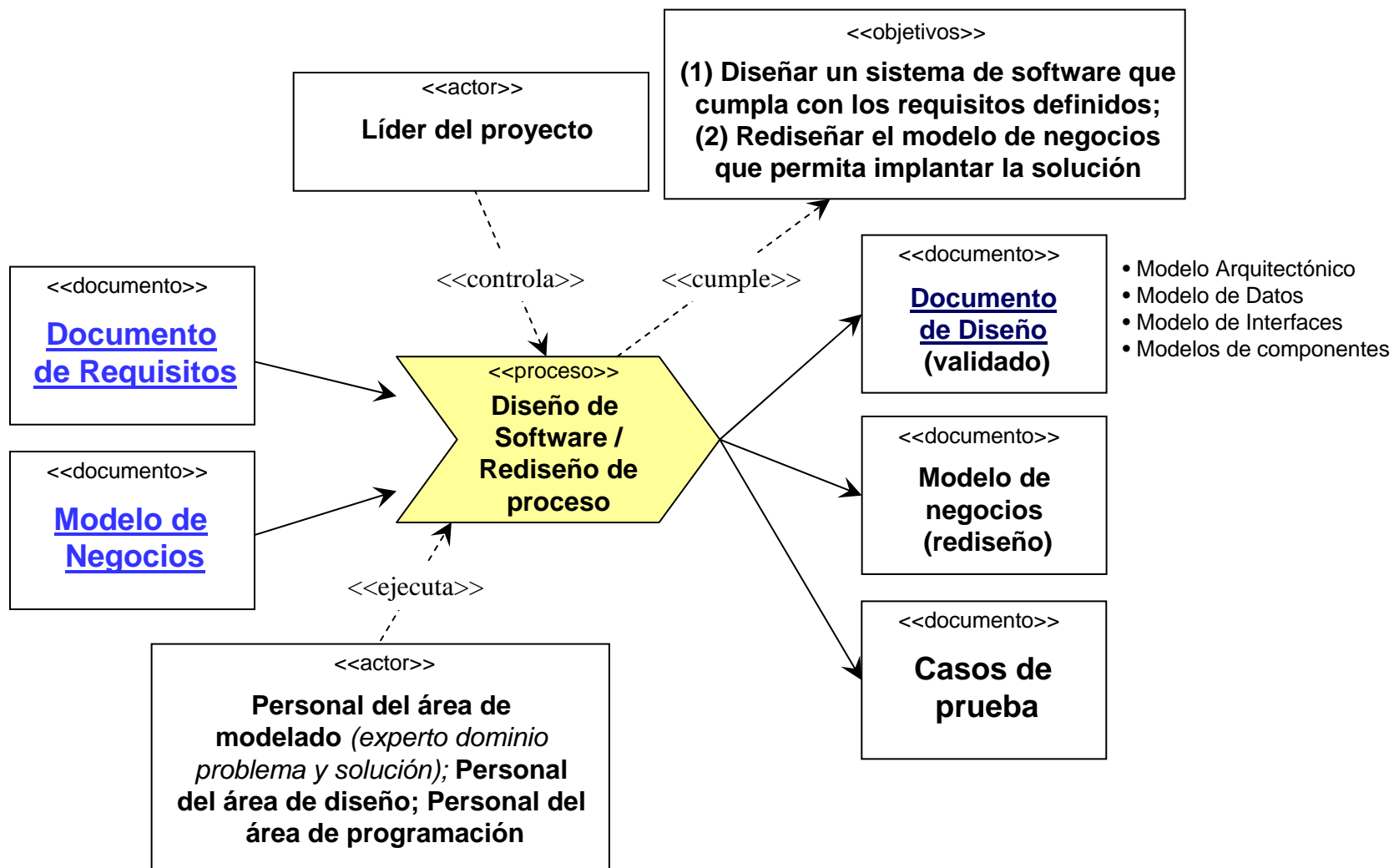
Método de desarrollo de software

Fase: Ingeniería de requisitos



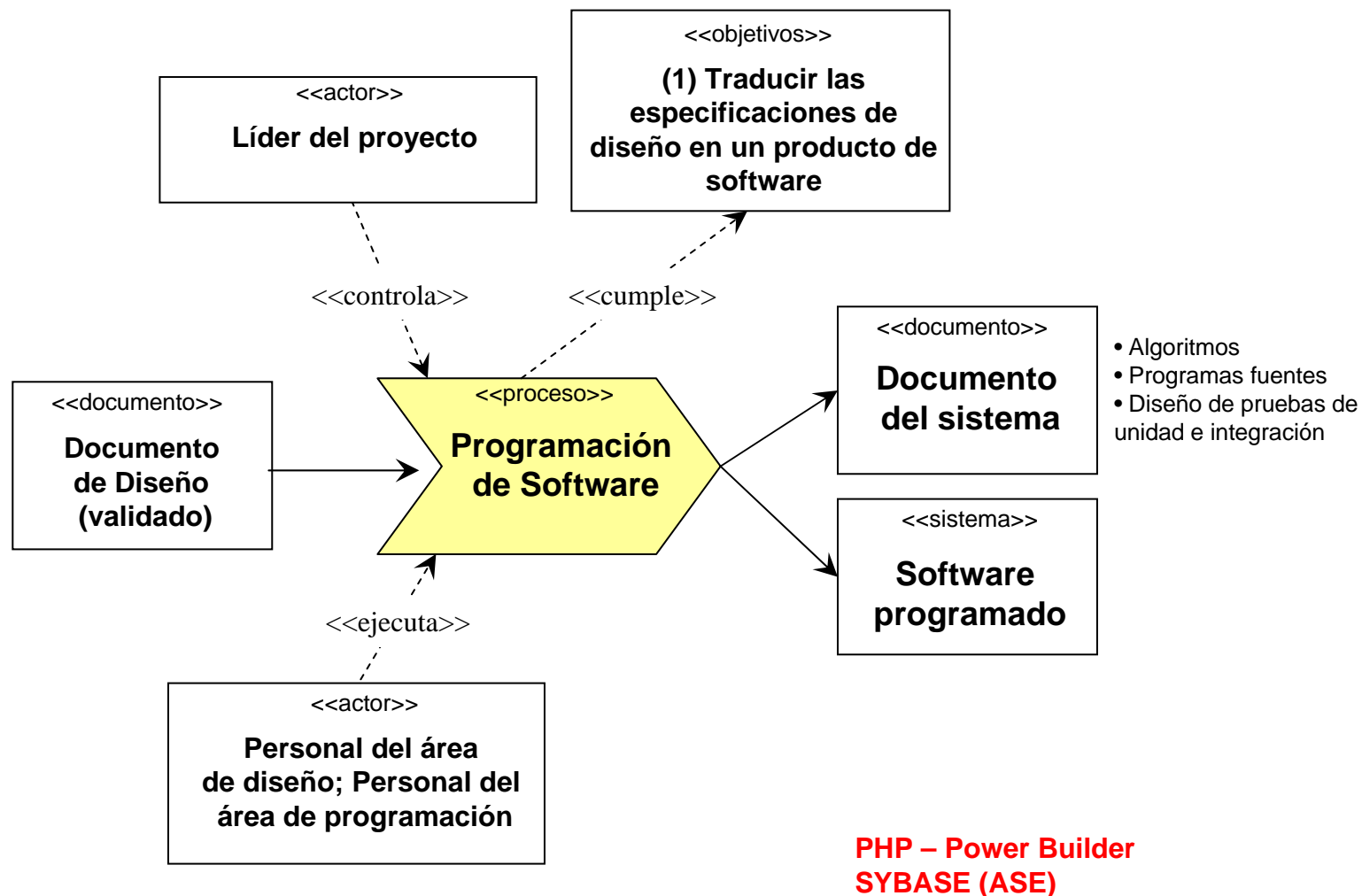
Método de desarrollo de software

Fase: Diseño de software y rediseño de procesos



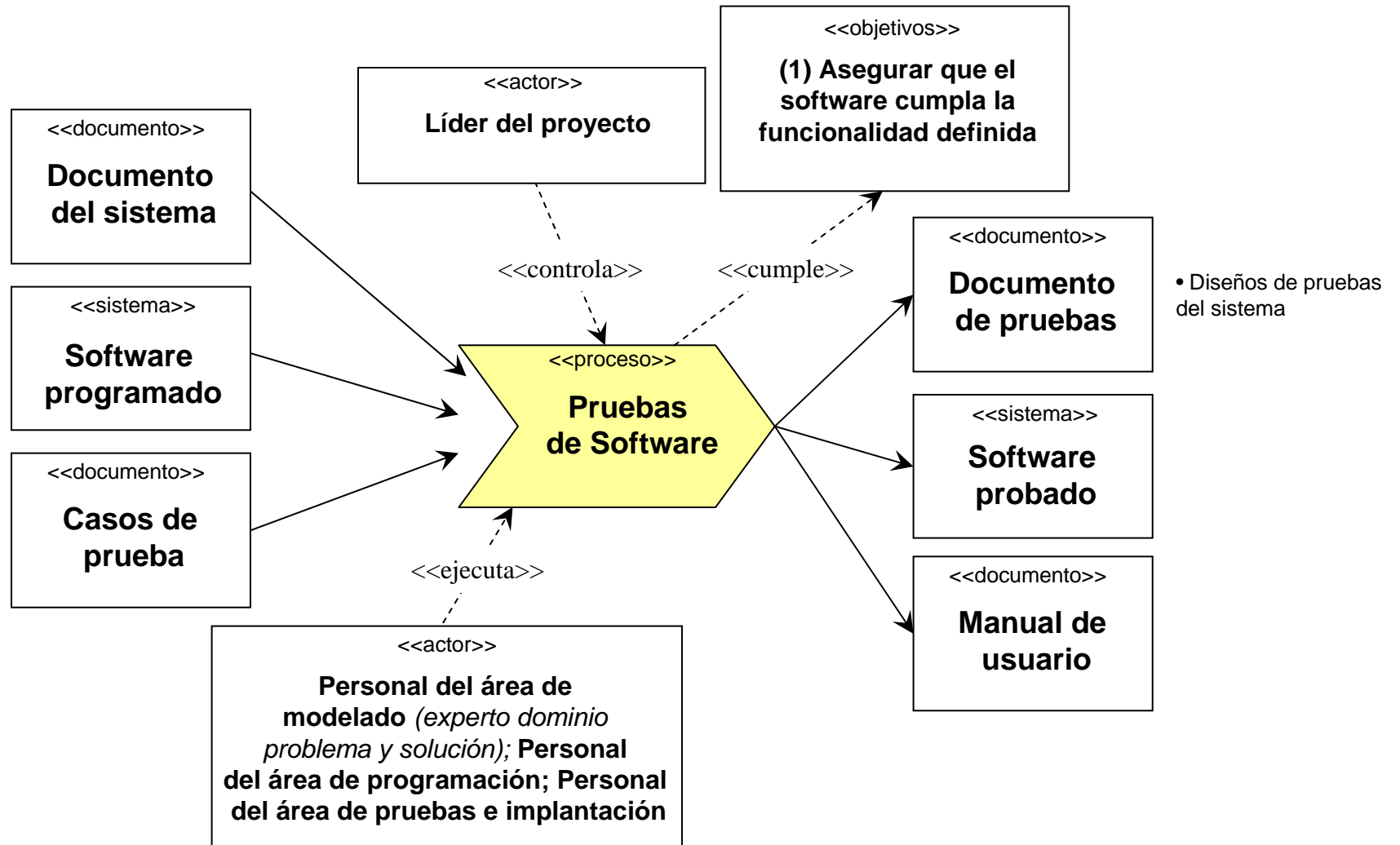
Método de desarrollo de software

Fase: Programación de software



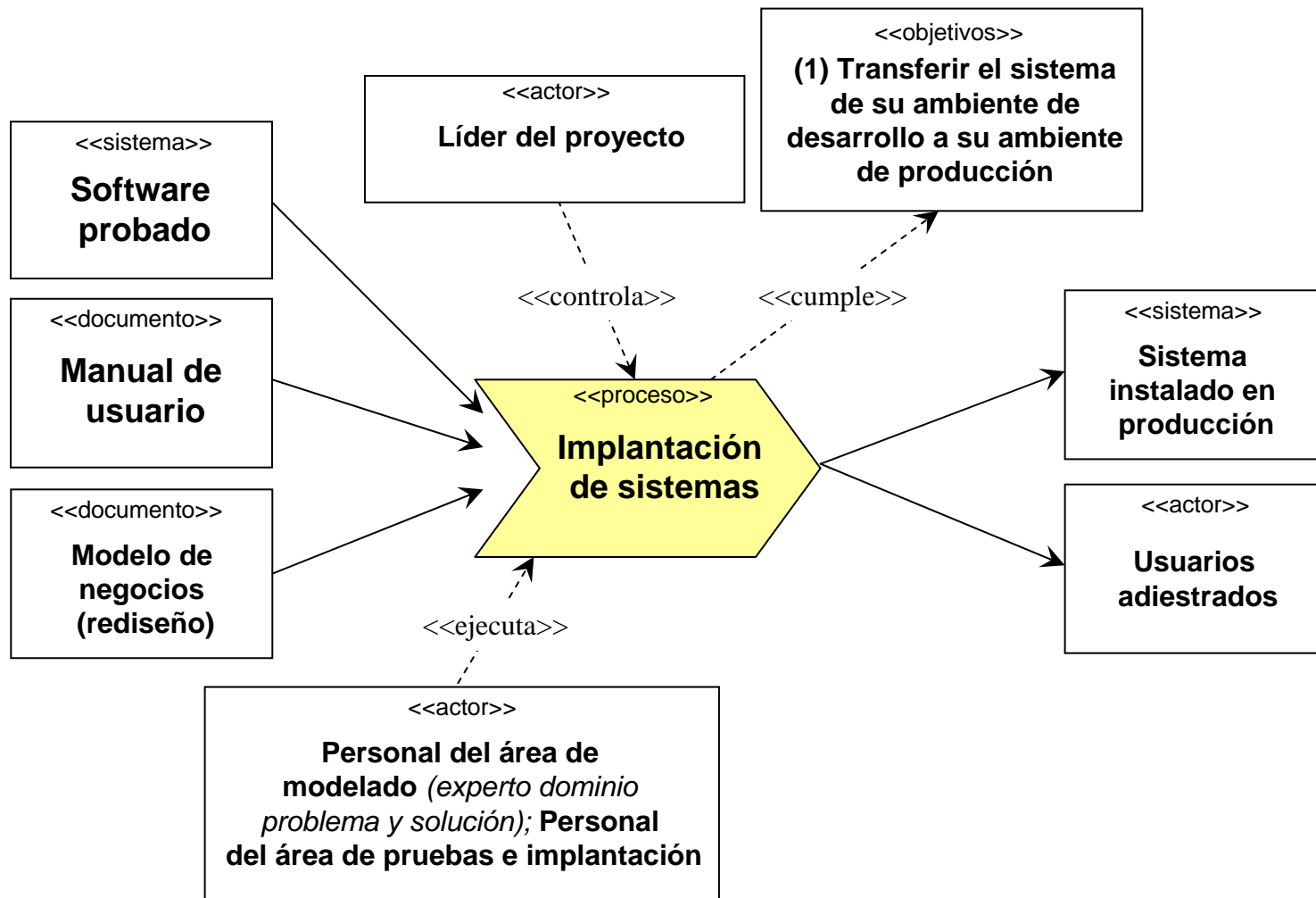
Método de desarrollo de software

Fase: Pruebas de software



Método de desarrollo de software

Fase: Implantación del sistema



Modelo: Es la representación abstracta de un sistema u objeto que se elabora para simplificar su comprensión u estudio.

Modelado: Proceso de adquisición y representación de conocimientos sobre un objeto de estudio dado, el cual, involucra tres conceptos: El objeto de estudio, el conocimiento (conceptos) y la representación (símbolos).

UML: Lenguaje Unificado de Modelado

El proceso de desarrollo de software se basa en la transformación de modelos



Método de trabajo

Método: Un método define de manera formal el conjunto de conceptos que se deben conocer y las actividades que se deben ejecutar para resolver un problema.

Un método está compuesto por un **modelo de producto** y un **modelo de proceso**.

El **modelo de producto** es el conjunto de conceptos que se pueden utilizar para construir un producto o sistema determinado.

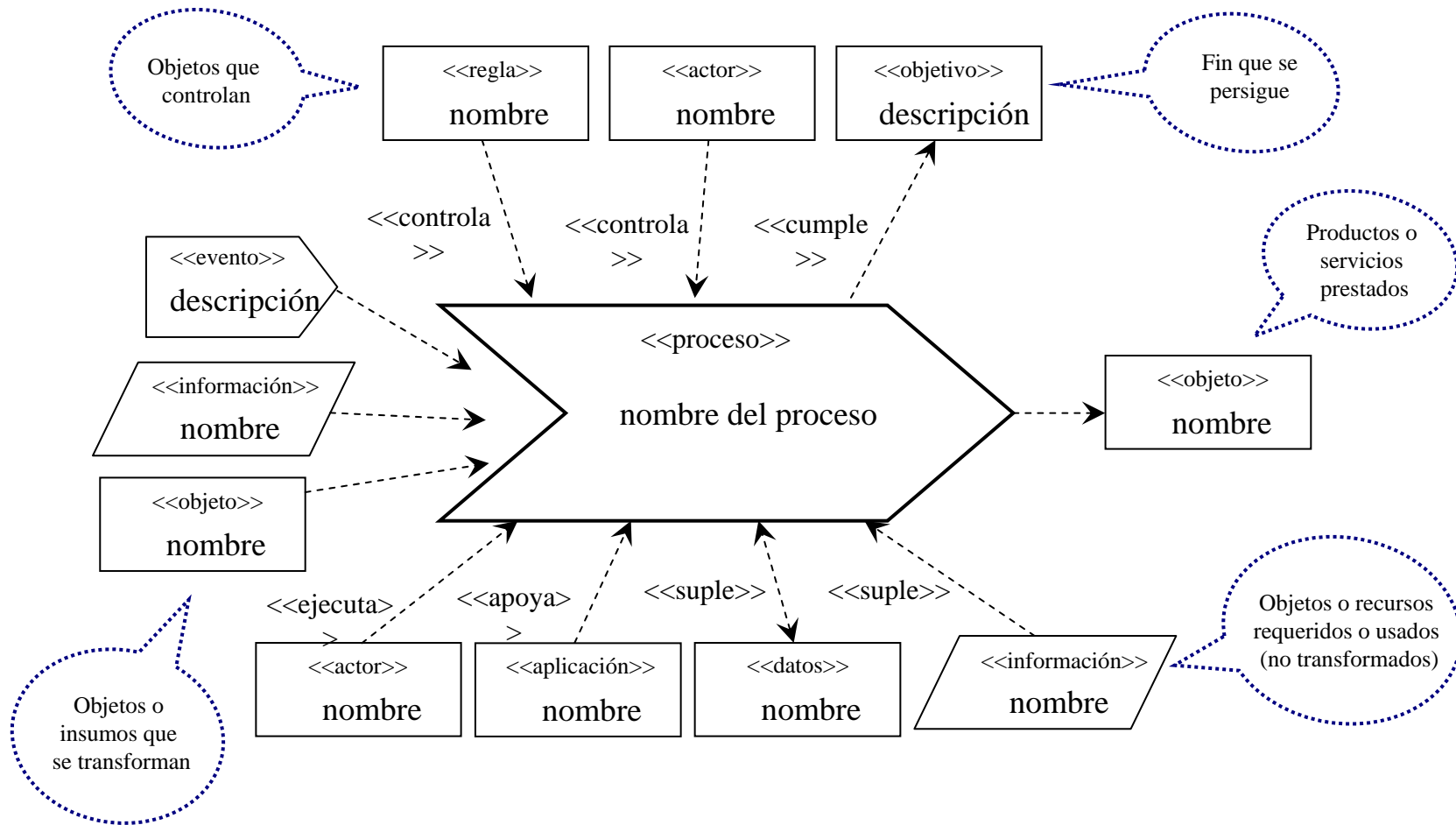
El **modelo de procesos** permite construir el producto (describe qué, cómo y cuándo deben ejecutarse las acciones para construir el producto o sistema).



Método de desarrollo de software

Notación UML Business
Eriksson y Penker (2000)

Diagrama de Procesos



Método de desarrollo de software

Fase: Diseño de software

Documento de diseño

- Arquitectura de software
 - Describe la estructura que tendrá la aplicación en términos de subsistemas, componentes y relaciones entre ellos.
- Modelos de datos
 - Describe el diseño de la (s) base (s) de datos que integrarán la aplicación (**modelo físico**).
- Diseños de interfaces
 - Describe las interfaces que debe tener la aplicación (**Interfaces usuario / sistema** e Interfaces entre aplicación y otros sistemas).
- Diseño de componentes
 - Describe los detalles del **diseño de cada uno de los componentes** (módulos) de software o programas que integran la arquitectura.

Enfoque: Estructural -> Orientado a Objetos

Documento de diseño de la aplicación

- 1.1. Objetivos de la aplicación
- 1.2. Requisitos de información
- 1.3. Requisitos de la aplicación
- 1.4. Restricciones (especificaciones)
- 1.5. Diseño de la aplicación
 - 1.5.1. Definición de usuarios
 - 1.5.2. Opciones del sistema (tabla de opciones del sistema: código de la opción, descripción, caso de uso asociado).
 - 1.5.3. Tabla de usuarios – opciones del sistema.
 - 1.5.4. Tabla de casos de uso (código, descripción).
 - 1.5.5. Descripción de los casos de uso y escenarios (solo en los casos diferentes a tablas básicas).
 - 1.5.6. Tabla de diagramas de estado (código, descripción).
 - 1.5.7. Diagramas de estado.
 - 1.5.8. Diseño físico de la base de datos (pdm - documentado).
 - 1.5.9. Tabla de interfaz de usuario (código, descripción).
 - 1.5.10. Interfaz de usuario.
 - 1.5.11. Tabla de reportes (código, descripción).
 - 1.5.12. Reportes.

Método de desarrollo de software

Fase: Ingeniería de Requisitos

Documento de requisitos

- *Modela el espacio de la solución*
- *Describe los requisitos que debe satisfacer el sistema de información*

Clasificación de los requisitos

- **Requisitos funcionales**
 - Los servicios que el sistema de información debe proveer a sus usuarios (se expresan, generalmente, mediante casos de uso)
- **Requisitos no funcionales**
 - Las restricciones que los diseñadores deben considerar para desarrollar la aplicación (plataformas de desarrollo y operación, estándares, recursos, etc.); Los atributos de calidad que la aplicación debe cumplir (confiabilidad, mantenibilidad, utilidad, etc.)

Modelos que describen la aplicación

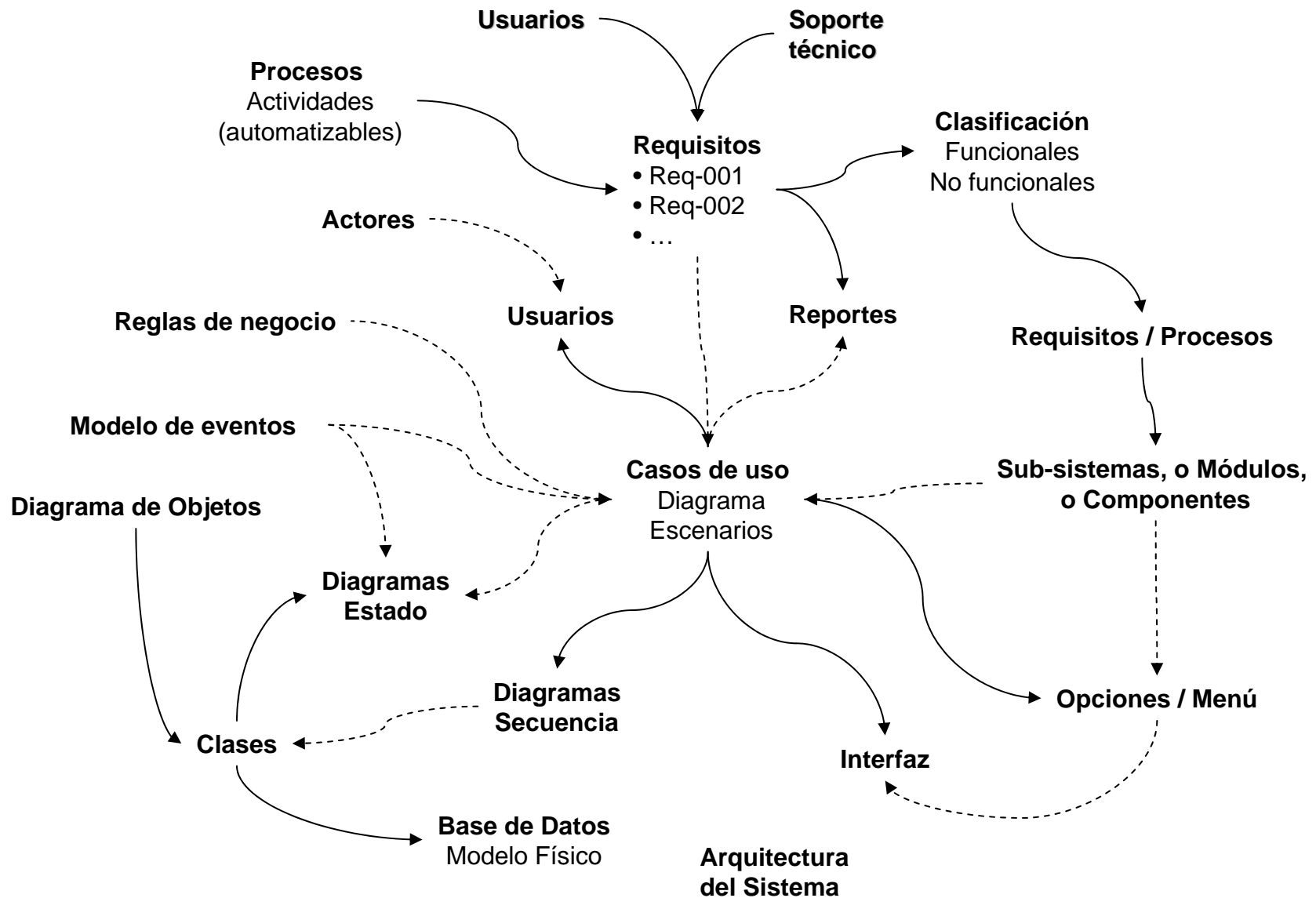
- **Modelo funcional**
 - Describe los requisitos funcionales de la aplicación (**diagramas de casos de uso, escenarios, usuarios, opciones del sistema**)
- **Modelo Estructural**
 - Describe los requisitos estructurales de la aplicación (**diagramas de clases y diagramas de componentes**)
- **Modelo Dinámico**
 - Describe el comportamiento de la aplicación (**diagramas de secuencia y diagramas de estado**)

¿A que nivel de detalle llegar en los modelos?



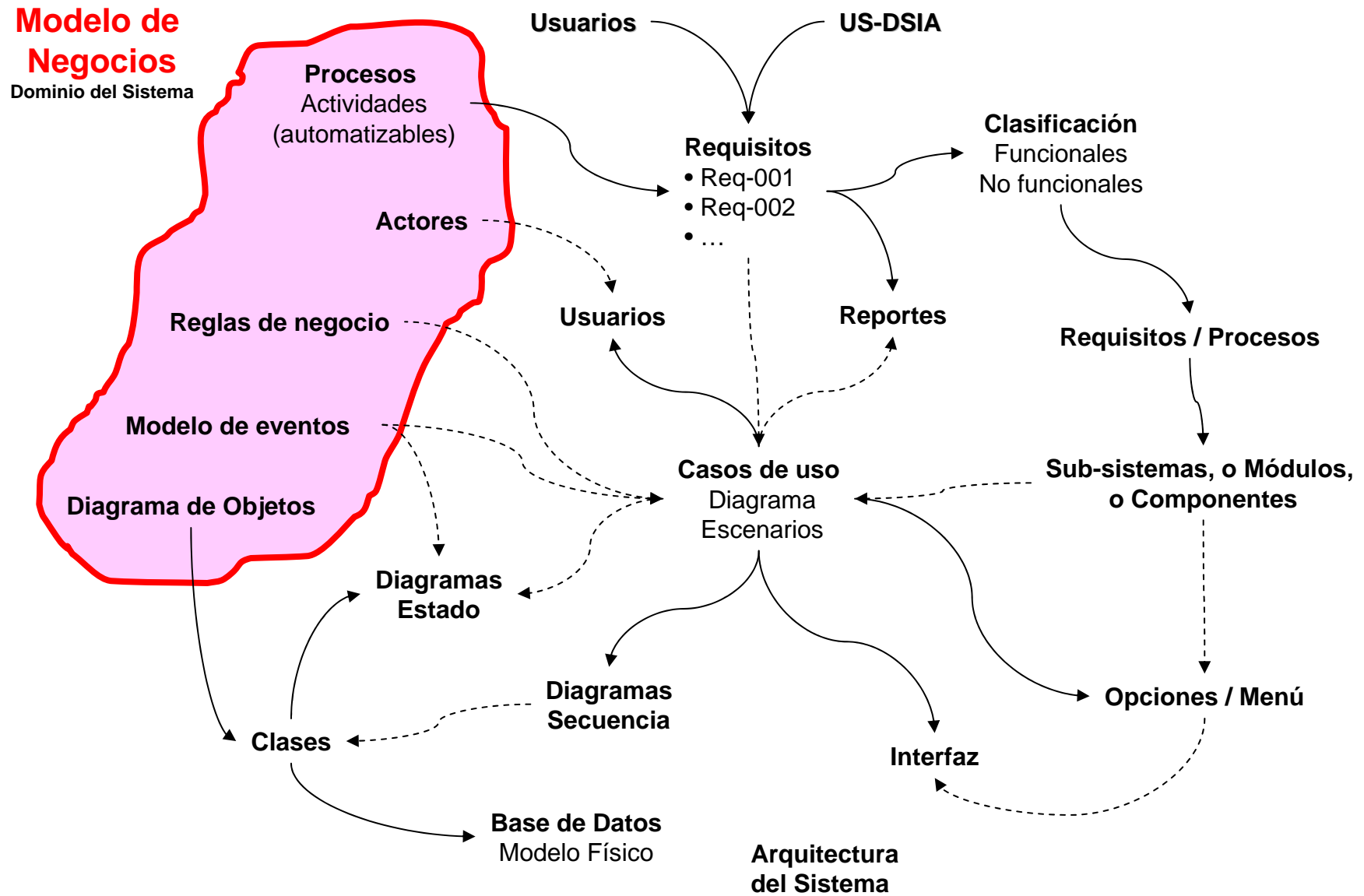
Método de desarrollo de software

Relación entre los productos que se generan en el proceso de desarrollo



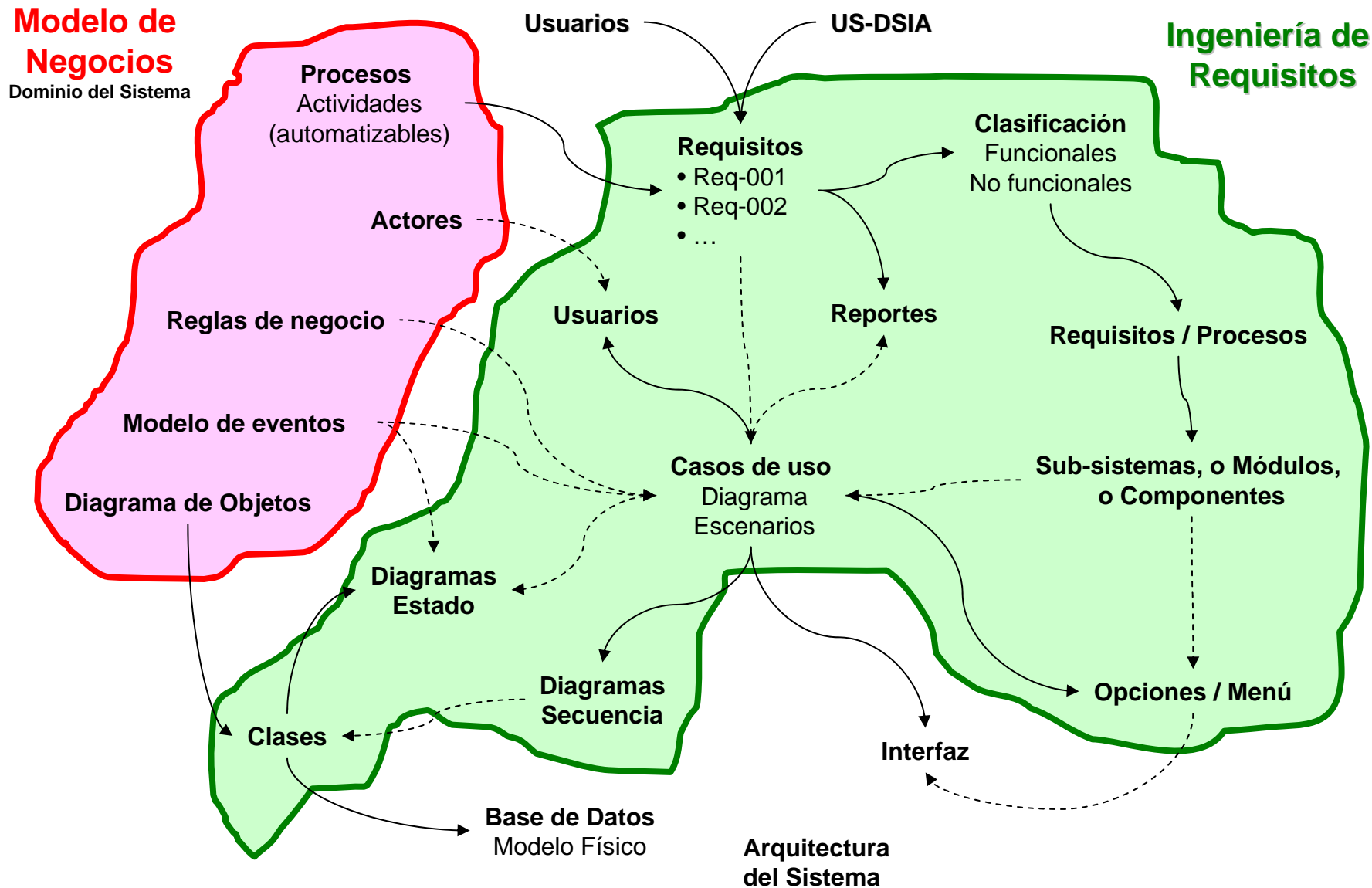
Método de desarrollo de software

Relación entre los productos que se generan en el proceso de desarrollo



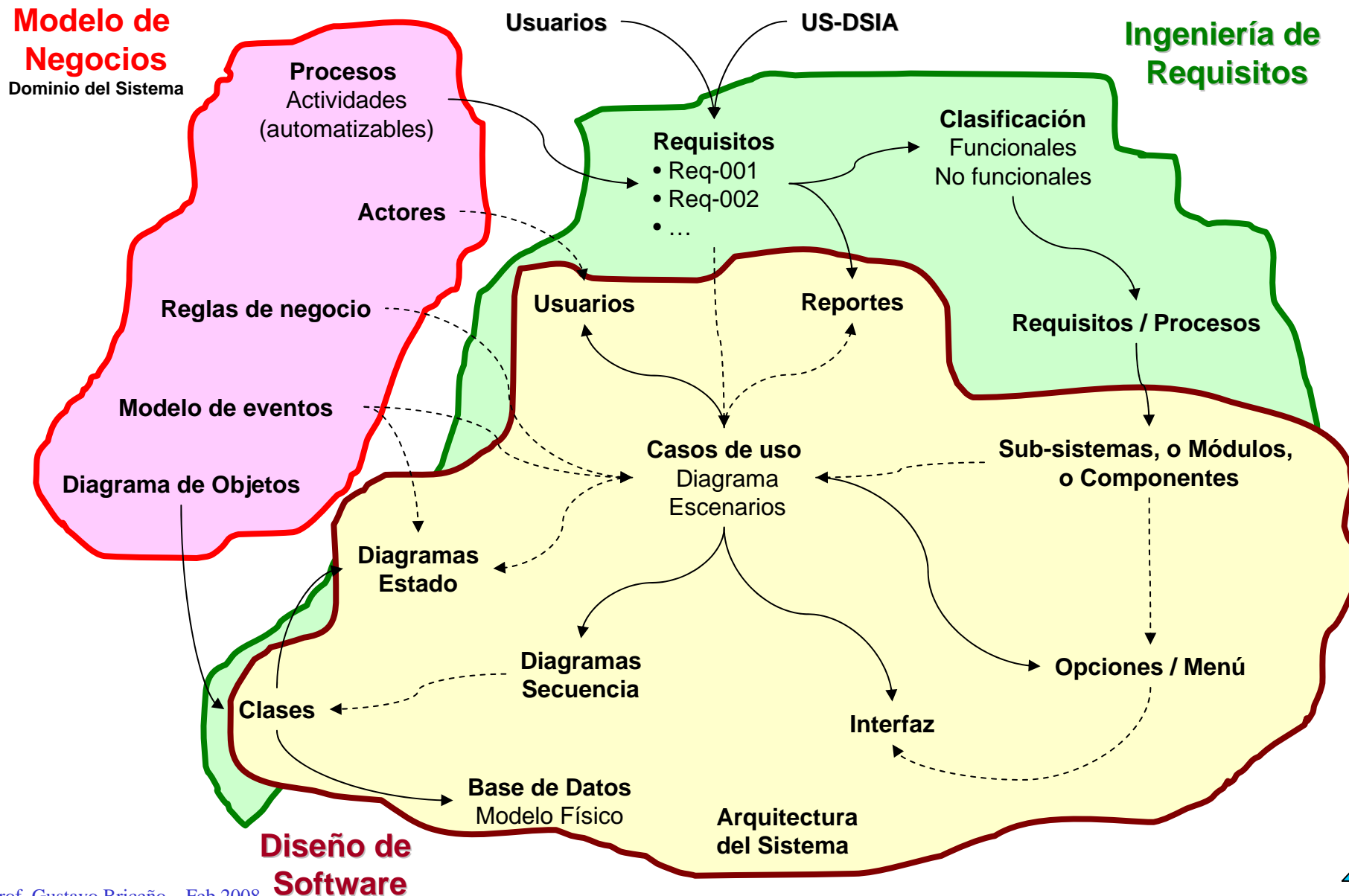
Método de desarrollo de software

Relación entre los productos que se generan en el proceso de desarrollo



Método de desarrollo de software

Relación entre los productos que se generan en el proceso de desarrollo



¿Qué es un requisito?

“es una propiedad que debe exhibir [cumplir o satisfacer] un sistema desarrollado o adaptado para resolver un problema particular” [Sawyer y Kotonya, 2001]

“es un aspecto de un sistema o una descripción de aquello que el sistema es capaz de hacer a fin de cumplir su propósito” [Pfleeger, 1998]

“Los requisitos expresan qué se supone debe hacer una aplicación [sistema] ... no intenta expresar cómo lograr estas funciones” [Braude, 2003]

Los requisitos expresan lo que el sistema debe hacer para satisfacer las necesidades de sus clientes o usuarios.

Los requisitos definen:

- Lo que la aplicación debe hacer
 - Las funciones que debe ejecutar
 - Los datos que debe capturar y almacenar
 - La información que debe producir
- La interacción entre los usuarios y el sistema
 - La interfaz gráfica usuario-sistema (GUI)
- Las restricciones bajo las cuales se debe operar
 - La plataforma de operación del sistema (hardware y software)
 - La tecnología de información que debe utilizar
 - Las interfaces con otros sistemas
- Los atributos de calidad que el sistema debe satisfacer
 - Seguridad, facilidad de uso, documentación, utilidad, etc.

2. Ingeniería de requisitos

Objetivos: Determinar las necesidades de información y automatización de los procesos de negocios, que tienen los usuarios de la aplicación empresarial en desarrollo, mediante la definición y especificación de sus requisitos.

Producto: Documento de requisitos del sistema

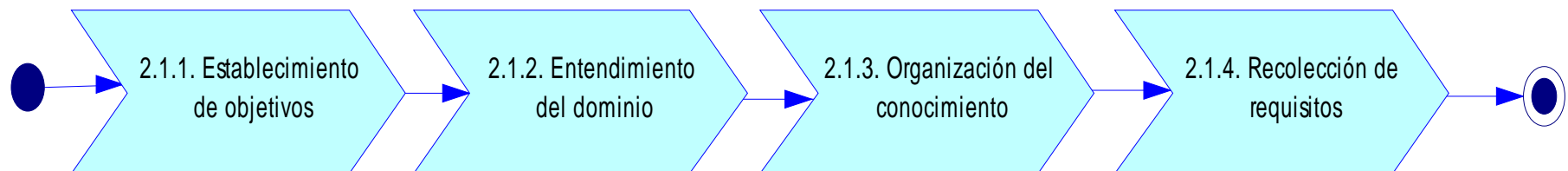
Actividades del proceso de ingeniería de requisitos

- 2.1. [Descubrimiento de requisitos](#)
- 2.2. [Análisis de requisitos](#)
- 2.3. [Especificación de requisitos](#)
- 2.4. [Validación de requisitos](#)

2.1. Descubrimiento de requisitos

Objetivo: Capturar las necesidades que los usuarios y otros interesados tienen en relación al sistema de software que se desea desarrollar.

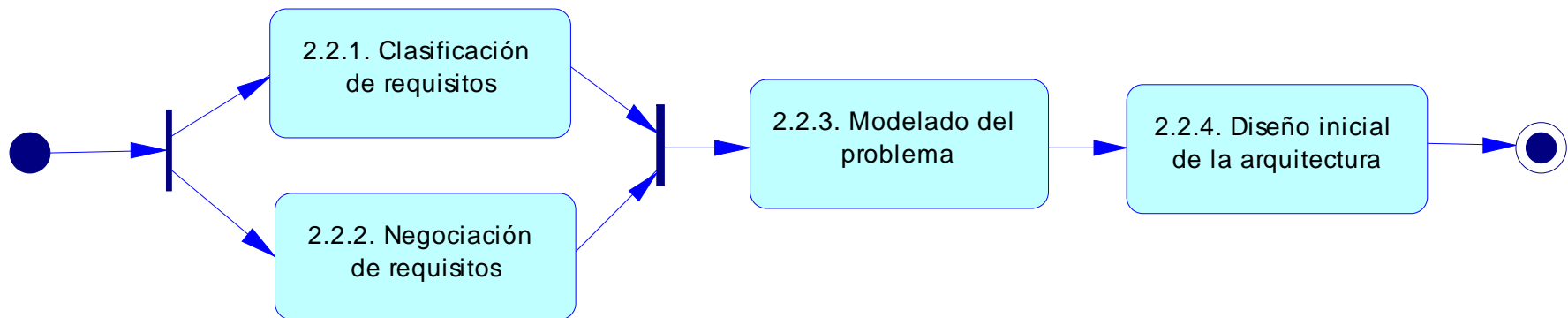
Producto: Inventario de necesidades de los usuarios relacionadas con el sistema de software



2.2. Análisis de requisitos

Objetivo: Analizar las necesidades identificadas de los usuarios, para llegar a una definición de los requisitos del sistema de software que se desea desarrollar.

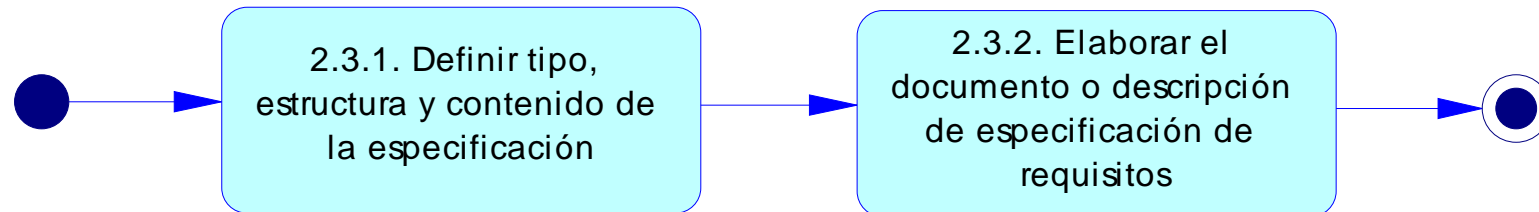
Producto: Inventario de requisitos definidos



2.3. Especificación de requisitos

Objetivo: Documentar (descripción detalla) los requisitos definidos por los usuarios.

Producto: Documento de requisitos del sistema



2.4. Validación de requisitos

Objetivo: Evaluar y validar el documento de requisitos para asegurar que defina el sistema de software correctamente.

Producto: Documento de requisitos del sistema validado

