

Ejemplo de UML con Racional Rose

PROBLEMA DEL ELEVADOR

Tabla de Contenidos

1. Problema del Elevador
 2. Lenguaje de Modelamiento Unificado
 3. Análisis
 - 3.1 Diagrama de Casos de Uso
 - 3.2 Diagrama de Clases
 - 3.3 Diagrama de Transición de estados
 4. Diseño
 - 4.1 Diagrama de secuencia
 - 4.2 Diagrama de Colaboración
 5. Diseño Detallado
 - 5.1 Diagrama de Clases Detallado
 - 5.2 Descripción detallada de Operación
 - 5.3 Pseudo – Código
 6. Agradecimientos
-

0. Introducción

La ayuda de este tutorial es para mostrar como usar UML en el desarrollo de software en un medio ambiente real.

1. Problema del Elevador

El producto será instalado para controlar elevadores en un edificio con “m” pisos. El problema tiene que ver con la lógica requerida para mover elevadores entre pisos de acuerdo a las siguientes restricciones :

- Cada elevador tiene un conjunto de “m” botones, uno por cada piso. Estos se iluminan cuando son presionados y causan que el elevador se dirija al piso correspondiente. La iluminación es cancelada cuando el elevador arriba al piso señalado.
- Cada piso, excepto el primer y último piso tienen dos botones, uno para requerir que el elevador suba y el otro para que baje. Estos botones también se iluminan cuando son presionados . La iluminación se cancela cuando luego de su arribo el ascensor y se dirige a la dirección deseada.
- Cuando un elevador no tiene requerimientos, este permanece en el piso actual con sus puertas cerradas.

2. Lenguaje de Modelamiento Unificado

UML es un lenguaje de modelamiento que sólo especifica notación y semántica pero ningún proceso esta en ese momento definido. Esto es, nosotros decidimos hacer el análisis como sigue;

- Diagrama de Casos de Uso
- Diagrama de Secuencia
- Diagrama de Colaboración
- Diagrama de Clases
- Diagrama de Transición de Estados

3. Análisis

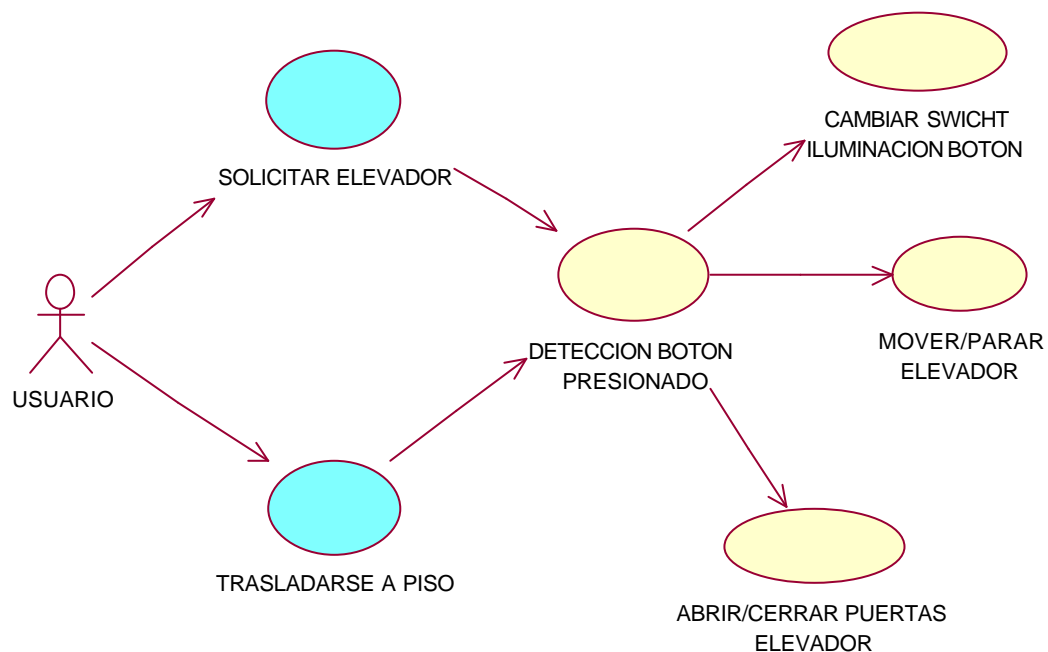
3.1. Diagrama de Casos de Uso

Descripción de un Caso de uso:

- Expresa una descripción general de como un sistema será usado.
- Provee un punto de vista de como se quiere que funcione el sistema.
Comprensible por los no expertos y también por los profesionales.

Diagrama de Casos de Uso:

DIAGRAMA DE CASOS DE USO PARA PROBLEMA DEL ELEVADOR



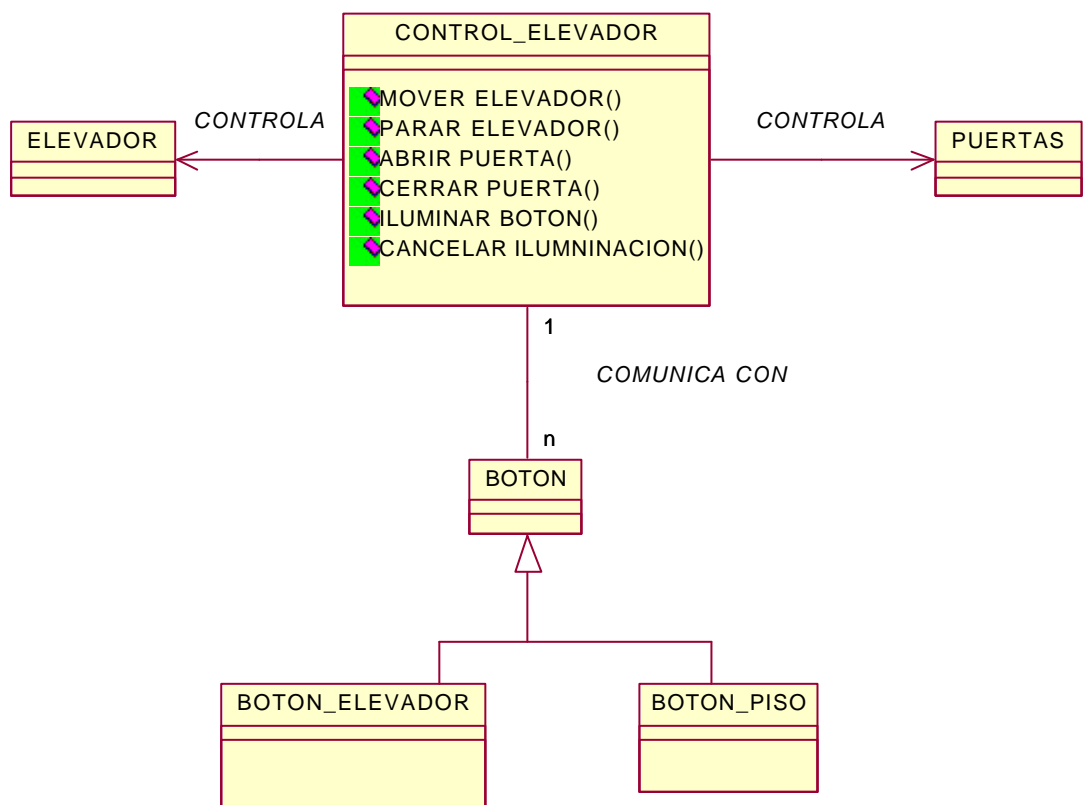
El escenario básico que puede ser obtenido desde el Diagrama de Casos de Uso:

- Arriba un pasajero que requiere el elevador y presiona el botón de elevador.
- El sistema del elevador detecta botón de piso presionado.
- El elevador se mueve hacia el piso señalado.
- El elevador procede a abrir sus puertas.
- Pasajero selecciona y presiona botón de piso.
- Elevador cierra las puertas.
- El elevador se mueve al piso señalado.
- El elevador abre las puertas.
- Pasajero sale del ascensor.
- El elevador cierra las puertas.

3.2. Diagrama de Clases

Los Diagramas de Clases muestran la estructura estática del objeto, su estructura interna, y sus relaciones.

Diagrama de Clases :



3.3. Diagrama de Transición de Estados

Un diagrama de estado muestra la secuencia de estados por los que un objeto pasa durante su ciclo de vida, en respuesta a un estímulo, conjuntamente con sus respuestas y acciones

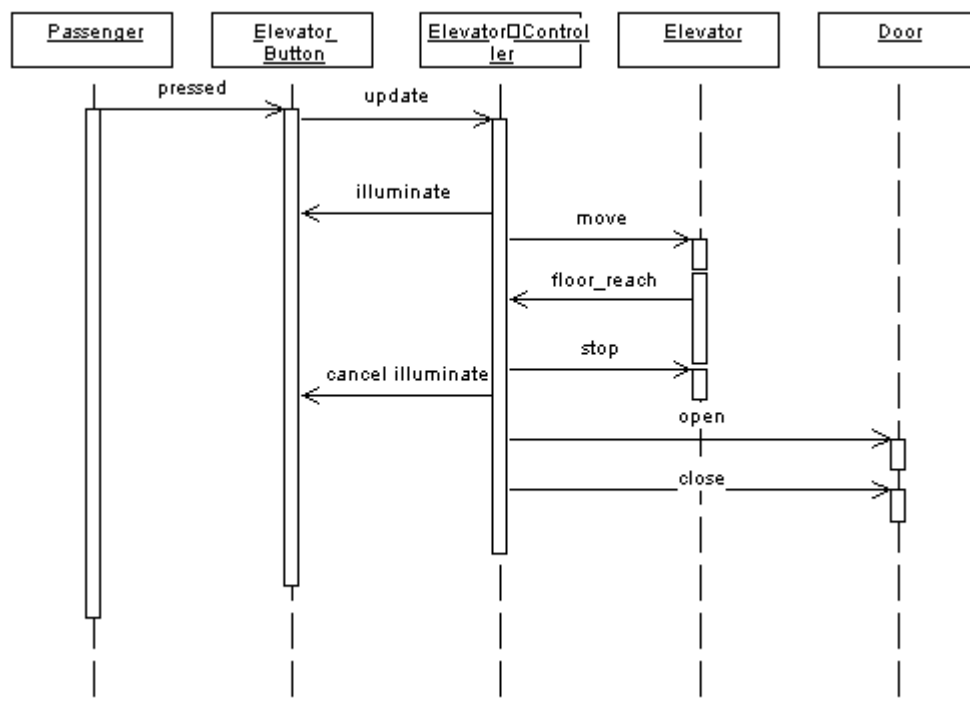
4. Diseño

La fase de diseño deberá producir : el diagrama de clases detallado,.diagramas de colaboración, diagramas de secuencia, diagramas de estado, y los diagramas de actividad. Sin embargo, el problema del elevador es demasiado simple como para un diagrama de actividad. Esto es , nosotros no vamos a usar un diagrama de actividad para el problema del elevador.

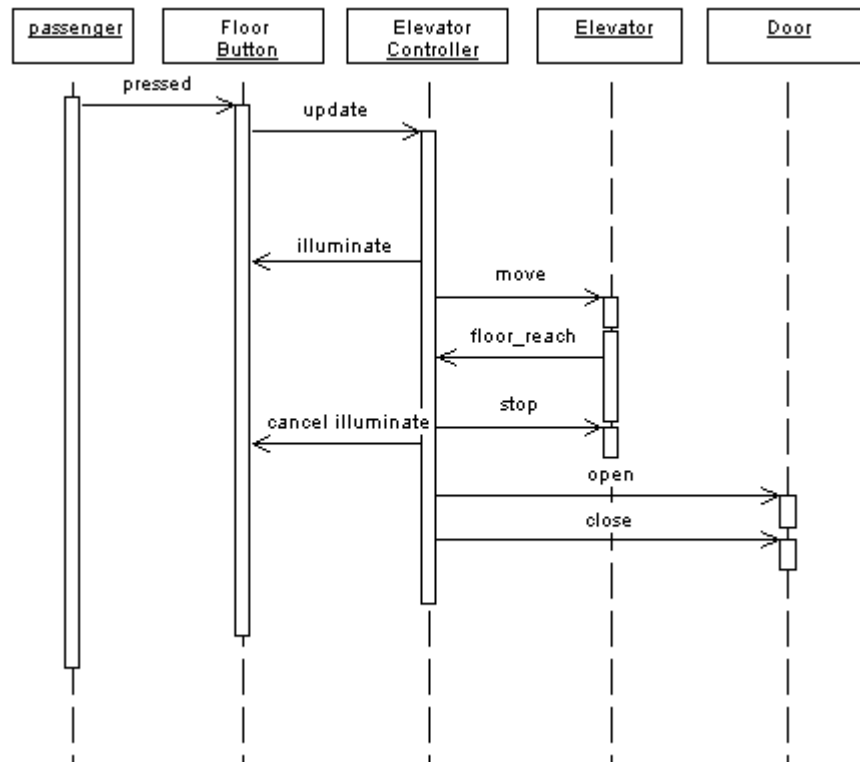
4.1. Diagrama de Secuencia

Los diagramas de secuencia o colaboración transmiten información similar pero expresada en formas diferentes. Un diagrama de secuencia muestra la secuencia explícita de mensajes adecuados para modelar un sistema en tiempo real, mientras que un diagrama de colaboración muestra las relaciones entre objetos y como ellos colaboran para satisfacer el requerimientos de una actor.

Sequence Diagrams:



Sequence Diagram for Serving Elevator Button

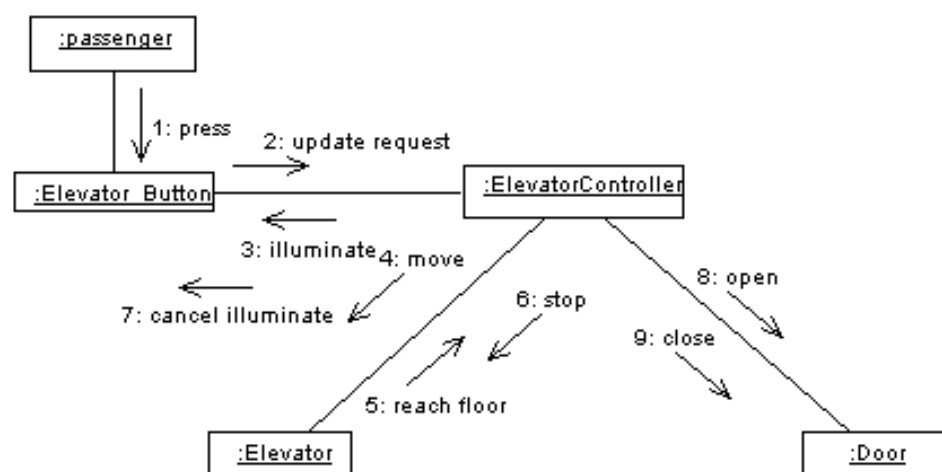


Sequence Diagram for Serving Door Button

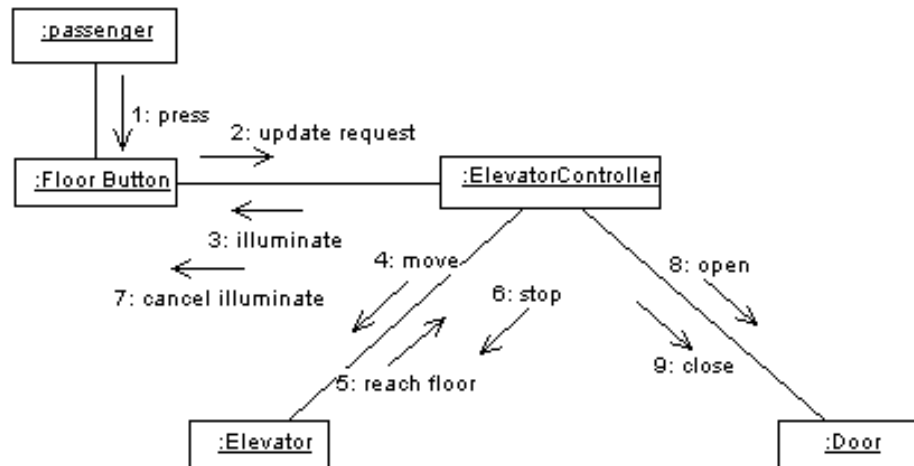
4.2. Diagrama de Colaboración

- Describe el conjunto de interacciones entre clases o tipos.
- Muestra las relaciones entre objetos.

Collaboration diagrams:



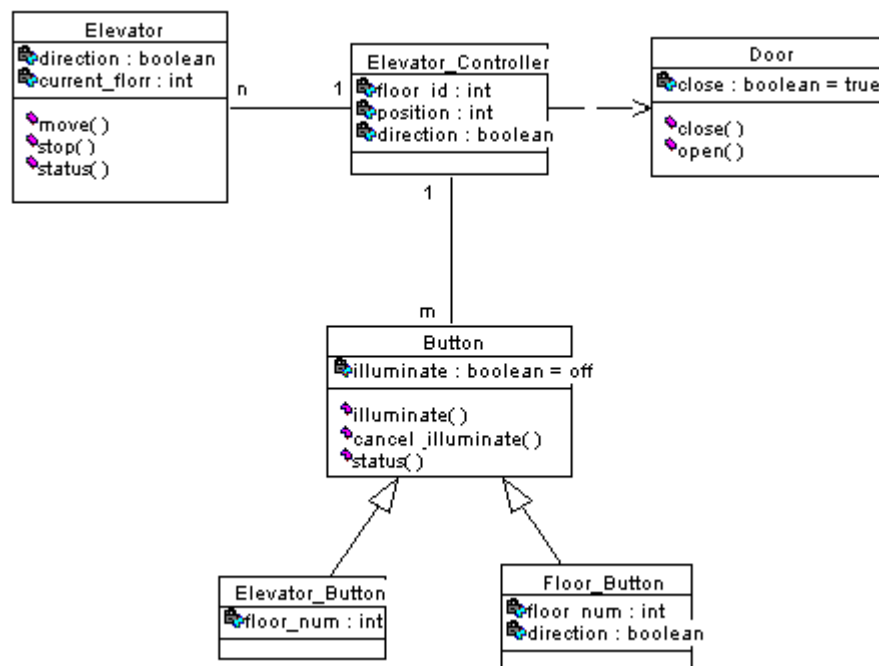
Collaboration Diagram for Serving Elevator Button



Collabracion Digaram for Serving Door Button

5. Diseño detallado

5.1. Diagrama de Clases detallado



5.2. Descripción del detalle de una Operación

Module Name	Control_Elevador::Loop_Control_Elevador
Module Type	Method
Input Argument	None
Output Argument	None
Error Message	None
File Access	None
File Change	None
Method Invoke	boton::iluminar, boton::cancelar_iluminacion, puerta::abrir, puerta::cerrar, elevador::mover, elevador::parar
Narrative	

5.3. Pseudo - Código

```
void control_elevador (void)
{
    WHILE un boton este presionado
        IF boton no esta encendido (on)
        {
            boton::iluminar;
            update lista peticiones;
        }
        ELSE IF elevador esta subiendo
        {
            IF no hay peticion de parar en piso X
                elevador::mover un piso arriba;
            ELSE
        }
}
```