

# Introducción a la ingeniería de software – Clase 3





## En la clase anterior ...

### Flujo de proceso

Organización de las actividades y sus tareas con respecto a secuencia y tiempo

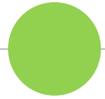
### Ciclo de vida

Proceso de desarrollo de SW  
Diferentes modelos de proceso

### Metodologías ágiles

Modelo de proceso centrado en el cambio





## Temas de hoy



### Ingeniería de requerimientos

Comprensión

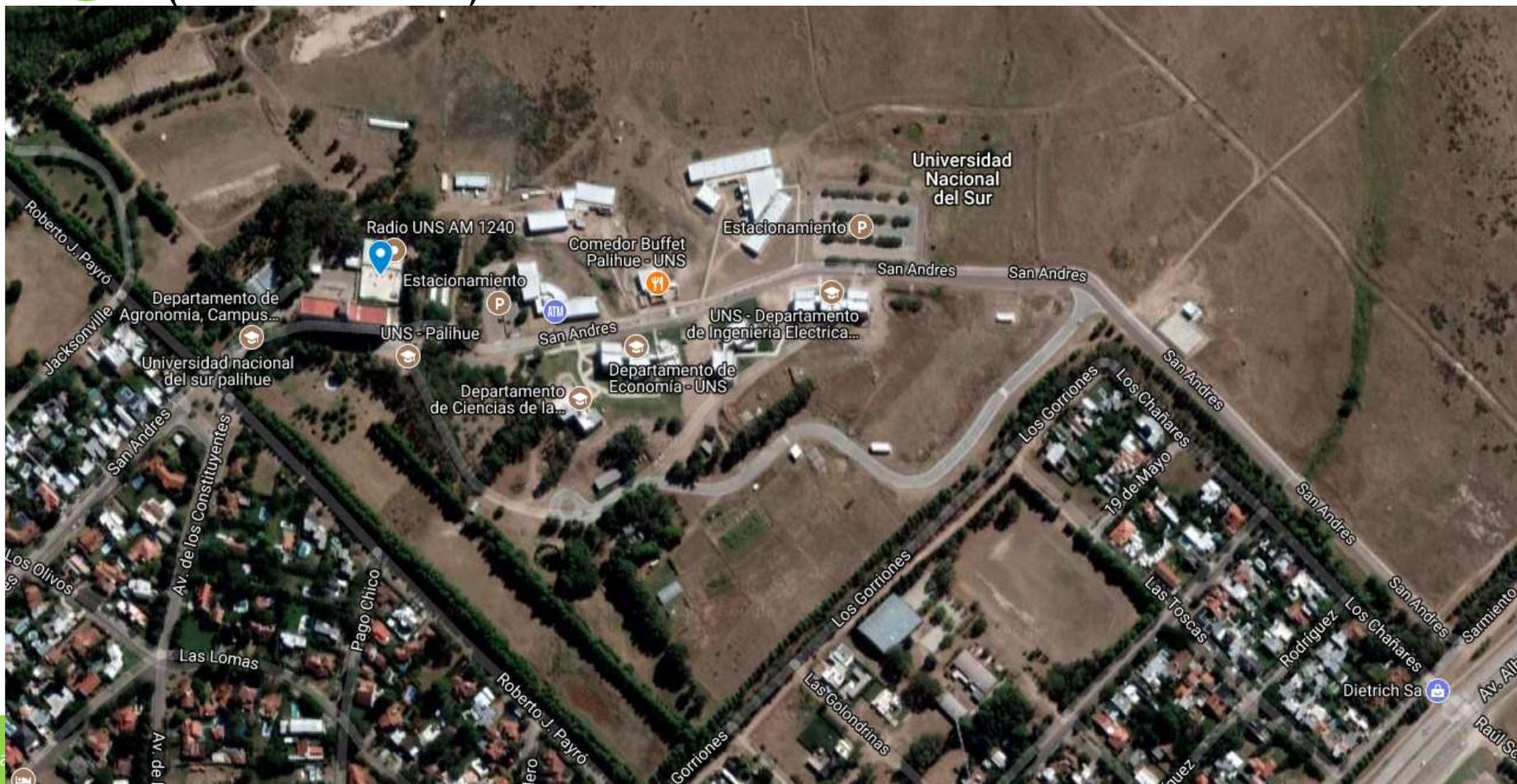
Modelado

Expresión

Herramientas

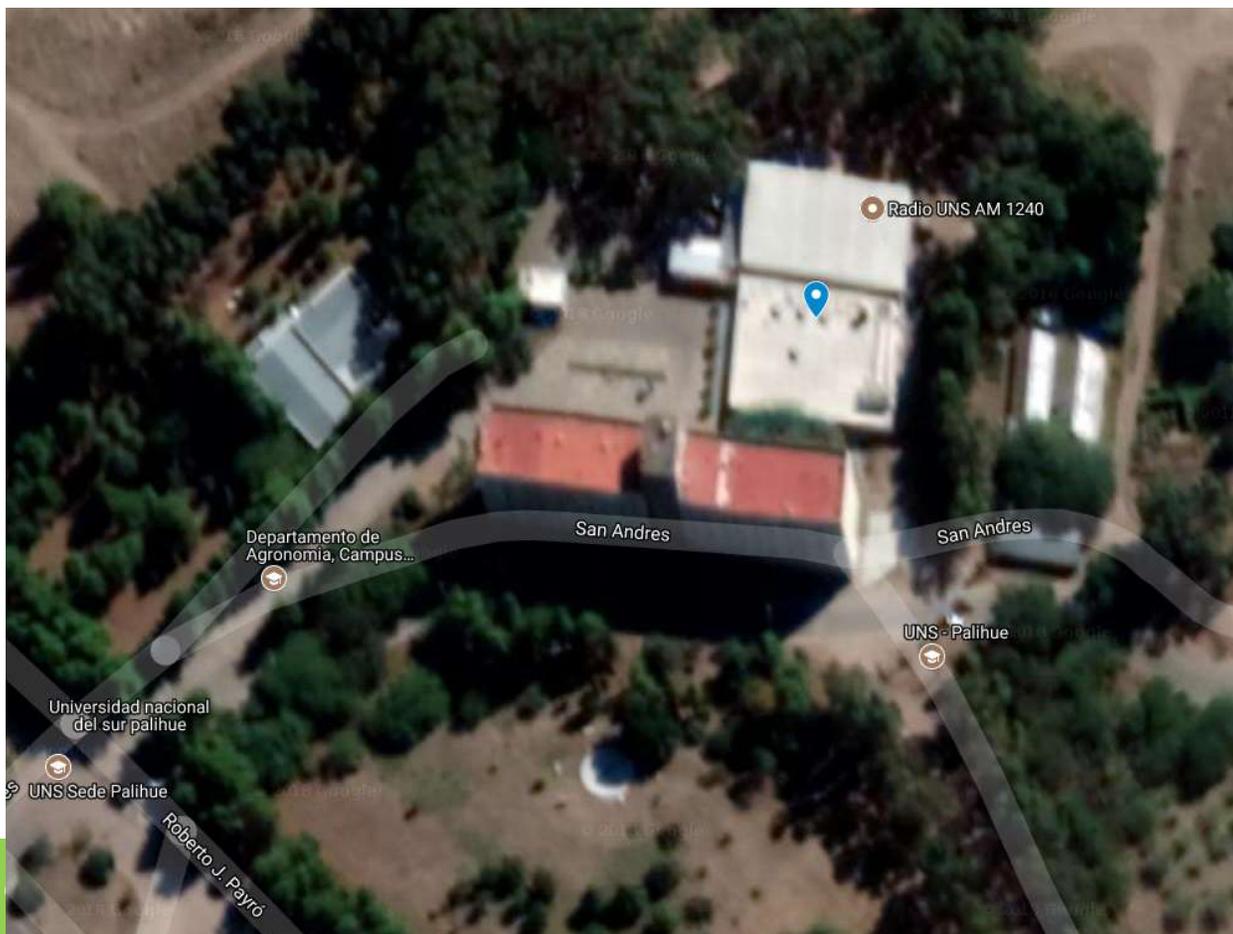


## Examen Sabádo Aula 2 Agronomía (comienzo 9.00hs)





## Examen Sabádo Aula 2 Agronomía (comienzo 9.00hs)



1

# Ingeniería de requerimientos





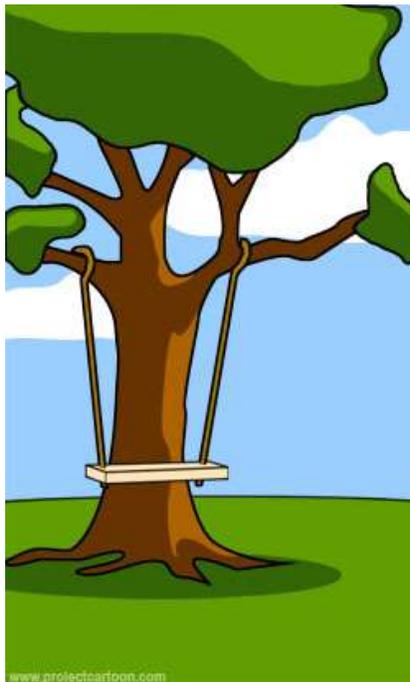
## La hamaca en el árbol



Cómo lo  
describió el  
cliente



## La hamaca en el árbol



Lo que  
entendió el  
líder del  
proyecto

## La hamaca en el árbol



Cómo lo  
diseño el  
analista



## La hamaca en el árbol



Lo que  
codificó el  
programador



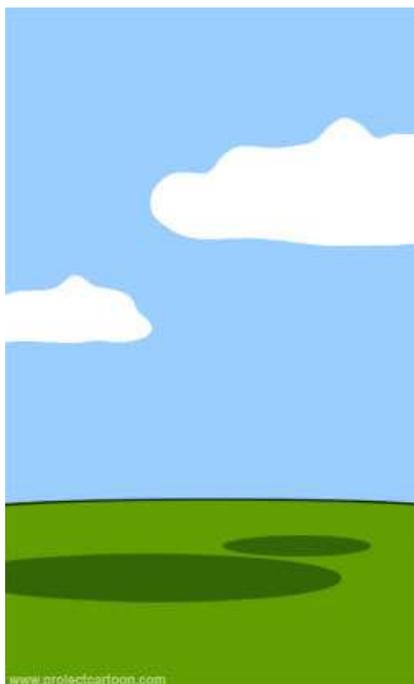
## La hamaca en el árbol



Cómo lo describió el agente de negocios



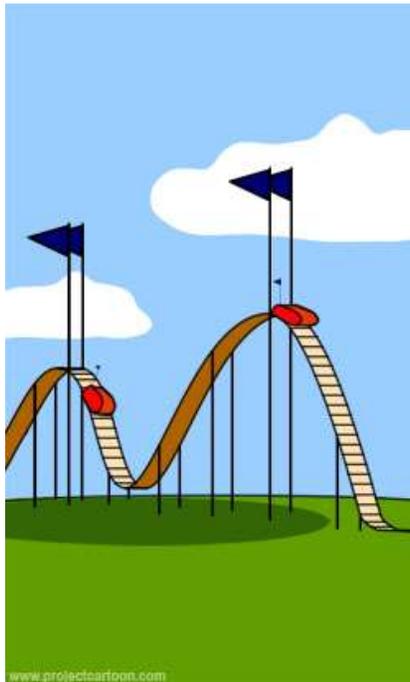
## La hamaca en el árbol



Cómo se  
documentó  
el proyecto



## La hamaca en el árbol



Cómo se le  
cobró al  
cliente

**La hamaca en el árbol**



Lo que el cliente  
necesitaba ...

<http://www.projectcartoon.com/>



## Requerimiento

Característica del sistema o descripción de algo que el sistema es capaz de hacer con el objeto de satisfacer su propósito





## Ingeniería de requerimientos

- Es una de las acciones más importantes de la ingeniería de software.
- Proporciona el mecanismo para:
  - Entender lo que desea el cliente
  - Analizar las necesidades
  - Evaluar la factibilidad
  - Negociar una solución razonable
  - Especificar la solución sin ambigüedades
  - Validar la especificación
  - Administrar los requerimientos mientras se transforman





## Ingeniería de requerimientos

### ● Incluye:

1. Concepción
2. Indagación
3. Elaboración
4. Negociación
5. Especificación
6. Validación
7. Administración

### ● Algunas ocurren en paralelo



## Ing. RQs – 1. Concepción

- ¿Cómo se inicia un proyecto de software?
  - Conversación casual
  - Necesidad de negocio
  - Nuevo mercado
  - Nuevo servicio
  
- Se establece el entendimiento básico del problema.



## Ing. RQs – 1. Concepción

- Identificar a los participantes
- Reconocer los diferentes puntos de vista
- Buscar la colaboración: identificar áreas de interés común y las de conflictos
- Comenzar con las preguntas: centrar en el cliente y participantes, así como en metas y beneficios generales
  - ¿Quién ...? ¿Cuál ...? ¿Opciones ...?





## Ing. RQs – 2. Indagación

- ¿Simplemente preguntar?
- Problemas que ocurren
  - **De alcance:** frontera mal definida, especificación de detalles que confunden o son innecesarios
  - **De entendimiento:** el cliente no está seguro de lo que necesita, dificultades para transmitirlo al ingeniero, omiten información por considerarla obvia, requerimientos en conflicto o ambiguos o que no se pueden probar
  - **De volatilidad:** requerimientos cambian con el tiempo





## Ing. RQs – 2. Indagación

### ○ Distinguir requerimientos

- Que deben ser absolutamente satisfechos. A veces llamados atributos o requerimientos críticos.
- Que son deseables pero no indispensables
- Que son posibles pero podrían eliminarse

### ○ El QUÉ. No el CÓMO. No confundir requerimientos con soluciones.





## Ing. RQs – 2. Indagación

### ○ Requerimientos funcionales

- ¿Qué hará el sistema?
- Describen interacciones entre el sistema y su entorno

### ○ Requerimientos no funcionales (o restricciones)

- Describen restricciones sobre el sistema que limitarán las elecciones en la construcción de la solución al problema





## Ing. RQs – 2. Indagación

- Es necesario entender cómo van a utilizar los usuarios finales las funciones y características
- Escenarios y Casos de uso
  - Captan contratos que describen el comportamiento del sistema en distintas condiciones en las que responde a peticiones de los participantes
  - Narra una historia sobre la interacción con el usuario
  - Se definen los actores de la historia





## Ing. RQs – 2. Indagación

- Preguntas a responder por un caso de uso
  - ¿Quién es el actor principal y quienes los secundarios?
  - ¿Cuáles son los objetivos de los actores?
  - ¿Qué precondiciones deben existir al empezar la historia?
  - ¿Qué excepciones deben considerarse al describir la historia?
  - ¿Qué variaciones son posibles en la interacción?
  - ¿Qué información adquiere, produce o cambia el actor?
  - ...





## Ing. RQs – 3. Elaboración

- ⦿ Expandir y refinar la información obtenida
- ⦿ Desarrollar un modelo de los requerimientos que identifique: funciones del software, comportamiento, información
- ⦿ Elementos: basados en el escenario, basados en las clases, basados en el comportamiento (diagrama de estados), orientados al flujo





## Ing. RQs – 4. Negociación

- ⦿ Es necesario un proceso de negociación cuando
  - El cliente pide más de lo que se puede conseguir
  - Hay requerimientos conflictivos
- ⦿ Enfoque iterativo que dé prioridad a los requerimientos, evalúe costos y riesgos
- ⦿ Se eliminan, combinan, o modifican requerimientos hasta lograr un cierto grado de satisfacción





## Ing. RQs – 5. Especificación

- Documento escrito
- Conjunto de modelos gráficos
- Modelo matemático formal
- Conjunto de escenarios de uso
- Prototipo
- O combinaciones de los anteriores



## Ing. RQs – 5. Especificación

- Plantillas de requerimientos

Requirement #: **75**      Requirement Type: **9**      Event/BUC/PUC #: **7, 9**

Description: **The product shall record all the roads that have been treated**

Rationale: **To be able to schedule untreated roads and highlight potential danger**

Originator: **Arnold Snow - Chief Engineer**

Fit Criterion: **The recorded treated roads shall agree with the drivers' road treatment logs and shall be up to date within 30 minutes of the completion of the road's treatment**

Customer Satisfaction: **3**      Customer Dissatisfaction: **5**

Dependencies: **All requirements using road and scheduling data**      Conflicts: **105**

Supporting Materials: **Work context diagram, terms definitions in section 5**

History: **Created February 29, 2010**

**Volere**  
Copyright © Atlantic Systems Guild



## Ing. RQs – 5. Especificación

### ○ Plantillas de requerimientos

#### Template para Requerimientos Funcionales

nombre	nombre del requerimiento
req. id	etiqueta de identificación del requerimiento
categoría	tipo de requerimiento (en este caso: funcional)
descripción	breve descripción del comportamiento requerido del sistema
términos	terminología utilizada en la descripción y cuya definición se incluye en el glosario
justificación	razón por la que es necesario
prioridad	alta, media o baja
dependencias	otros requerimientos de los cuales este depende
documentos	documentos con información relevante para este requerimiento
argumentos de factibilidad	por qué / de qué manera se garantiza que puede cumplirse
método de verificación	cómo se verificará que se cumpla



## Ing. RQs – 5. Especificación

### ○ Plantillas de requerimientos

#### Template para Requerimientos No Funcionales

nombre	nombre del requerimiento
req. id	etiqueta de identificación del requerimiento
categoría	tipo de requerimiento
descripción	breve descripción del requerimiento (calidad, restricción, etc.)
escala	unidades de medida definidas para la medición
test	forma en la que se medirá el atributo
peor caso	peor nivel aceptable bajo cualquier circunstancia
nivel planificado	nivel esperado de un atributo
mejor caso	mejor nivel alcanzado, es una referencia no un requerimiento
nivel actual	valores actuales del atributo (si existieran) con los cuales se comparará



## Ing. RQs – 6. Validación

- Se analiza la especificación para garantizar que:
  - todos los requerimientos han sido enunciados
  - no hay ambigüedades
  - se detectaron y corrigieron inconsistencias, omisiones y errores
  - se siguieron los estándares establecidos
- La revisión técnica involucra ingenieros de software, clientes, usuarios, expertos del dominio, y todo otro participante.





## Ing. RQs – Validación

- Algunas preguntas para realizar una validación
  - ¿Están enunciados con claridad? ¿Podrían interpretarse mal?
  - ¿Está identificada la fuente del requerimiento?
  - ¿Qué requerimientos se relacionan con éste?
  - ¿Puede someterse a pruebas?
  - ¿Puede rastrearse en todos los modelos del sistema?
  - ¿Se ha creado un índice para la especificación?
  - ...





## Ing. RQs – 7. Administración

- ⦿ Se debe identificar, controlar y dar seguimiento a los requerimientos y a los cambios que sufran.
- ⦿ Es importante determinar cómo se van a administrar los cambios.



2

## Extracción, modelado y herramientas





## Requerimientos: características

- ⦿ **Correctos:** sin errores
- ⦿ **Consistentes:** sin conflictos ni ambigüedades
- ⦿ **Completos:** externa e internamente completos
- ⦿ **Realistas:** ¿puede hacerse realmente lo solicitado?
- ⦿ **Necesarios:** sin restricciones innecesarias
- ⦿ **Verificables:** ¿se puede probar que se cumplen?
- ⦿ **Rastreables:** ¿se pueden identificar fácilmente?





## Modelado de requerimientos

- Basados en el escenario: requerimientos desde el punto de vista de “actores” del sistema
- De datos: ilustran el dominio de información del problema
- Orientados a clases: representan objetos y la manera en la que colaboran para cumplir requerimientos del sistema
- Orientados al flujo: transformación de los datos a medida que avanzan en el sistema
- De comportamiento: modo en el que se comporta el software como consecuencia de eventos





## Expresión de requerimientos

### Descripciones estáticas

- Relaciones de recurrencia
- Condición inicial y reglas de transformación de condiciones
  - Definición axiomática
- Propiedades básicas del sistema y comportamiento
  - Expresión como un lenguaje
- Expresiones de la sintáxis de las cadenas
  - Abstracciones de datos
- Definir clases, objetos, tipos de datos





## Expresión de requerimientos

- Descripciones dinámicas

  - Tablas de decision

  - Condiciones posibles, reglas para reaccionar ante estímulos y acciones a ser tomadas como resultado





## Tablas de decisión

	R1	R2	R3	R4	R5
Notas altas de examen	V	F	F	F	F
Buen promedio	-	V	F	F	F
Actividades externas	-	-	V	F	F
Buenas recomendaciones	-	-	-	V	F
Enviar carta de rechazo			X	X	X
Enviar formularios de admisión	X	X			

Condiciones y acciones sobre el lado izquierdo  
Cada columna es un conjunto de condiciones  
(estado del sistema)



## Tablas de decisión

	R1	R2	R3
Notas altas de examen	V	F	F
Buen promedio	-	V	F
Actividades externas	-	-	V
Buenas recomendaciones	-	-	-
Enviar carta de rechazo			X
Enviar formularios de admisión	X	X	

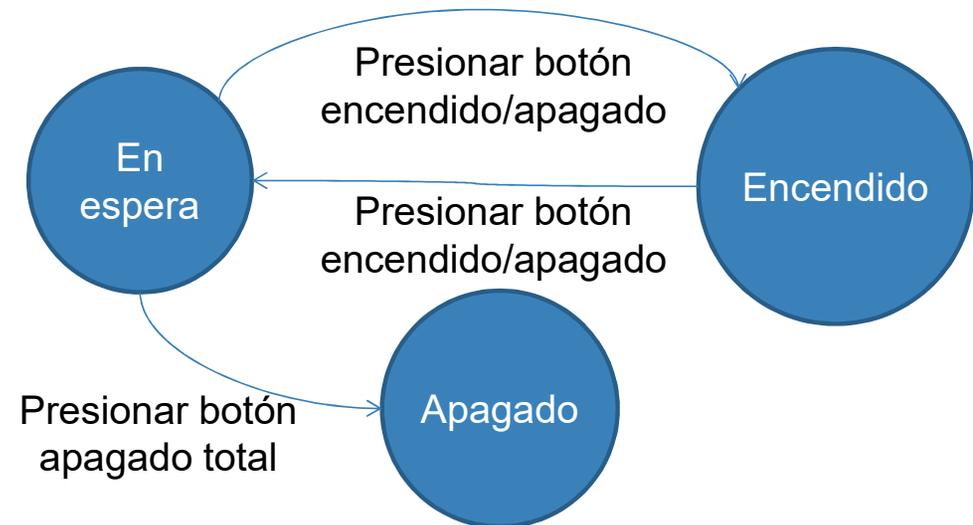
Permite determinar si la especificación está completa  
Ayuda a detectar inconsistencias o conflictos



## Expresión de requerimientos

### ○ Descripciones dinámicas

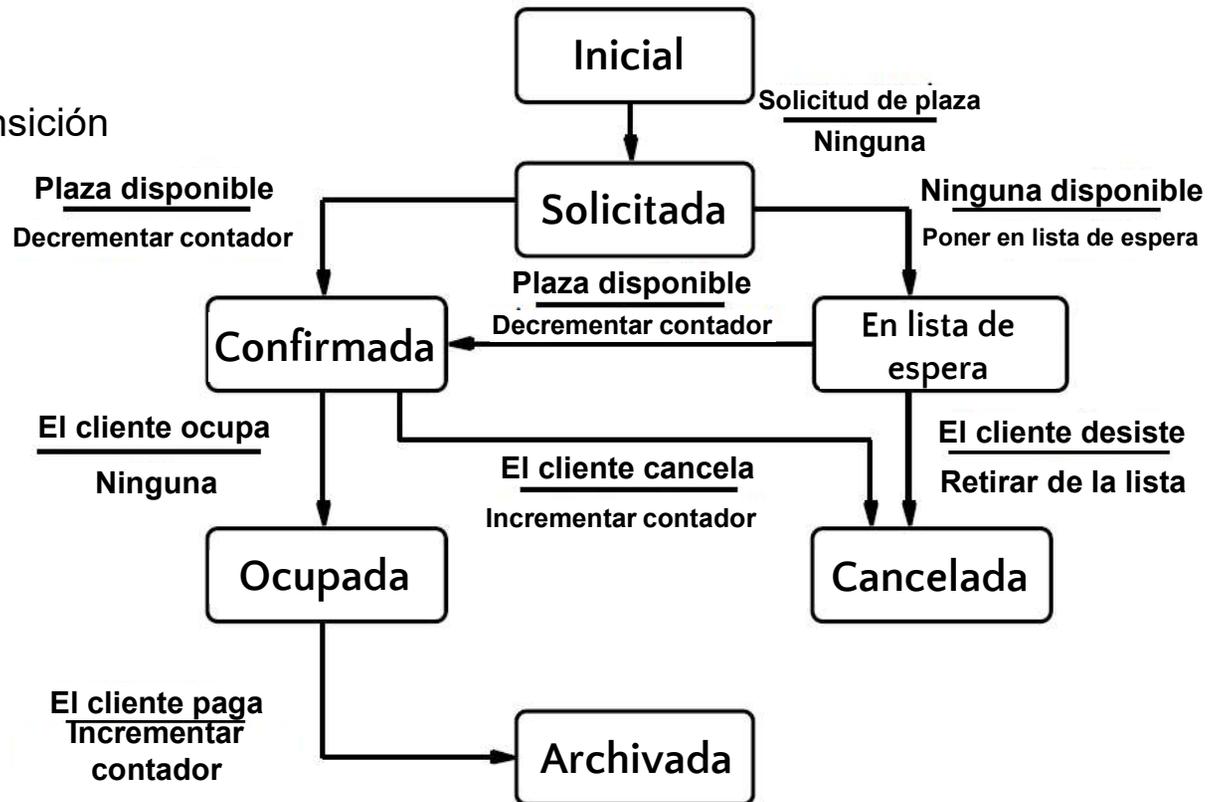
- Tablas de decisión
- **Descripciones funcionales y diagramas de transición**
- Conjunto de estados y reacciones del sistema ante eventos. Sistema pensado como serie de funciones.





## Expresión de requerimientos

Diagrama de transición





## Expresión de requerimientos

### ○ Descripciones dinámicas

- Tablas de decisión

- Descripciones funcionales y diagramas de transición

- **Diagramas de flujos de datos**

- Permiten modelar el flujo de los datos hacia, en y desde el sistema.

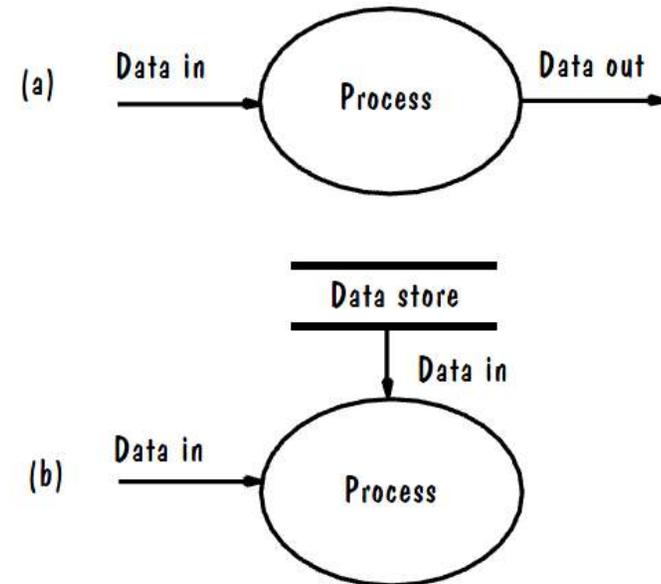




## Expresión de requerimientos

### Diagramas de flujo de datos

- Datos de entrada
- Procesos
- Datos de salida
- Almacenamientos
  - Bases de datos
  - Archivos

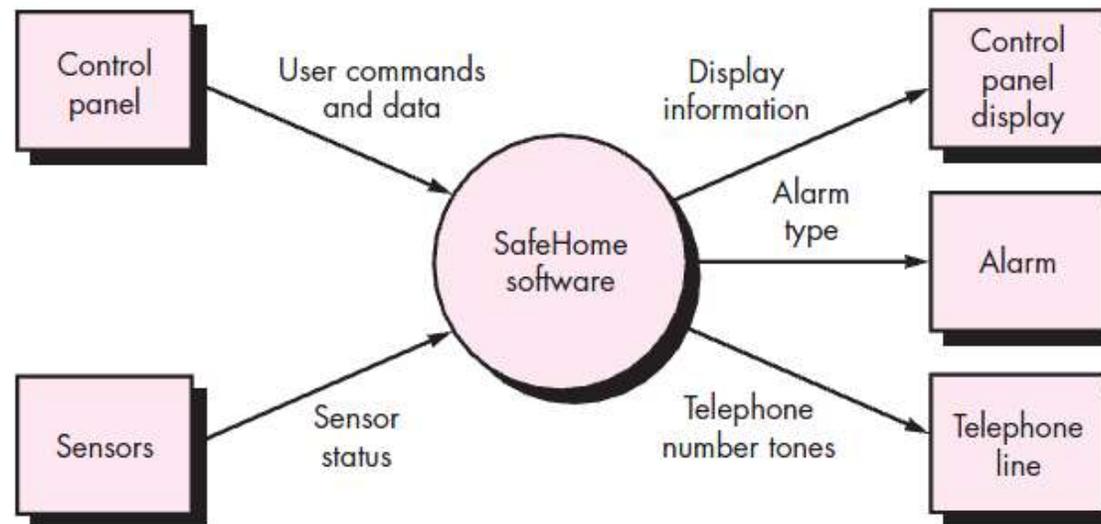


## Expresión de requerimientos

### Diagramas de flujo de datos

#### ○ Construcción jerárquica

- DFD de Nivel 0 o diagrama de contexto: única burbuja

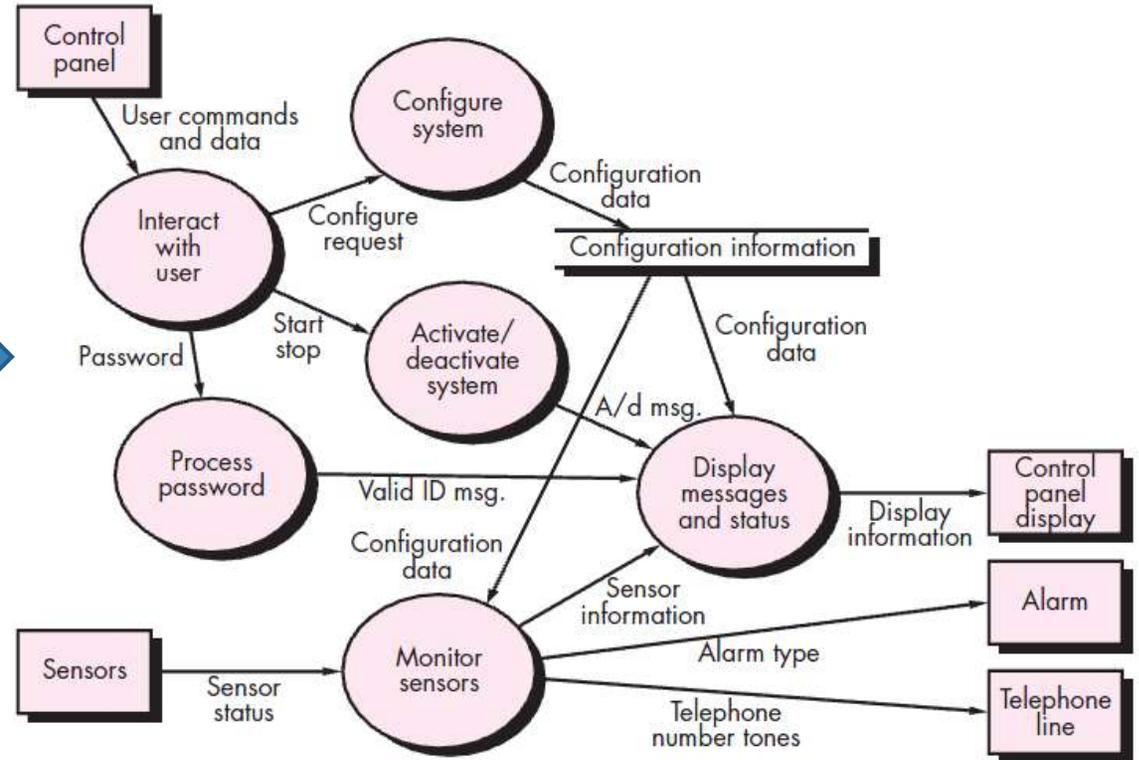
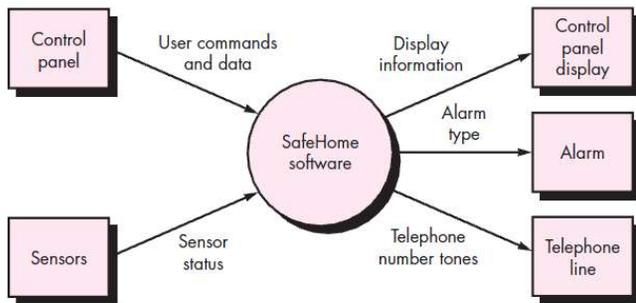




# Expresión de requerimientos

## Diagramas de flujo de datos

### DFD de nivel 1

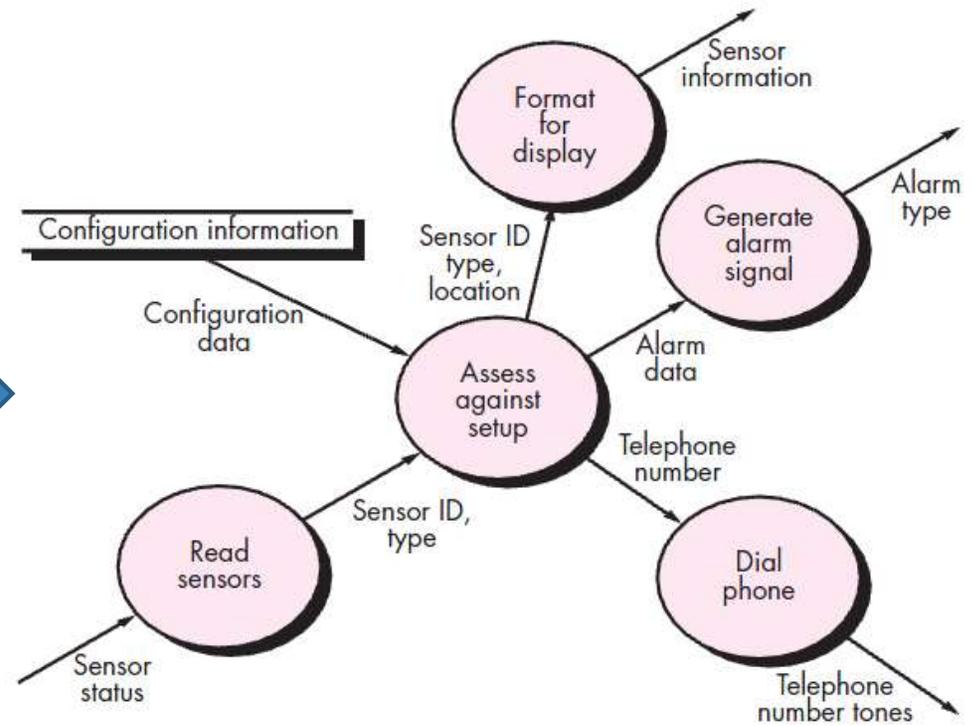
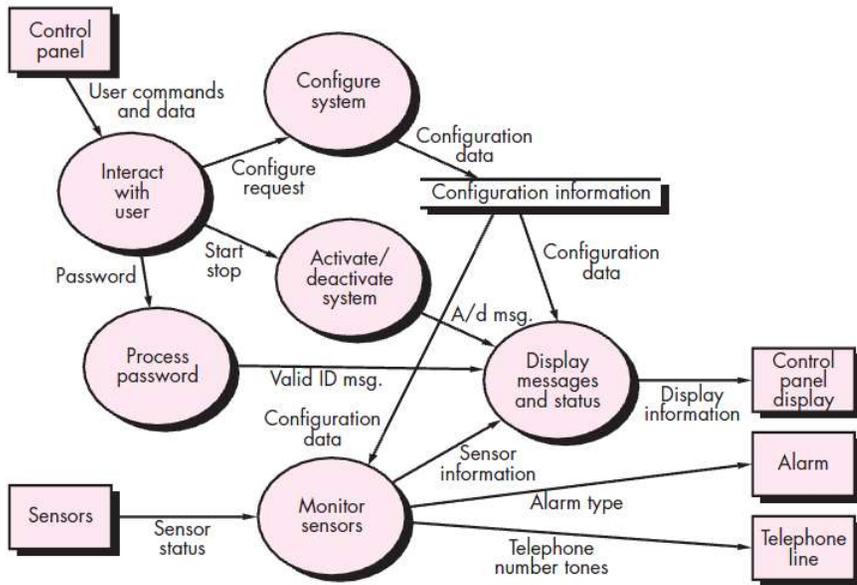




## Expresión de requerimientos

### Diagramas de flujo de datos

DFD de nivel 2 – Detalle de “Monitorear sensores”





## Expresión de requerimientos

### ○ Descripciones dinámicas

- Tablas de decisión
- Descripciones funcionales y diagramas de transición
- Tabla de eventos
- Diagramas de flujos de datos
- **Especificación orientada a objetos**
- Enfoque sobre las entidades involucradas en el sistema





## Especificación orientada a objetos

- Objeto: entidad del sistema
- Método u operación: acción que puede ser ejecutada por el objeto o que le puede pasar a él
- Conceptos característicos de la OO
  - Encapsulación
  - Jerarquía de clases
  - Herencia
  - Polimorfismo



## Expresión de requerimientos

### Modelos UML

- Casos de uso
- Diagrama de actividad
- Diagrama de canal
- Diagramas de secuencia
- Diagramas de clases
- ...





## Casos de Uso - Narración informal

⦿ Caso de Uso: Acceder a la vigilancia con cámaras a través de internet, mostrar vistas de cámaras. (AVC-MVC)

Actor: propietario

Si estoy en una localidad alejada, puedo usar cualquier PC con un software de navegación apropiado para entrar al sitio web de *Productos Casa Segura*. Introduzco mi identificación de usuario y dos niveles de claves; una vez validadas, tengo acceso a toda la funcionalidad de mi sistema instalado. Para acceder a la vista de una cámara específica, selecciono “vigilancia” de los botones mostrados para las funciones principales. Luego selecciono “escoger una cámara” y aparece el plano de la casa. Después elijo la [...]





## Casos de Uso - Secuencia de acciones

### ● Caso de Uso: (AVC-MVC)

Actor: propietario

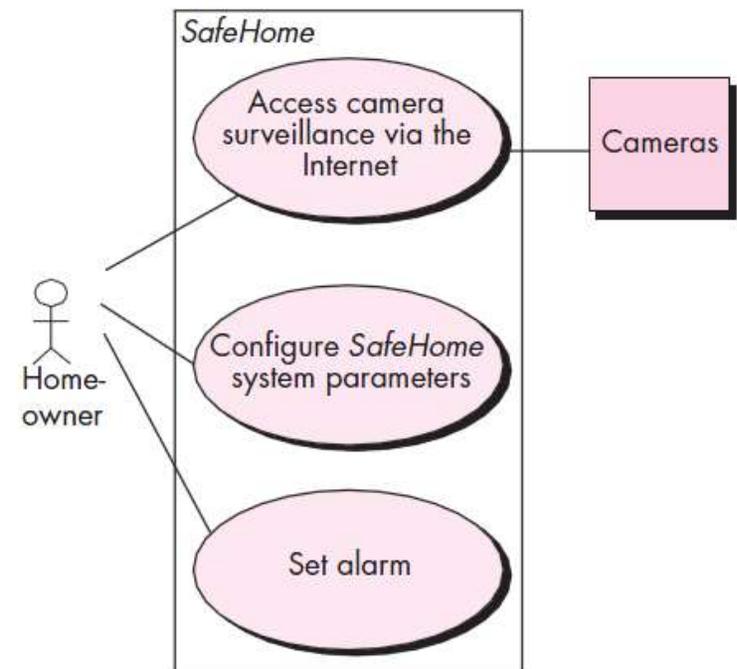
1. El propietario accede al sitio web *Productos Casa Segura*
2. El propietario introduce su identificación de usuario
3. El propietario escribe dos claves (cada una de al menos ocho caracteres de longitud)
4. El sistema muestra los botones de todas las funciones principales
5. El propietario selecciona “vigilancia” de los botones de las funciones principales
6. El propietario elige “seleccionar una cámara”
7. ...





## Casos de Uso - Diagrama gráfico

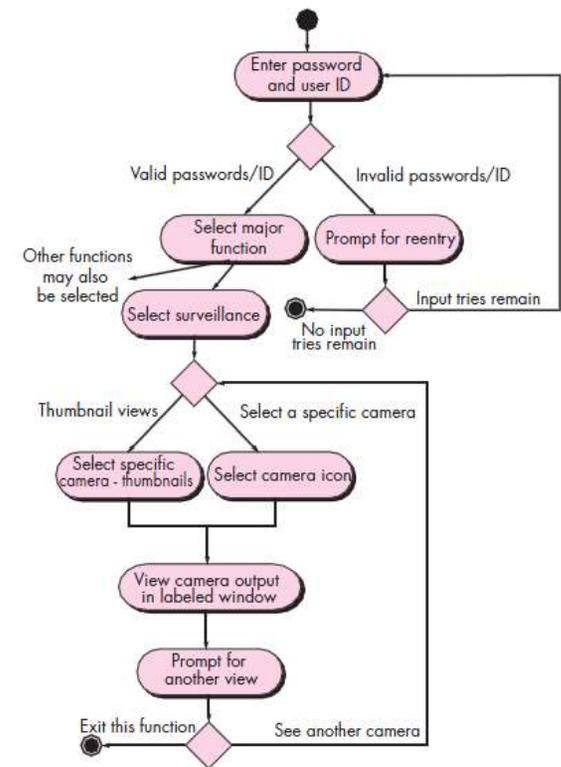
○ Caso de Uso: (AVC-MVC)  
Diagrama UML





## Diagrama de actividad

- Representación gráfica del flujo de interacción dentro de un escenario específico
- Rectángulo redondeado: función del sistema
- Flecha: flujo entre funciones
- Rombo: decisión





## Expresión de requerimientos

- Modelado basado en clases
  - Objetos y Clases
  - Operaciones (métodos o servicios)
  - Relaciones entre los objetos
  - Colaboraciones entre clases



## Modelado basado en clases

- Identificar clases, objetos y atributos
  - Guiarse con: sustantivos y sinónimos
- Entidades externas que producen o consumen la información del sistema
- Cosas que forman parte del dominio de información
- Ocurrencias o eventos que suceden dentro del contexto





## Modelado basado en clases

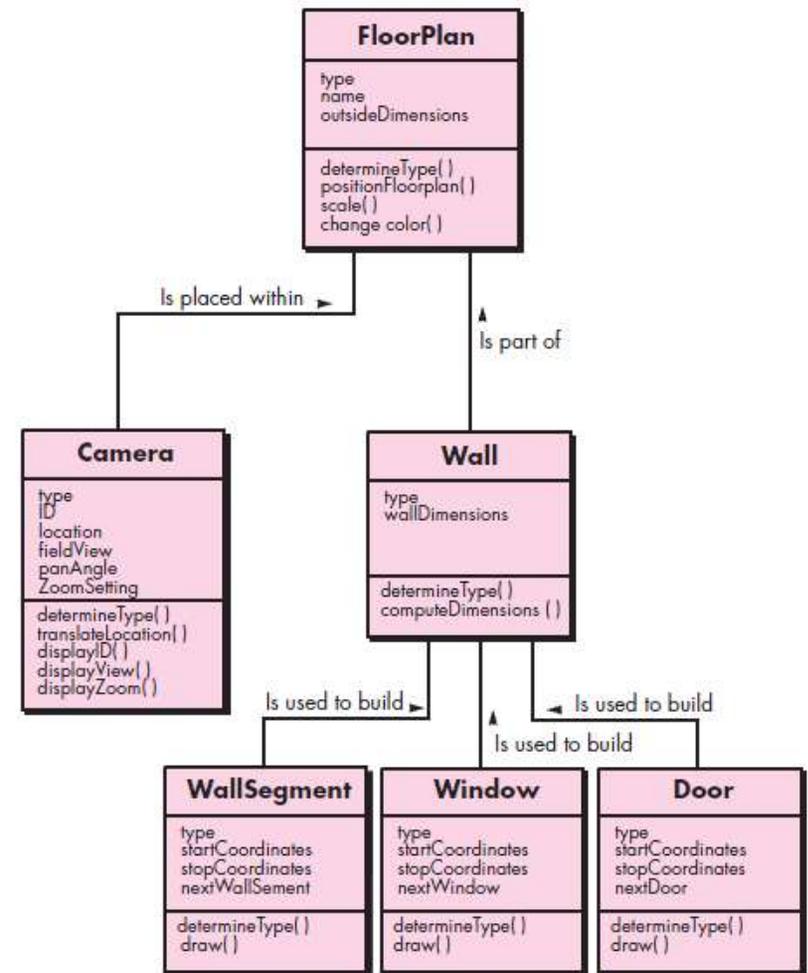
- ⦿ Roles que desempeñan los que interactúan con el sistema
- ⦿ Unidades organizacionales relevantes
- ⦿ Lugares que establecen el contexto del sistema
- ⦿ Estructuras que definen clases de objetos



## Modelado basado en clases

### Características de selección

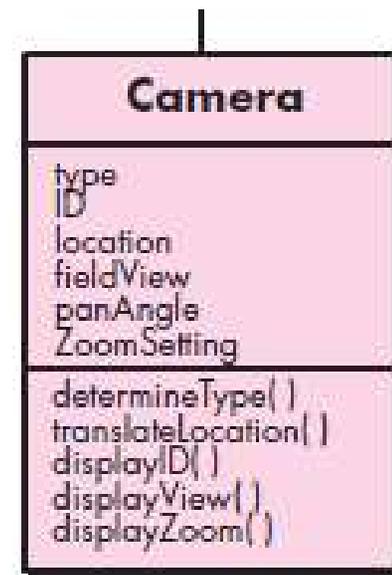
- Información retenida
- Servicios necesarios
- Atributos múltiples
- Atributos comunes
- Operaciones comunes
- Requerimientos esenciales





## Modelado basado en clases

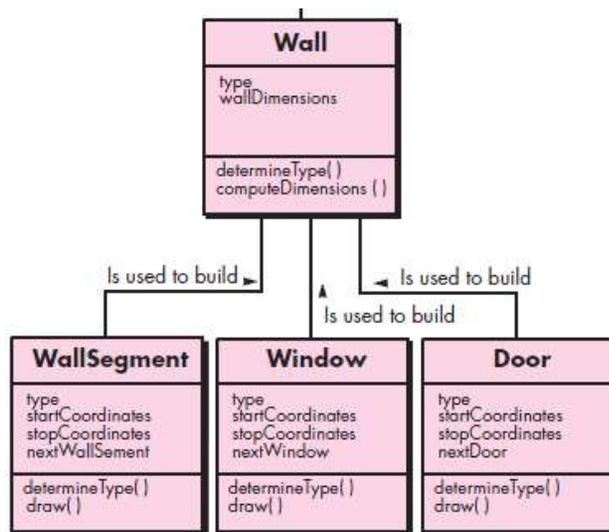
- Especificar atributos
- Definir operaciones
  - Que manipulan datos
  - Que realizan cálculos
  - Que consultan el estado del objeto
  - Que vigilan ocurrencias de eventos



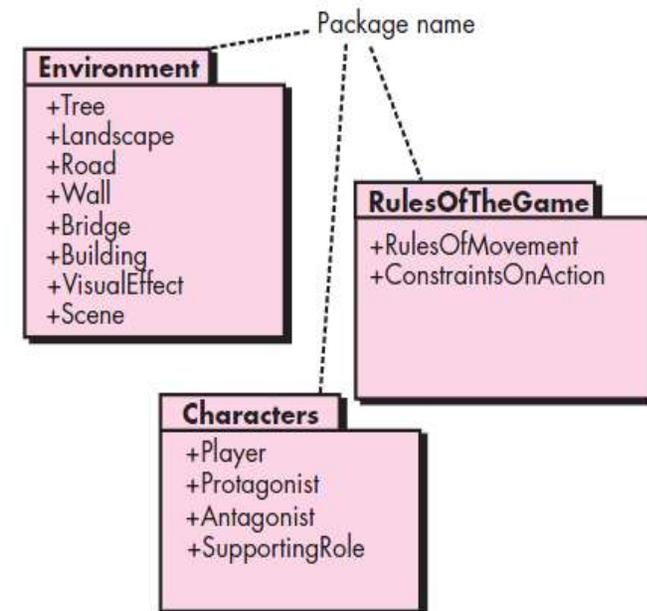


## Modelado basado en clases

### Asociaciones y dependencias



### Paquetes





## Bibliografía



- *Ingeniería de software . Teoría y Práctica* – S. L. Pfleeger  
Capítulo 4 – La determinación de los requerimientos.
- *Ingeniería del software. Un enfoque práctico* – R. Pressman  
Capítulo 5 – Comprensión de los requerimientos.  
Capítulo 6 – Modelado de los requerimientos: escenarios, información y clases de análisis.  
Capítulo 7 – Modelado de los requerimientos: flujo, comportamiento, patrones y webapps.

Template: [www.slidescarnival.com](http://www.slidescarnival.com)

Mg. M. Clara Casalini. 2017.

Introducción a la ingeniería de Software – Ingeniería en Sistemas de Información

Departamento de Ciencias e Ingeniería de la Computación – Universidad Nacional del Sur