

---

# **UN SISTEMA BASADO EN CONOCIMIENTO PARA PLANIFICAR ENTRENAMIENTO**

## **MODELADO CONCEPTUAL EN COMMONKADS**

---

**David González Torres  
Pablo Seoane Vázquez  
Carlos Vieites Costoya  
Diego Doldán Francos**

**Grupo de prácticas: 3.1**

**Desarrollo de Sistemas Inteligentes  
Universidade da Coruña  
Curso 2024/2025**



# Índice

<b>1. Modelo de Conocimiento.</b>	<b>1</b>
1.1. Fase de Identificación . . . . .	1
1.1.1. Glosario . . . . .	1
1.1.2. Escenarios . . . . .	1
1.1.3. Elementos reutilizables . . . . .	2
1.2. Fase de Especificación . . . . .	2
1.2.1. Metodología empleada . . . . .	2
1.2.2. Plantilla anotada . . . . .	3
1.2.3. Esquema inicial del dominio . . . . .	5
1.2.4. Estructura inferencial y Mapeo . . . . .	6
1.2.5. Tarea . . . . .	7
1.2.6. Esquema del dominio final . . . . .	8
1.3. Fase De Refinamiento . . . . .	10
1.3.1. Validación . . . . .	10
1.3.2. Bases de conocimientos . . . . .	12
<b>2. Modelo de Comunicación</b>	<b>14</b>
2.1. Plan de comunicación general . . . . .	14
2.2. Descripción de las Transacciones . . . . .	15
2.3. Especificación de las transacciones . . . . .	16



# 1. Modelo de Conocimiento.

## 1.1. Fase de Identificación

El modelo está basado en la segunda tarea del OM-3: Preparar una rutina de entrenamiento personalizada

### 1.1.1. Glosario

- Rutina "full body": rutina de entrenamiento caracterizada por que todos los días de entrenamiento se trabajan en mayor o menor medida todos o gran parte de los grupos musculares.
- Rutina HIIT (High Intensity Interval Training): Es un método de entrenamiento cardiovascular de alta intensidad que alterna periodos cortos de ejercicio intenso con periodos de descanso o ejercicio a menor intensidad.
- Rutina Push-pull-leg: Es un tipo de rutina de entrenamiento que consta de 3 tipos de días de entrenamiento: día de empujes (pecho, hombro y tríceps), día de tirones (espalda y bíceps) y día de pierna.

### 1.1.2. Escenarios

Un chico de 20 años que quiere empezar a entrenar en el gimnasio con el objetivo de ganar masa muscular. Tiene un peso de 70 kg, mide 180 cm y tiene una disponibilidad semanal para entrenar de 5 días (de lunes a viernes) de los cuales quiere entrenar 3 con una duración de hora y media. Además tiene una lesión en la pierna. Se le ofrece un plan de entrenamiento personalizado de 3 días a la semana siguiendo una rutina "full body".

Día 1 (lunes): Press de banca: 3 series 10 repeticiones 60 kg Remo con barra: 3 series 10 repeticiones 30 kg Curl de bíceps: 4 series 12 repeticiones 7 kg

Día 2 (miércoles): Dominadas: 3 series 5 repeticiones Fondos en paralelas: 3 series 6 repeticiones Elevaciones laterales: 3 series 12 repeticiones 5 kg

Día 3 (viernes): Press inclinado con mancuernas: 3 series 8 repeticiones 10 kg Remo con mancuerna: 3 series 8 repeticiones 10 kg Abdominales en polea: 3 series 12 repeticiones 10 kg

### 1.1.3. Elementos reutilizables

No hay ningún elemento reutilizable.

## 1.2. Fase de Especificación

### 1.2.1. Metodología empleada

La metodología empleada ha sido la *Middle-out* ya que inicialmente se ha hecho el diseño de la estructura inferencial con la plantilla de la tarea y después se ha definido el conocimiento del dominio y de las tareas y se ha mapeado a la estructura inferencial. Se ha escogido esta metodología porque la plantilla inferencial nos ofrece una propuesta que se adecúa bastante a la estructura inferencial de la tarea, lo que nos permite un desarrollo más directo que si comenzáramos con la definición del dominio.

### 1.2.2. Plantilla anotada

Inicialmente para nuestra tarea escogimos la plantilla de Planificación, que a priori parecía que se ajustaba más a nuestra tarea:

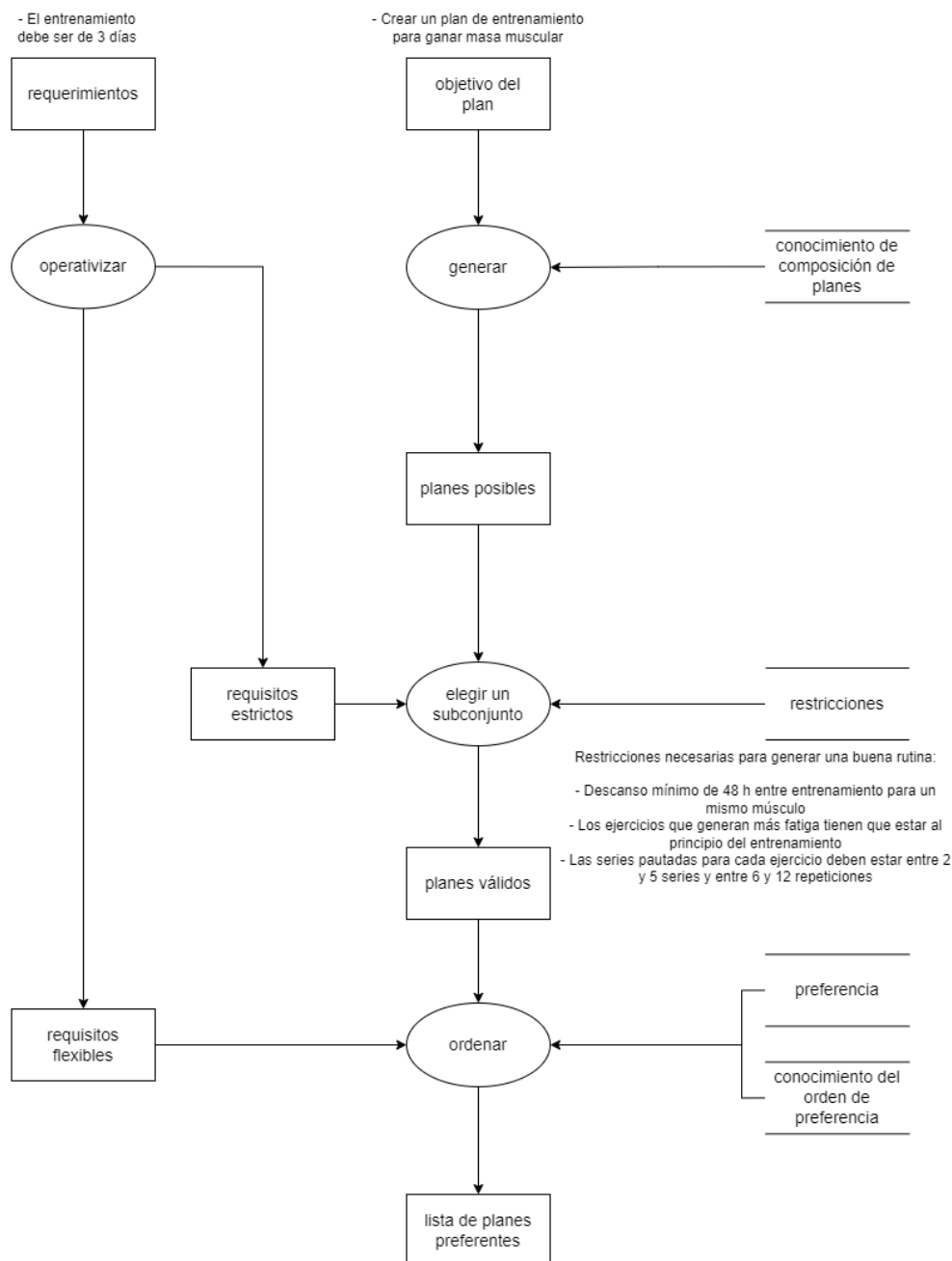


Figura 1: Plantilla de planificación anotada

Tras intentar anotar la plantilla teniendo en cuenta nuestro dominio nos dimos cuenta que no era viable implementar la tarea basándonos en este algoritmo (debido a que para generar todos los planes posibles nos encontrábamos con infinitas posibilidades), y en su lugar nos decantamos por la plantilla de Configuration Design:

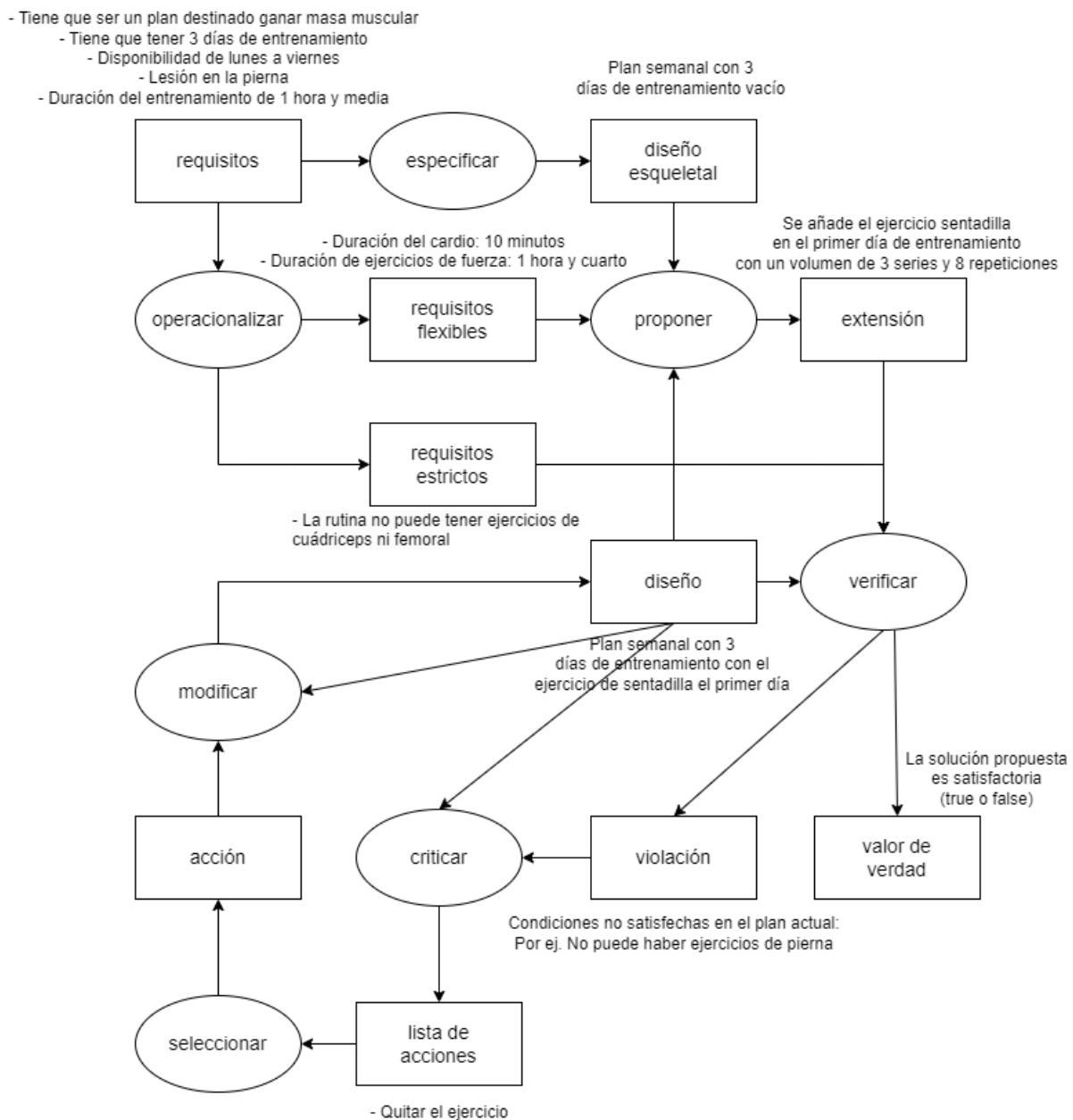


Figura 2: Configuration Design



### 1.2.3. Esquema inicial del dominio

Boceto del que se parte que, posteriormente, se irá refinando al realizar el mapeo de las inferencias, etc.

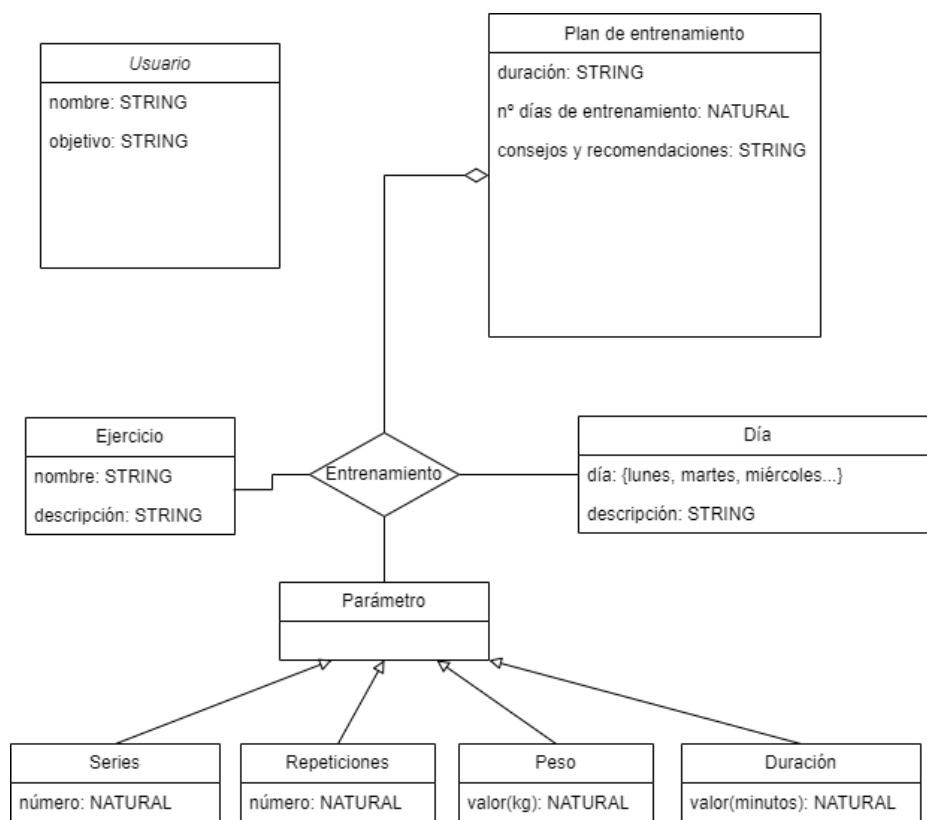


Figura 3: Esquema del dominio inicial

En un primer momento identificamos estas clases y relaciones necesarias para hacer la representación de nuestro problema. Contábamos con un plan de entrenamiento que se debía generar al usuario con un objetivo determinado que tenía una duración, un n<sup>o</sup> días de entrenamiento, y un conjunto de ejercicios realizados en un día y con unos parámetros específicos dependientes del tipo de ejercicio.

## 1.2.4. Estructura inferencial y Mapeo

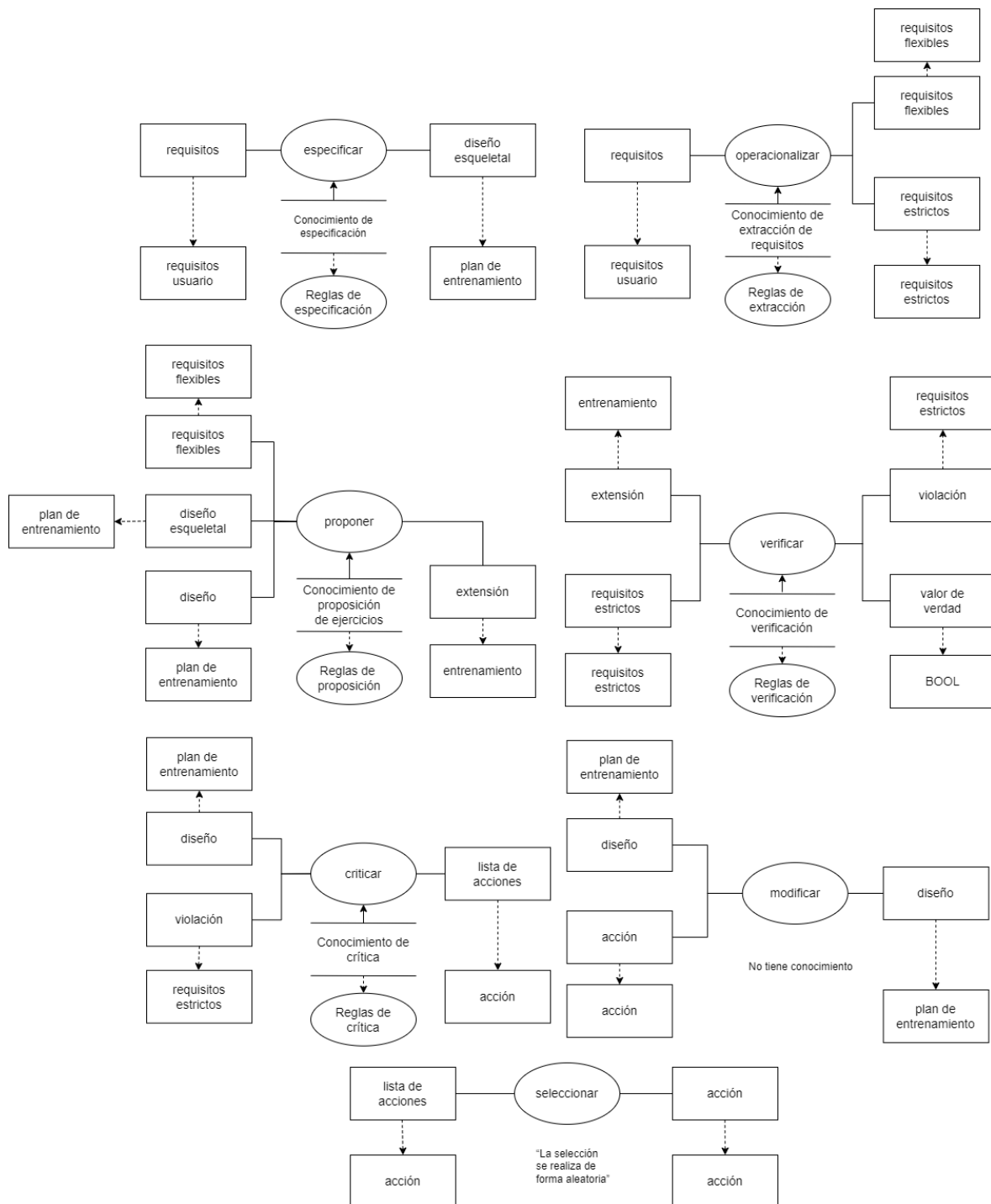


Figura 4: Mapeo de roles

En estas imágenes se muestra el mapeo de las inferencias y la interacción entre los roles dinámicos y estáticos. Se muestran así las entradas y las salidas de las inferencias relacionadas con los conceptos de nuestro dominio y las reglas de conocimiento que utilizan.

### 1.2.5. Tarea

```

TASK configuration-design;
  ROLES:
    INPUT: requirements: "The user is a 20-year-old man, 70 kg and
      180 cm tall. He wants to gain muscle mass. He is available to train
      five days a week, from Monday to Friday, but he wants to train
      only three of those days, for 1 hour and 30 minutes each day.
      He has a leg injury";
    OUTPUT: design: "The resulting design is a personalized
      3-day-a-week training plan that follows a "full-body" routine;
END TASK configuration-design;

TASK-METHOD propose-and-revise;
  REALIZES: configuration-design;
  DECOMPOSITION:
    INFERENCES: operationalize, specify, propose, verify, critique,
      select, modify;
  ROLES:
    INTERMEDIATE:
      soft-requirements: "requirements to be used as preferences";
      hard-requirements: "requirements that are hard constraints";
      skeletal-design: "set of design elements";
      extension: "a single new value for a design element";
      violation: "constraint violated by the current design";
      truth-value: "boolean indicating result of the verification";
      action-list: "ordered list of possible repair (fix) actions";
      action: "a single repair action";
    CONTROL-STRUCTURE:
      operationalize(requirements -> hard-requirements
        + soft-requirements);
      specify(requirements -> skeletal-design);
      WHILE NEW-SOLUTION propose(skeletal-design + design
        + soft-requirements -> extension) DO
        design := extension ADD design;
        verify(design + hard-requirements
          -> truth-value + violation);
        IF truth-value == false
        THEN
          critique(violation + design -> action-list);
        REPEAT
          select(action-list -> action);

```

```

    modify(design + action -> design);
    verify(design + hard-requirements
        -> truth-value + violation);
    UNTIL truth-value == true;
    END REPEAT
  END IF
END WHILE
END TASK-METHOD propose-and-revise;

```

### 1.2.6. Esquema del dominio final

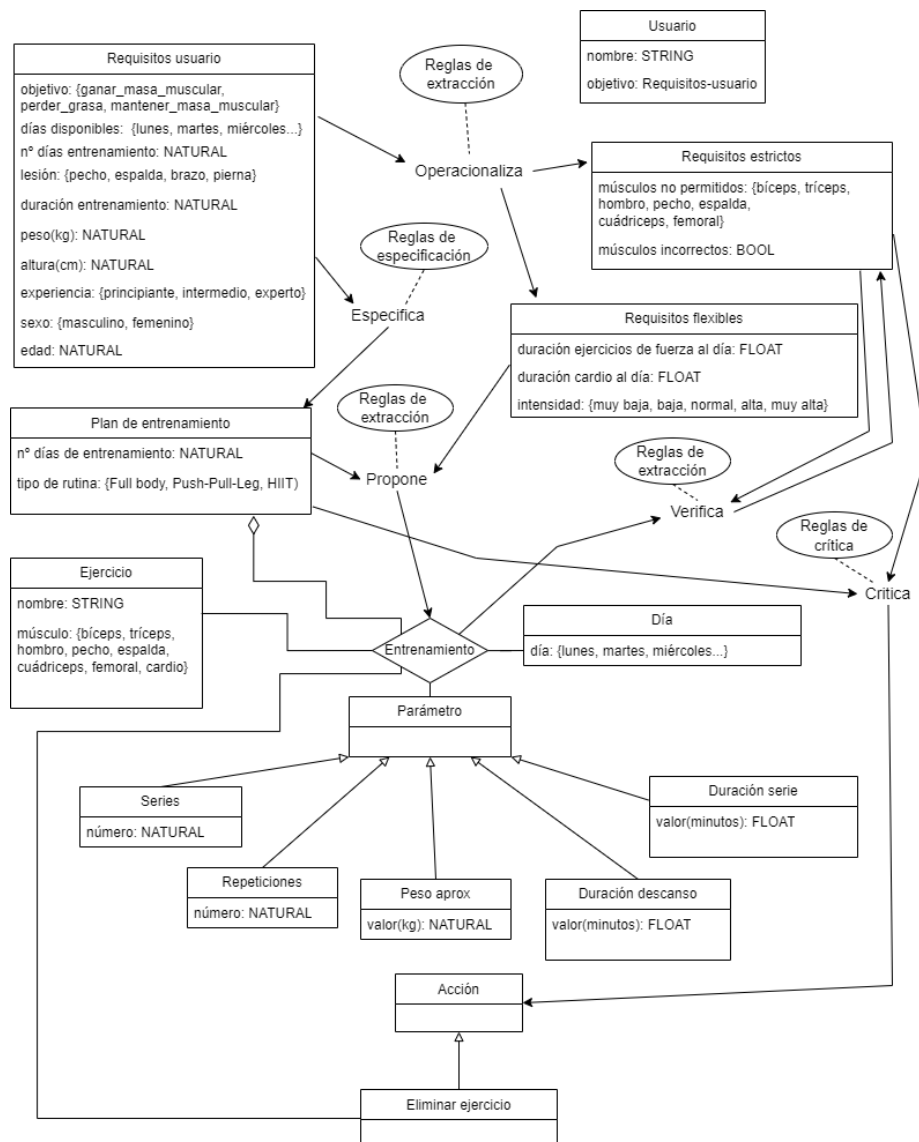


Figura 5: Esquema del dominio final

La Figura 5 muestra el diagrama del dominio final, el cual presenta diferencias con respecto al diagrama inicial. Estas modificaciones se realizaron durante el proceso de mapeo de las inferencias, con el objetivo de reflejar de forma adecuada los conceptos y relaciones necesarios para el funcionamiento del sistema inferencial.

### 1.3. Fase De Refinamiento

#### 1.3.1. Validación

Dominio	Modelo	Explicación
El usuario es un hombre de 20 años, de 70 kg y 180 cm de altura. Nunca ha entrenado y quiere ganar masa muscular y puede entrenar 5 días a la semana, de lunes a viernes de los cuales quiere entrenar 3 con una duración de hora y media. Además tiene una lesión en la pierna.	Requisitos-Usuario.objetivo = ganar-masa-muscular Requisitos-Usuario.días-disponibles = {lunes, martes, miércoles, jueves, viernes} Requisitos-Usuario.n <sup>o</sup> -días-entrenamiento = 3 Requisitos-Usuario.lesión = pierna Requisitos-Usuario.duración-entrenamiento = 90 Requisitos-Usuario.peso = 70 Requisitos-Usuario.altura = 180 Requisitos-Usuario.experiencia = principiante Requisitos-Usuario.sexo = masculino Requisitos-Usuario.edad = 20	El sistema vuelca la información que recibe del usuario en Requisitos-Usuario:
	OPERACIONALIZAR Si Requisitos-Usuario.objetivo = ganar-masa-muscular y Requisitos-Usuario.experiencia = baja entonces Requisitos-Flexibles.intensidad = normal; Si Requisitos-Usuario.lesión: pierna entonces Requisitos-Estrictos.músculos-no-permitidos: {cuádriceps, femoral}; Si Requisitos-Usuario.objetivo = ganar-masa-muscular entonces Requisitos-Flexibles.duracion_cardio_al_dia = Requisitos-Usuario.duración_entrenamiento * 0,08 y Requisitos-Flexibles.duracion_ejecicios_fuerza_al_dia = Requisitos-Usuario.duración_entrenamiento * 0,88	Se dividen los requisitos del usuario en estrictos (que deben cumplirse para que el plan de entrenamiento sea válido) y flexibles (preferencias).
	ESPECIFICAR Si Requisitos-usuario.días-disponibles > Requisitos-usuario.n <sup>o</sup> -días-entrenamiento entonces Plan-de-entrenamiento.n <sup>o</sup> -días-de-entrenamiento = Requisitos-usuario.n <sup>o</sup> -días-entrenamiento y para todos los Plan-de-entrenamiento.Entrenamiento: Entrenamiento.Día = lunes o miércoles o viernes; Si Requisitos-usuario.objetivo = ganar-masa-muscular y Requisitos-usuario.n <sup>o</sup> -días-entrenamiento <= 3 entonces Plan-de-entrenamiento.tipo-de-rutina = Full body	Se genera un diseño inicial para el plan de entrenamiento (especificando el tipo de rutina y los días de entrenamiento a partir de los requisitos de usuario)

	<b>PROPONER</b> Si Plan-de-entrenamiento.tipo-de-rutina = Full-body y Requisito-flexible.intensidad = normal y Plan-de-entrenamiento.Entrenamiento.Día.día no tiene ejercicios entonces Entrenamiento.Día.día = Plan-de-entrenamiento.Entrenamiento.Día.día; Entrenamiento.Ejercicio.nombre = Press banca; Entrenamiento.Ejercicio.músculo = Pecho; Entrenamiento.Ejercicio.Series.número = 3; Entrenamiento.Ejercicio.Repeticiones.número = 10; Entrenamiento.Ejercicio.Peso-aprox.valor = 60; Entrenamiento.Ejercicio.Duración-descanso.valor = 3;	El sistema propone añadir un ejercicio al plan teniendo en cuenta el diseño actual y los requisitos flexibles
	<b>VERIFICA</b> valor-de-verdad = TRUE	Se comprueba si el ejercicio propuesto no cumple con algún requisito estricto
	<b>PROPONER</b> Si Plan-de-entrenamiento.tipo-de-rutina = Full-body y Requisito-flexible.intensidad = normal y Plan-de-entrenamiento.Entrenamiento.Día.día no tiene ejercicios de Cuádriceps o Femoral entonces Entrenamiento.Día.día = Plan-de-entrenamiento.Entrenamiento.Día.día; Entrenamiento.Ejercicio.nombre = Sentadilla; Entrenamiento.Ejercicio.músculo = Cuádriceps; Entrenamiento.Ejercicio.Series.número = 4; Entrenamiento.Ejercicio.Repeticiones.número = 10; Entrenamiento.Ejercicio.Peso-aprox.valor = 50; Entrenamiento.Ejercicio.Duración-descanso.valor = 4;	
	<b>VERIFICAR</b> Si Entrenamiento.Ejercicio.músculo = Requisitos-Estrictos.músculos-no-permitidos entonces Requisitos-estrictos.músculos-incorrectos = true y valor-de-verdad = FALSE	
	<b>CRITICAR</b> Si Requisito-estrictos.músculos-incorrectos = TRUE y Requisitos-Estrictos.músculos-no-permitidos = Plan-de-entrenamiento.Entrenamiento.Ejercicio.músculo entonces Acción.Eliminar-ejercicio.Ejercicio = Plan-de-entrenamiento.Entrenamiento.Ejercicio.nombre = Sentadilla	Se mira el diseño actual y se miran los requisitos estrictos que se han violado para obtener una lista de posibles acciones que modifiquen el diseño para cumplir con los requisitos (en nuestro caso eliminar un ejercicio).
	<b>SELECCIONAR</b> La selección se hace de manera aleatoria.	Se selecciona una acción de la lista de acciones.
	<b>MODIFICAR</b> Se aplica la acción seleccionada	Se modifica el plan de entrenamiento aplicando la acción seleccionada.

Tabla 1: Simulación del comportamiento del sistema sobre el escenario planteado

### 1.3.2. Bases de conocimientos

#### Reglas de especificación

1. Si  $\text{Requisitos\_usuario.días\_disponibles} > \text{Requisitos\_usuario.n}^\circ\text{días\_entrenamiento}$   
**ESPECIFICA**  
 $\text{Plan\_de\_entrenamiento.n}^\circ\text{días\_de\_entrenamiento} = \text{Requisitos\_usuario.n}^\circ\text{días\_entrenamiento}$   
 y  $\text{maximizar\_separación}(\text{Entrenamiento.Día})$
2. Si  $\text{Requisitos\_usuario.objetivo} = \text{ganar-masa-muscular}$  y  $\text{Requisitos\_usuario.n}^\circ\text{días\_entrenamiento} > 3$   
**ESPECIFICA**  
 $\text{Plan\_de\_entrenamiento.tipo\_de\_rutina} = \text{Push-Pull-Leg}$
3. Si  $\text{Requisitos\_usuario.objetivo} = \text{ganar-masa-muscular}$  y  $\text{Requisitos\_usuario.n}^\circ\text{días\_entrenamiento} \leq 3$   
**ESPECIFICA**  
 $\text{Plan\_de\_entrenamiento.tipo\_de\_rutina} = \text{Full body}$

#### Reglas de extracción

1. Si  $\text{Requisitos\_Usuario.lesión} = \text{pierna}$   
**OPERACIONALIZA**  
 $\text{Requisitos\_Estrictos.músculos\_no\_permitidos} = \{\text{cuádriceps, femoral}\}$
2. Si  $\text{Requisitos\_Usuario.objetivo} = \text{ganar-masa-muscular}$  y  $\text{Requisitos\_Usuario.experiencia} = \text{intermedio}$   
**OPERACIONALIZA**  
 $\text{Requisitos\_Flexibles.intensidad} = \text{alta}$
3. Si  $\text{Requisitos\_Usuario.objetivo} = \text{ganar-masa-muscular}$  y  $\text{Requisitos\_Usuario.experiencia} = \text{principiante}$   
**OPERACIONALIZA**  
 $\text{Requisitos\_Flexibles.intensidad} = \text{normal}$
4. Si  $\text{Requisitos\_Usuario.objetivo} = \text{ganar-masa-muscular}$   
**OPERACIONALIZA**  
 $\text{Requisitos\_Flexibles.duración\_cardio\_al\_día} = \text{Requisitos\_Usuario.duración\_entrenamiento} * 0,08$  y  
 $\text{Requisitos\_Flexibles.duración\_ejercicios\_fuerza\_al\_día} = \text{Requisitos\_Usuario.duración\_entrenamiento} * 0,88$



## Reglas de proposición

1. Si `Plan_de_entrenamiento.tipo_de_rutina = Full-body` y `Requisito_flexible.intensidad = normal` y `Plan_de_entrenamiento.Entrenamiento.Día.día` no tiene ejercicios

### PROPONE

`Entrenamiento.Día.día = Plan_de_entrenamiento.Entrenamiento.Día.día;`  
`Entrenamiento.Ejercicio.nombre = Press banca;`  
`Entrenamiento.Ejercicio.músculo = Pecho;`  
`Entrenamiento.Ejercicio.Series = 3;`  
`Entrenamiento.Ejercicio.Repeticiones = 10;`  
`Entrenamiento.Ejercicio.Peso_aprox = 60;`  
`Entrenamiento.Ejercicio.Descanso = 3`

2. Si `Plan_de_entrenamiento.tipo_de_rutina = HIIT` y `Requisito_flexible.intensidad = muy-alta` y `Requisitos_flexibles.duración_cardio_al_día >`  
para cada `Ejercicio` en `Plan de Entrenamiento` donde `Ejercicio.músculo = cardio`  

$$\sum((Ejercicio.Duración\_serie.valor + Ejercicio.Duración\_descanso.valor) \times Ejercicio.Series.número)$$

### PROPONE

`Entrenamiento.Día.día = Plan_de_entrenamiento.Entrenamiento.Día.día;`  
`Entrenamiento.Ejercicio.nombre = Burpees`  
`Entrenamiento.Ejercicio.músculo = cardio`  
`Entrenamiento.Ejercicio.Series.valor = 4`  
`Entrenamiento.Ejercicio.Duración_serie.valor = 1.5`  
`Entrenamiento.Ejercicio.Duración_descanso.valor = 0.5`

3. Si `Plan_de_entrenamiento.tipo_de_rutina = Full-body` y `Requisito_flexible.intensidad = normal` y  
`Plan_de_entrenamiento.Entrenamiento.Día.día` no tiene ejercicios de `Cuádriceps` y `Femoral`

### PROPONE

`Entrenamiento.Día.día = Plan_de_entrenamiento.Entrenamiento.Día.día;`  
`Entrenamiento.Ejercicio.nombre = Sentadilla;`  
`Entrenamiento.Ejercicio.músculo = Cuádriceps;`  
`Entrenamiento.Ejercicio.Series.número = 4;`  
`Entrenamiento.Ejercicio.Repeticiones.número = 10;`  
`Entrenamiento.Ejercicio.Peso_aprox.valor = 50;`  
`Entrenamiento.Ejercicio.Duración_descanso.valor = 4`

## Reglas de verificación

1. Si `Entrenamiento.Ejercicio.músculo` está en `Requisitos_Estrictos.músculos_no_permitidos`

### VERIFICA

`Requisitos_estrictos.músculos_incorrectos = true` y `valor_de_verdad = false`

### Reglas de crítica

1. Si `Requisitos_estrictos.músculos_incorrectos = true` y  
`Requisitos_Estrictos.músculos_no_permitidos =`  
`Plan_de_entrenamiento.Entrenamiento.Ejercicio.músculo`

#### **CRITICA**

Acción.`Eliminar_ejercicio.Ejercicio = Plan_de_entrenamiento.Entrenamiento.Ejercicio`

## 2. Modelo de Comunicación

### 2.1. Plan de comunicación general

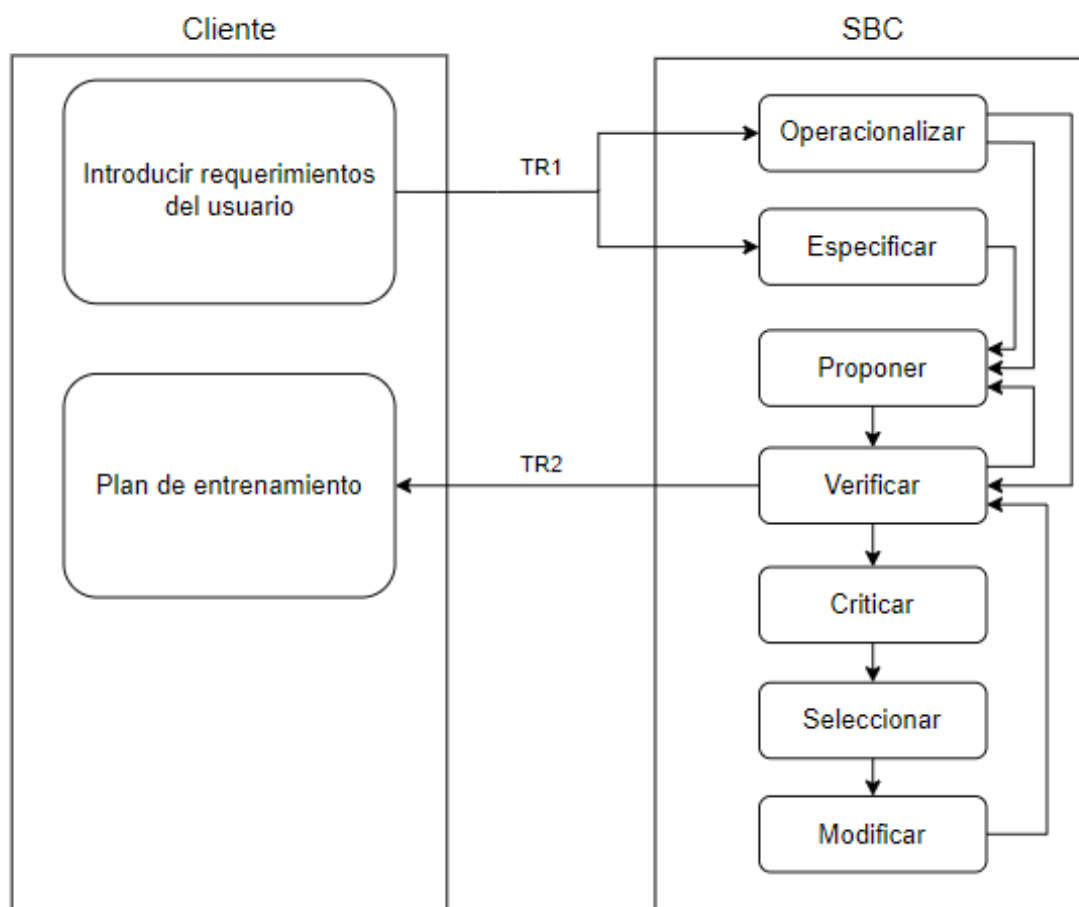


Figura 6: Diagrama de diálogo



Figura 7: Diagrama de transición

En la Figura 6 se encuentra el diagrama de dialogo, que representa la comunicación entre los agentes (el cliente y el sistema) en la tarea de preparar una rutina de entrenamiento personalizada. Se ha elaborado teniendo en cuenta los agentes que realizan la tarea y las funciones de transferencia, que en este caso no hay. Las únicas transacciones que hay son la entrada (el cliente expone sus requisitos) y la salida del sistema, que proporciona plan de entrenamiento. Estas dos transacciones están representadas en el diagrama de transición de la Figura 7.

## 2.2. Descripción de las Transacciones

Modelo CM-1 que incluya las transacciones identificadas

Modelo de comunicación	Formulario CM-1
NOMBRE DE LA TRANSACCIÓN	TR1 - Introducir los requisitos del usuario
OBJETOS DE INFORMACIÓN	Los requisitos del usuario
AGENTES INVOLUCRADOS	<i>Emisor:</i> usuario <i>Receptor:</i> SBC
PLAN DE COMUNICACIONES	Diagrama de diálogo de la figura 5.
RESTRICCIONES	El SBC debe de estar listo
ESPECIFICACIÓN DE INTERCAMBIO DE INFORMACIÓN	Es una tarea de delegación de tipo Order.

Tabla 2: CM-1 de TR1

Modelo de comunicación	Formulario CM-1
NOMBRE DE LA TRANSACCIÓN	TR2 - Enviar rutina de entrenamiento
OBJETOS DE INFORMACIÓN	La rutina de entrenamiento personalizada a los requisitos del usuario
AGENTES INVOLUCRADOS	<i>Emisor:</i> SBC <i>Receptor:</i> usuario
PLAN DE COMUNICACIONES	Diagrama de diálogo de la figura 5.
RESTRICCIONES	Se inicia la transacción cuando el SBC encuentra una rutina entrenamiento personalizada de acuerdo a los requisitos del usuario.
ESPECIFICACIÓN DE INTERCAMBIO DE INFORMACIÓN	Es un intercambio de información pura de tipo Report.

Tabla 3: CM-1 de TR2

### **2.3. Especificación de las transacciones**

Como el modelo de comunicación no es complejo, solamente participan dos agentes, no es necesaria la especificación de transacciones del formulario CM-2.