

Modelo Orientado a Objetos

1. Identificación de Objetos

- Narrativa de Procesamiento ->Análisis Gramatical
- Subrayado de sustantivos (Lista de Objetos Potenciales)
- Tipificación y filtrado (Obtener Objetos del Sistema)
- Determinación de Atributos (Adjetivos)
- Determinación de Métodos (Verbos)

2. Casos de Uso(Escenarios ó Escenas de uso)

3. Modelado CRC (Tarjetas Índice)

4. Definición de Jerarquías de Clases

Modelo Orientado a Objetos

- 5. Validación
- 6. Modelo Objeto-Relación
- 7. Modelo Objeto-Comportamiento

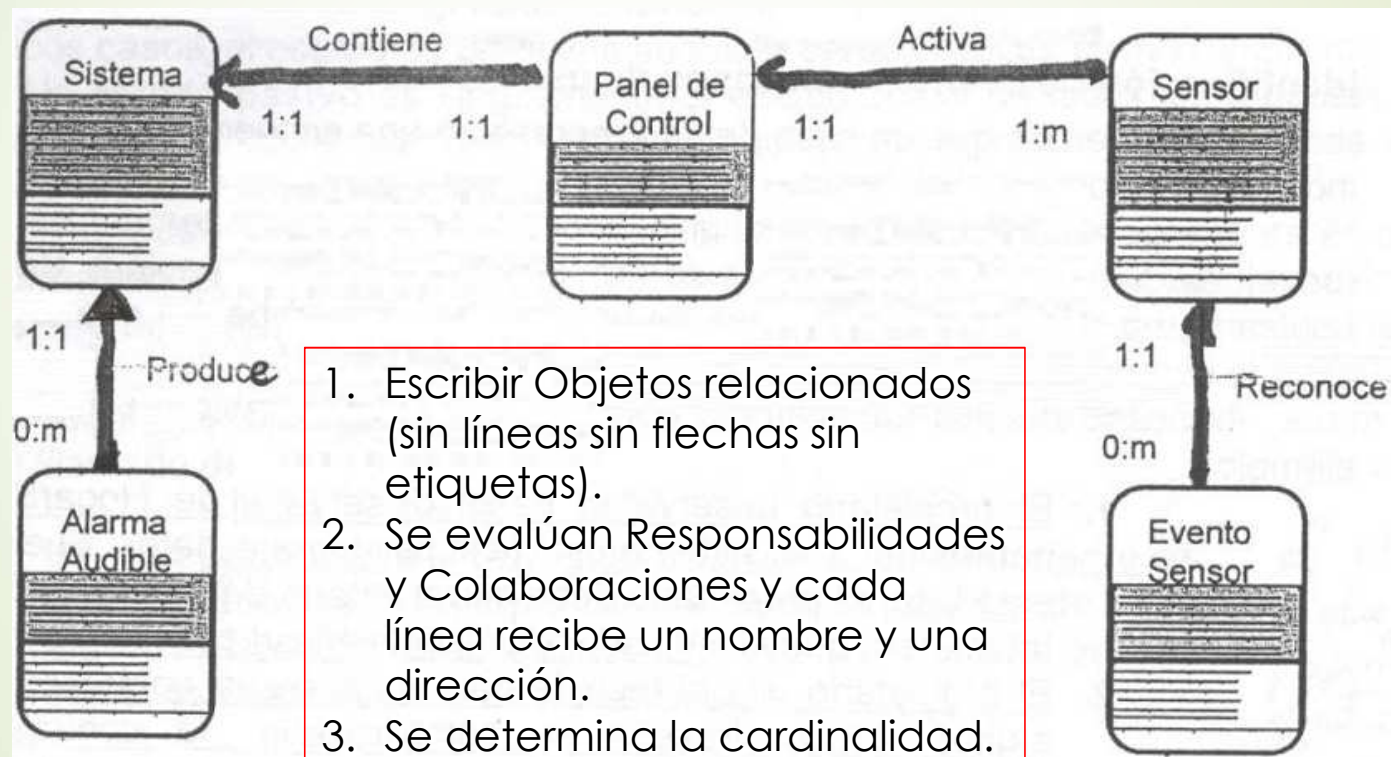
Modelo Orientado a Objetos

6. Modelo Objeto Relación

- En las tarjetas índice están presentes las responsabilidades y los colaboradores que hacen posible cada responsabilidad.
- **Una Relación** existe entre dos clases cualesquiera que estén conectadas. Dos clases están conectadas cuando una brinda colaboración a otra.

Modelo Orientado a Objetos

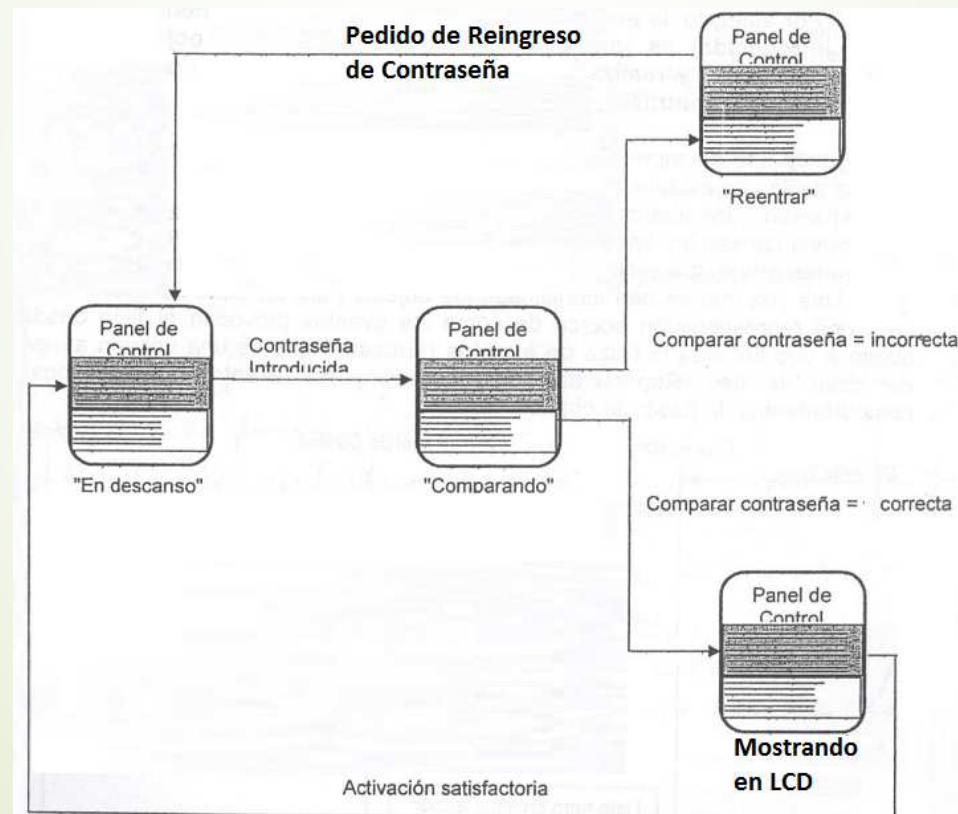
6. Modelo Objeto Relación



Modelo Orientado a Objetos

7. Modelo Objeto Comportamiento

- Representa el comportamiento Dinámico en el tiempo.



Transición[condiciones, actividades].

Modelo Orientado a Objetos

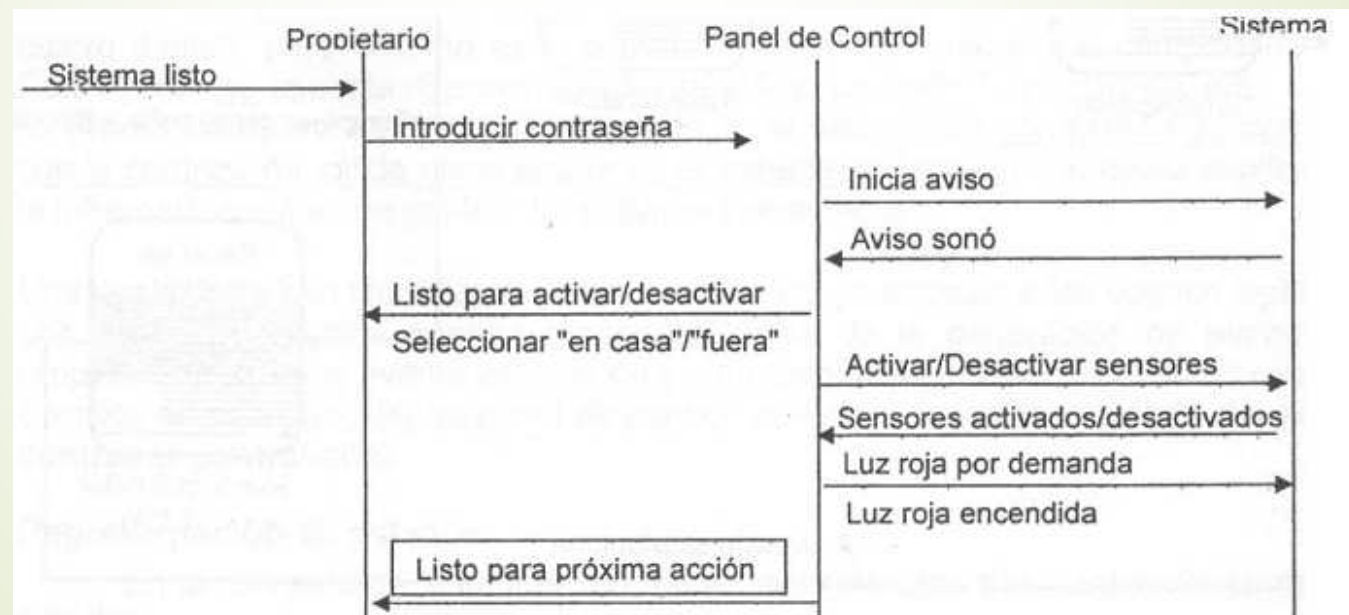
7. Modelo Objeto Comportamiento

- ¿Como se lo construye?
1. Evaluar todos los casos de uso para comprender la secuencia de interacción dentro del sistema.
 2. **Identificar Eventos** que dirigen la secuencia de interacción.
 3. Crear una traza de Eventos para cada caso de uso.
 4. Construir un diagrama de transición de estados para el sistema.
 5. Revisar el modelo para verificar exactitud y consistencia.

Modelo Orientado a Objetos

➤ Modelo de trazas de Eventos

- Modelo dinámico que **representa todo el sistema**



Modelo Orientado a Objetos

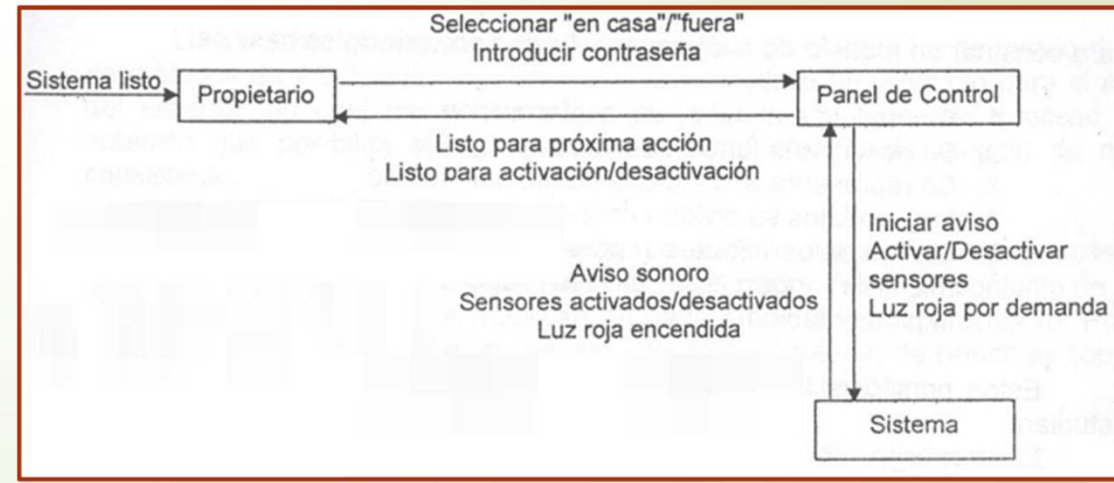
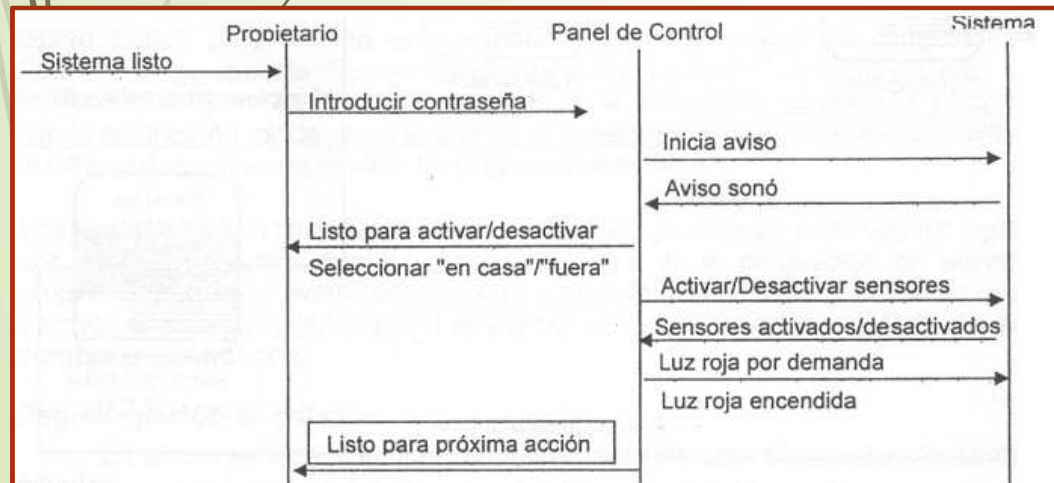
➤ Modelo de Flujo de Eventos

- Una vez desarrollada la traza de eventos por cada caso de uso, todos aquellos que provoquen transiciones entre objetos del sistema pueden incluirse en un conjunto de eventos de entradas y eventos de salidas desde cada objeto.



Modelo Orientado a Objetos

M. de trazas de Eventos (Diagrama de Secuencia) M. De Flujo de Eventos (Diagrama de Colaboración)



Análisis Orientado a Objetos

- **El AOO, responde a las siguientes preguntas:**
 - ¿Cómo se caracteriza un sistema o producto a construir para que sea tratado por la Ingeniería de Software OO?
 - ¿Cuáles son los objetos relevantes?
 - **¿Cómo se relacionan entre sí?**
 - **¿Cómo se comportan los objetos en el contexto del sistema?**
 - **¿Cómo se modela o especifica un problema de manera tal de crear un diseño eficaz?**

Análisis Orientado a Objetos

► Los 5 principios básicos para construir un Modelo de Análisis

1. Se modela el dominio de la información.
2. Se describe la función del módulo.
3. Se representa el comportamiento del modelo.
4. Los modelos se dividen para mostrar más detalles.
5. Los modelos iniciales representan la esencia del problema mientras que los últimos aportan detalles de la implementación.

Análisis Orientado a Objetos

► Tareas generales a todo Modelo de AOO

- El propósito del AOO es definir todas las clases (y las relaciones y comportamientos asociados a ellas) que son relevantes para el problema a resolver
1. Los requisitos básicos del usuario deben comunicarse entre el cliente y el ingeniero de software.
 2. Identificar las clases (atributos y métodos).
 3. Especificar una jerarquía de clases.
 4. Representar las relaciones objeto a objeto (conexiones).
 5. Modelar el comportamiento del objeto
 6. Repetir iterativamente las tareas 1 a 5 hasta completar el modelo.

Enfoque Convencional y Enfoque OO

► Enfoques Convencionales(AE)

- Hace un fuerte uso de la descomposición funcional(DFDs).
- Separa los datos (y flujos de datos) de los procesos transformadores.

► Enfoques Orientados a Objetos (CRC)

- Realiza la descomposición en objetos

Enfoque Convencional y Enfoque OO

► El panorama del AOO

► Cada método de AOO, introduce:

- Un producto(metodología) para el análisis del sistema.
- Un conjunto de modelos que evolucionan
- Una notación que posibilita crear modelos de manera consistente.

► Métodos de AOO:

- El método de Booch
- El método de Coad y Jourdon
- El método de Jacobson
- El método de Rumbaugh
- El método Unified (Booch, Rumbaugh, Wirfs-Brock)

Enfoque Convencional y Enfoque OO

► El método de Booch

Caracterizado por su "*micro proceso de desarrollo*" y un "*macro proceso de desarrollo*". Aplica un enfoque evolutivo.

► Etapas del método (extracto reducido):

► Identificar clases y objetos

- Identificar escenarios relevantes (significativos).

► Identificar la semántica de clases y objetos

- Definir operaciones para satisfacer las responsabilidades.
- Buscar colaboraciones entre objetos.

► Identificar relaciones entre clases y objetos

- Validar los escenarios por revisión completa.

► Realizar una serie de refinamientos

- Producir diagramas apropiados para el trabajo realizado en los puntos anteriores.
- Definir jerarquías de clases apropiadas

Enfoque Convencional y Enfoque OO

► El método de Coad y Jourdon

Considerado el método de AOO mas sencillo de aprender. La notación del modelo es simple y las reglas para desarrollar el modelo son evidentes.

► Etapas del método:

- Identificar objetos usando el criterio de "qué buscar".
- Definir una estructura de generalización - especificación.
- Definir una estructura de todo - parte.
- Identificar temas (representaciones de componentes de subsistemas).
- Definir atributos.
- Definir servicios.

Enfoque Convencional y Enfoque OO

► Etapas genéricas (procesos generales) del AOO:

- Obtener los requisitos del cliente para el Sistema OO.
 - Identificar escenarios o casos de uso.
 - Construir un modelo de requisitos.
- Seleccionar clases y objetos **usando requisitos básicos** como guías.
- **Identificar atributos y operaciones** para cada objeto del sistema.
- **Definir estructuras y jerarquías** que organicen las clases.
- Construir un **modelo objeto - relación**.
- Construir un **modelo objeto - comportamiento**.
- Revisar el modelo de análisis OO en relación a los casos de uso/escenarios.