# **Model Report**

Version



Date/Time Generated:

Author:

9/10/2023 09:25:55

Valentin

EA Repository : D:\ULP\Ingenieria de Software\tp\_gym\SuperGymna.eapx



## **Table of Contents**

Model	4
claseDeDominio	4
claseDeDominio diagram	4
Note	
Note	
Note	5
Note	5
Note	6
instructor	6
State1	
Starter Use Case Model	7
Starter Use Case Model diagram	7
EL SUPER GIMNASIO	7
instructor	
socio	Q
Dueños del gimnasio	9
Recepcionista	
Generar Estadísticas de Clases Asistidas	10
Generar Listados	
Registrar alquiler	10
Registrar Venta de Productos	
Inscribirse	14
Actors	15
instructor	
socio	16
Dueños del gimnasio	
Recepcionista	17
Use Cases	
Generar Estadísticas de Clases Asistidas	19
Generar Listados	20
Registrar alquiler	20
Registrar Venta de Productos	
Inscribirse	777
Imprimir horario	23
Consultar Clases Asignadas	
A) Buscar socio	24
Aplicar descuento en abono	25
Armar clase de fitness	25
Asistir a clase	26
B) Consultar pase	
C) Consultar instructor	29
Cancelar Clases Asignadas	29
Contratar - renovar pase	30
contratar pase	21
D) Consultar clase vigente	
Dar de baja pases vencidos	32
E) Consutar tipo de clase	32
F) Buscar insumo	22
G) Aplicar descuento a bebidas energeticas	33

Generar horario	34
Generar Listado de Presentismo de Instructores entre Fechas	35
Generar Listados	36
Generar listados	36
Generar un registro de asistencia	36
H) Aplicar promocion de descuento	37
Marcar Hora de llegada y salida	38
Modificar o agregar instructores	38
Registrar venta de insumos	38
Reprogramar una clase	39

## Model

Package in package "

Model
Version Phase 1.0 Proposed
Valentin created on 9/10/2023. Last modified 9/10/2023

## claseDeDominio

Package in package 'Model'

claseDeDominio Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023

## claseDeDominio diagram

StateMachine diagram in package 'claseDeDominio'

claseDeDominio Version 1.0 Eduardo created on 8/10/2023. Last modified 8/10/2023

#### Instructor

Atributos: Nombre (String) Apellido (String) Especialización (String): "Yoga", "Pilates", etc. Hora\_Entrada (Time) Hora\_Salida (Time)

#### **Producto**

Atributos:
Nombre (String)
Categoría (String): "Ropa
de Gimnasia", "Bebidas
Energéticas", etc.
Precio (Double)
cantidad\_Stock (Integer)
Descripción (String)

#### Clase

Atributos:
Tipo\_Clase (String):
"Yoga", "Pilates", etc.
fecha(Date):
Hora\_Inicio (Time)
Hora\_Fin (Time)
Instructor\_Asignado
(String)

#### Socio

Atributos:
Nombre (String)
Apellido (String)
Edad (Integer)
Condicion\_Medica (String)
Foto (../ruta del archivo)

#### <u>Pase</u>

Atributos:
Tipo (String): "Fitness",
"Spinning", "Aparatos",
"Combo".
Precio (Double)
Cantidad\_Clases (Integer)
Válido (Date)

Figure 1: claseDeDominio

#### **Note**

Note in package 'claseDeDominio'

#### Instructor

Atributos: Nombre (String) Apellido (String)

Especialización (String) : "Yoga", "Pilates", etc. Hora\_Entrada (Time) Hora\_Salida (Time)

Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023 Extends

#### Note

Note in package 'claseDeDominio'

#### **Producto**

Atributos:
Nombre (String)
Categoría (String): "Ropa de Gimnasia", "Bebidas Energéticas", etc.
Precio (Double)
cantidad\_Stock (Integer)
Descripción (String)

Note
Version 1.0 Phase 1.0 Proposed
Eduardo created on 8/10/2023. Last modified 8/10/2023
Extends

## Note

Note in package 'claseDeDominio'

#### Clase

Atributos:
Tipo\_Clase (String): "Yoga", "Pilates",etc.
fecha(Date):
Hora\_Inicio (Time)
Hora\_Fin (Time)
Instructor\_Asignado (String)

Note
Version 1.0 Phase 1.0 Proposed
Eduardo created on 8/10/2023. Last modified 8/10/2023
Extends

#### **Note**

Note in package 'claseDeDominio'

#### **Pase**

Atributos: Tipo (String): "Fitness", "Spinning", "Aparatos", "Combo". Precio (Double) Cantidad\_Clases (Integer) Válido (Date)

Note

Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023 Extends

#### Note

Note in package 'claseDeDominio'

#### Socio

Atributos: Nombre (String) Apellido (String) Edad (Integer) Condicion\_Medica (String) Foto (../ruta del archivo)

Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023 Extends

#### instructor

Object in package 'claseDeDominio'

sadas

instructor
Version 1.0 Phase 1.0 Proposed
Eduardo created on 8/10/2023. Last modified 8/10/2023

#### State1

State in package 'claseDeDominio'

Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023

## **Starter Use Case Model**

Package in package 'Model'

Starter Use Case Model
Version 1.0 Phase 1.0 Proposed
Eduardo created on 7/10/2023. Last modified 25/1/2018

## **Starter Use Case Model diagram**

Use Case diagram in package 'Starter Use Case Model'

Starter Use Case Model
Version 1.0
Eduardo created on 7/10/2023. Last modified 9/10/2023

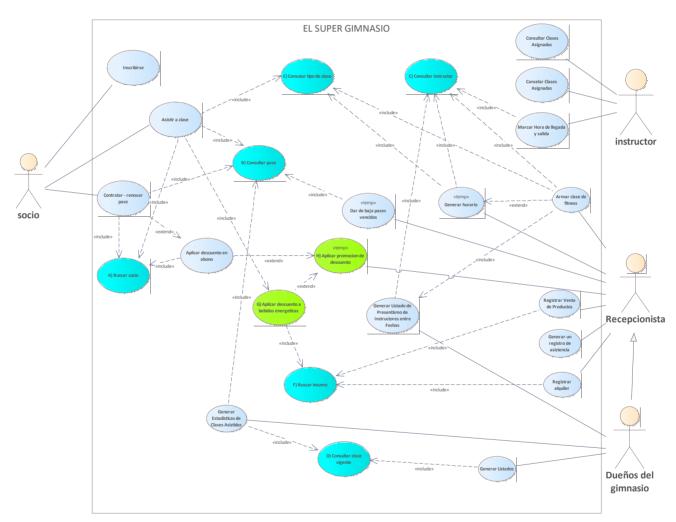


Figure 2: Starter Use Case Model

#### **EL SUPER GIMNASIO**

Boundary in package 'Starter Use Case Model'

[Description of the System]

EL SUPER GIMNASIO Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023 Extends

#### instructor

Actor in package 'Actors'

instructor Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### CONNECTORS

✓ UseCaseLink Source -> Destination From: instructor : Actor, Public

To: Marcar Hora de llegada y salida : UseCase, Public

UseCaseLink Source -> Destination

From: Consultar Clases Asignadas : UseCase, Public

To: instructor : Actor, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Cancelar Clases Asignadas Target: Public (Actor) instructor

#### socio

Actor in package 'Actors'

Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: contratar pase : UseCase, Public

✓ UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Asistir a clase : UseCase, Public

✓ UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Contratar - renovar pase : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) socio Target: Public (UseCase) Inscribirse

#### ASSOCIATIONS

## **Dueños del gimnasio**

Actor in package 'Actors'

Dueños del gimnasio Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **OUTGOING STRUCTURAL RELATIONSHIPS**

- Generalization from Dueños del gimnasio to Recepcionista

[ Direction is 'Source -> Destination'. ]

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Generar Listados Target: Public (Actor) Dueños del gimnasio

Association (direction: Unspecified)

Source: Public (UseCase) Generar Estadísticas de Clases Asistidas Target: Public (Actor) Dueños del gimnasio

Association (direction: Unspecified)

Source: Public (UseCase) Generar Listado de Presentismo de Instructores entre Fechas

Target: Public (Actor) Dueños del gimnasio

## Recepcionista

Actor in package 'Actors'

Recepcionista
Version 1.0 Phase 1.0 Proposed
Eduardo created on 7/10/2023. Last modified 7/10/2023

#### INCOMING STRUCTURAL RELATIONSHIPS

→ Generalization from Dueños del gimnasio to Recepcionista

[ Direction is 'Source -> Destination'. ]

#### **CONNECTORS**

✓ UseCaseLink Source -> Destination From: Recepcionista: Actor, Public

To: Reprogramar una clase : UseCase, Public

#### CONNECTORS

✓ UseCaseLink Source -> Destination From: Recepcionista: Actor, Public

To: Armar clase de fitness : UseCase, Public

VseCaseLink Source -> Destination
From: Generar horario : UseCase, Public
To: Recepcionista : Actor, Public

✓ UseCaseLink Source -> Destination

From: Dar de baja pases vencidos : UseCase, Public

To: Recepcionista: Actor, Public

UseCaseLink Source -> Destination

From: H) Aplicar promocion de descuento : UseCase, Public

To: Recepcionista: Actor, Public

✓ UseCaseLink Source -> Destination

From: Generar listados

: UseCase, Public

To: Recepcionista: Actor, Public

UseCaseLink Source -> Destination

From: Generar un registro de asistencia: UseCase, Public

To: Recepcionista : Actor, Public

✓ UseCaseLink Source -> Destination

From: Modificar o agregar instructores : UseCase, Public

To: Recepcionista : Actor, Public

VseCaseLink Source -> Destination
From: Registrar alquiler : UseCase, Public
To: Recepcionista : Actor, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) Recepcionista Target: Public (UseCase) Imprimir horario

Association (direction: Unspecified)

Source: Public (UseCase) Registrar Venta de Productos Target: Public (Actor) Recepcionista

## **Generar Estadísticas de Clases Asistidas**

UseCase in package 'Use Cases'

Permite generar estadísticas de las clases más asistidas de Spinning y Fitness.

Generar Estadísticas de Clases Asistidas Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 9/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

1. Ingresa al sistema.

Alternate: 1a. No se pudo acceder al sistema

- 2. Se selecciona un rango de fechas para generar las estadísticas
- 3. <include> B) Consultar pase
- 4. Se genera un listado de los tipos de pases
- 5. <include> D) Consultar clase vigente
- 6. Se muestran las clases vigentes
- 7. Se genera un listado con las asistencias totales
- 8. El sistema calcula y muestra las estadísticas de las clases más asistidas por los socios.

Alternate. No se pudo acceder al sistema

#### CONSTRAINTS

Post-condition. Se generan las estadísticas de las clases más asistidas.

[ Approved, Weight is 0. ]

Pre-condition. El dueño del gimnasio tiene permisos para generar estadísticas.

[ Approved, Weight is 0. ]

#### CONNECTORS

Include «include» Source -> Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: B) Consultar pase : UseCase, Public

**✓ Include** «include» Source → Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Generar Estadísticas de Clases Asistidas Target: Public (Actor) Dueños del gimnasio

#### **Generar Listados**

UseCase in package 'Use Cases'

Permite generar diferentes listados (clases, socios, presentismo de instructores) según los filtros seleccionados.

Generar Listados Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El sistema presenta opciones para generar listados de clases, socios o presentismo de instructores
- 2. El Dueños del gimnasio selecciona el tipo de listado que desea generar.
- 3. El sistema genera y muestra el listado seleccionado.

#### **CONSTRAINTS**

Post-condition. Se generan los listados de acuerdo a los filtros seleccionados.

[ Approved, Weight is 0. ]

Pre-condition. El dueño del gimnasio tiene permisos para generar listados.

[ Approved, Weight is 0. ]

#### CONNECTORS

Include «include» Source -> Destination
From: Generar Listados : UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Generar Listados Target: Public (Actor) Dueños del gimnasio

## Registrar alquiler

UseCase in package 'Use Cases'

Se pueden alquilar en el gimnasio canchas de Paddle. Se alquilan paletas y el trío de pelotitas de tenis. Tambien se gestiona la devolución de los mismos.

Registrar alquiler Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El sistema muestra opciones para registrar alquiler de canchas, buscar insumo o procesar devolución de paletas.
- 2. <include> F) Buscar insumo
- 3. se obtiene los insumos disponibles.

Alternate: 3a. no hay insumo disponible en stock.

#### **SCENARIOS**

4. la Recepcionista registra el alquiler de cancha o procesa la devolución de paleta

Alternate: 4a. no hay disponibilidad de horario.

- 5. el sistema registra la operación y actualiza la información.
- 6. El sistema muestra un mensaje de confirmación de la operación.

Alternate. no hay insumo disponible en stock.

Alternate. no hay disponibilidad de horario.

#### **CONSTRAINTS**

Post-condition. Se registra el alquiler de canchas o el préstamo de insumos, y se registra la devolución de paletas.

[ Approved, Weight is 0. ]

Pre-condition. El cliente desea alquilar una cancha o solicitar insumos.

[ Approved, Weight is 0. ]

#### CONNECTORS

VseCaseLink Source -> Destination
From: Registrar alquiler : UseCase, Public
To: Recepcionista : Actor, Public

From: Registrar alquiler: UseCase, Public
To: F) Buscar insumo: UseCase, Public

## Registrar Venta de Productos

UseCase in package 'Use Cases'

Permite al empleado o encargado de ventas registrar la venta de prendas de gimnasia y productos regeneradores.

Registrar Venta de Productos Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El sistema muestra un catálogo de productos regeneradores disponibles.
- 2. El usuario selecciona los productos que el cliente desea comprar y especifica la cantidad.

Alternate: 2a. El usuario cancela la venta del producto.

- 3. El sistema calcula el total y registra la venta
- 4. El sistema actualiza el inventario de productos regeneradores.
- 5. El sistema muestra un mensaje de confirmación de la venta.

Alternate. El usuario cancela la venta del producto.

#### **SCENARIOS**

#### CONSTRAINTS

Post-condition. Se registra la venta de productos.

[ Approved, Weight is 0. ]

Pre-condition. El usuario tiene permisos para registrar ventas de productos.

[ Approved, Weight is 0. ]

#### **CONNECTORS**

Include «include»

Source -> Destination

Registrar Venta de Productos : UseCase, Public

F) Buscar insumo : UseCase, Public To:

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Registrar Venta de Productos

Target: Public (Actor) Recepcionista

#### **Inscribirse**

UseCase in package 'Use Cases'

Se inscribe al gimnasio por unica vez como socio o instructor.

Se cargan los datos (datos personales, condición médica, foto de su rostro)

Inscribirse Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. una persona ingresa al sistema.
- 2. la recepcionista lo registra como socio o instructor.

Alternate: 2a. si el socio no cumple con los requisitos medicos no se registra.

Alternate: 2b. si el instructor no cumple con los requisitos no se registra.

- 3. el sistema guia a la persona a completar sus datos.
- 4. la persona confirma la inscripcion.
- 5. el sistema lo registra como socio o instructor.
- Alternate. si el instructor no cumple con los requisitos no se registra.
- Alternate. si el socio no cumple con los requisitos medicos no se registra.

## CONSTRAINTS Pre-condition. tener todos los documentos necesarios para la inscripción [ Approved, Weight is 0. ] Post-condition. en caso de ser instructor cobra sueldo. [ Approved, Weight is 1. ] Post-condition. En caso de ser socio paga la cuota.

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) socio

Target: Public (UseCase) Inscribirse

[ Approved, Weight is 1. ]

#### **Actors**

Package in package 'Starter Use Case Model'

Contains Actors that represent the roles that users play with respect to the system.

Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 7/10/2023

#### instructor

Actor in package 'Actors'

instructor Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### CONNECTORS

UseCaseLink Source -> Destination

From: instructor: Actor, Public

Marcar Hora de llegada y salida : UseCase, Public To:

UseCaseLink Source -> Destination

From: Consultar Clases Asignadas : UseCase, Public

To: instructor: Actor, Public

#### ASSOCIATIONS

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Cancelar Clases Asignadas Target: Public (Actor) instructor

#### socio

Actor in package 'Actors'

socio
Version 1.0 Phase 1.0 Proposed
Eduardo created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: contratar pase : UseCase, Public

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Asistir a clase : UseCase, Public

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Contratar - renovar pase : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) socio Target: Public (UseCase) Inscribirse

## Dueños del gimnasio

Actor in package 'Actors'

Dueños del gimnasio Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### OUTGOING STRUCTURAL RELATIONSHIPS

- Generalization from Dueños del gimnasio to Recepcionista

[ Direction is 'Source -> Destination'. ]

#### ASSOCIATIONS

Association (direction: Unspecified)

#### ASSOCIATIONS

Source: Public (UseCase) Generar Listados Target: Public (Actor) Dueños del gimnasio

Association (direction: Unspecified)

Source: Public (UseCase) Generar Estadísticas de Clases Asistidas Target: Public (Actor) Dueños del gimnasio

Association (direction: Unspecified)

Source: Public (UseCase) Generar Listado de Presentismo de

Instructores entre Fechas

Target: Public (Actor) Dueños del gimnasio

## Recepcionista

Actor in package 'Actors'

Recepcionista
Version 1.0 Phase 1.0 Proposed
Eduardo created on 7/10/2023. Last modified 7/10/2023

#### INCOMING STRUCTURAL RELATIONSHIPS

→ Generalization from Dueños del gimnasio to Recepcionista

[ Direction is 'Source -> Destination'. ]

## CONNECTORS

✓ UseCaseLink Source -> Destination From: Recepcionista : Actor, Public

To: Reprogramar una clase : UseCase, Public

✓ UseCaseLink Source -> Destination From: Recepcionista: Actor, Public

To: Armar clase de fitness : UseCase, Public

VseCaseLink Source -> Destination
From: Generar horario : UseCase, Public
To: Recepcionista : Actor, Public

UseCaseLink Source -> Destination

From: Dar de baja pases vencidos : UseCase, Public

To: Recepcionista : Actor, Public

✓ UseCaseLink Source -> Destination

From: H) Aplicar promocion de descuento : UseCase, Public

To: Recepcionista: Actor, Public

#### CONNECTORS

UseCaseLink Source -> Destination

From: Generar listados

: UseCase, Public

To: Recepcionista: Actor, Public

UseCaseLink Source -> Destination

From: Generar un registro de asistencia: UseCase, Public

To: Recepcionista : Actor, Public

✓ UseCaseLink Source -> Destination

From: Modificar o agregar instructores : UseCase, Public

To: Recepcionista: Actor, Public

VseCaseLink Source -> Destination
From: Registrar alquiler : UseCase, Public
To: Recepcionista : Actor, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) Recepcionista Target: Public (UseCase) Imprimir horario

Association (direction: Unspecified)

Source: Public (UseCase) Registrar Venta de Productos Target: Public (Actor) Recepcionista

#### **Use Cases**

Package in package 'Starter Use Case Model'

Contains Use Case that represent the value or goal that the Actors wish to achieve.

Use Cases
Version 1.0 Phase 1.0 Proposed
Eduardo created on 7/10/2023. Last modified 7/10/2023

## Generar Estadísticas de Clases Asistidas

UseCase in package 'Use Cases'

Permite generar estadísticas de las clases más asistidas de Spinning y Fitness.

Generar Estadísticas de Clases Asistidas Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 9/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

1. Ingresa al sistema.

Alternate: 1a. No se pudo acceder al sistema

- 2. Se selecciona un rango de fechas para generar las estadísticas
- 3. <include> B) Consultar pase
- 4. Se genera un listado de los tipos de pases
- 5. <include> D) Consultar clase vigente
- 6. Se muestran las clases vigentes
- 7. Se genera un listado con las asistencias totales
- 8. El sistema calcula y muestra las estadísticas de las clases más asistidas por los socios.

Alternate. No se pudo acceder al sistema

#### **CONSTRAINTS**

Post-condition. Se generan las estadísticas de las clases más asistidas.

[ Approved, Weight is 0. ]

Pre-condition. El dueño del gimnasio tiene permisos para generar estadísticas.

[ Approved, Weight is 0. ]

#### **CONNECTORS**

**Include** «include» Source → Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: B) Consultar pase : UseCase, Public

**✓ Include** «include» Source → Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Generar Estadísticas de Clases Asistidas

Target: Public (Actor) Dueños del gimnasio

## **Generar Listados**

UseCase in package 'Use Cases'

Permite generar diferentes listados (clases, socios, presentismo de instructores) según los filtros seleccionados.

Generar Listados Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El sistema presenta opciones para generar listados de clases, socios o presentismo de instructores
- 2. El Dueños del gimnasio selecciona el tipo de listado que desea generar.
- 3. El sistema genera y muestra el listado seleccionado.

#### **CONSTRAINTS**

Post-condition. Se generan los listados de acuerdo a los filtros seleccionados.

[ Approved, Weight is 0. ]

Pre-condition. El dueño del gimnasio tiene permisos para generar listados.

[ Approved, Weight is 0. ]

#### CONNECTORS

Include «include» Source -> Destination From: Generar Listados: UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

#### ASSOCIATIONS



Association (direction: Unspecified)

Source: Public (UseCase) Generar Listados

Target: Public (Actor) Dueños del gimnasio

## Registrar alquiler

UseCase in package 'Use Cases'

Se pueden alquilar en el gimnasio canchas de Paddle. Se alquilan paletas y el trío de pelotitas de tenis.

Tambien se gestiona la devolución de los mismos.

Registrar alquiler Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El sistema muestra opciones para registrar alquiler de canchas, buscar insumo o procesar devolución de paletas.
- 2. <include> F) Buscar insumo
- 3. se obtiene los insumos disponibles.

Alternate: 3a. no hay insumo disponible en stock.

4. la Recepcionista registra el alquiler de cancha o procesa la devolución de paleta

Alternate: 4a. no hay disponibilidad de horario.

- 5. el sistema registra la operación y actualiza la información.
- 6. El sistema muestra un mensaje de confirmación de la operación.
- Alternate. no hay insumo disponible en stock.
- Alternate. no hay disponibilidad de horario.

#### **CONSTRAINTS**

Post-condition. Se registra el alquiler de canchas o el préstamo de insumos, y se registra la devolución de paletas.

[ Approved, Weight is 0. ]

Pre-condition. El cliente desea alquilar una cancha o solicitar insumos.

[ Approved, Weight is 0. ]

#### **CONNECTORS**

VseCaseLink Source -> Destination
From: Registrar alquiler : UseCase, Public
To: Recepcionista : Actor, Public

From: Registrar alquiler: UseCase, Public
To: F) Buscar insumo: UseCase, Public

## Registrar Venta de Productos

UseCase in package 'Use Cases'

Permite al empleado o encargado de ventas registrar la venta de prendas de gimnasia y productos regeneradores.

Registrar Venta de Productos Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

1. El sistema muestra un catálogo de productos regeneradores disponibles.

2. El usuario selecciona los productos que el cliente desea comprar y especifica la cantidad.

Alternate: 2a. El usuario cancela la venta del producto.

- 3. El sistema calcula el total y registra la venta
- 4. El sistema actualiza el inventario de productos regeneradores.
- 5. El sistema muestra un mensaje de confirmación de la venta.

Alternate. El usuario cancela la venta del producto.

#### CONSTRAINTS

Post-condition. Se registra la venta de productos.

[ Approved, Weight is 0. ]

Pre-condition. El usuario tiene permisos para registrar ventas de productos.

[ Approved, Weight is 0. ]

#### CONNECTORS

**Include** «include» Source → Destination

From: Registrar Venta de Productos : UseCase, Public

To: F) Buscar insumo : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Registrar Venta de Productos Target: Public (Actor) Recepcionista

#### **Inscribirse**

UseCase in package 'Use Cases'

Se inscribe al gimnasio por unica vez como socio o instructor.

Se cargan los datos (datos personales, condición médica, foto de su rostro)

Inscribirse Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. una persona ingresa al sistema.
- 2. la recepcionista lo registra como socio o instructor.

Alternate: 2a. si el socio no cumple con los requisitos medicos no se registra.

Alternate: 2b. si el instructor no cumple con los requisitos no se registra.

#### **SCENARIOS**

- 3. el sistema guia a la persona a completar sus datos.
- 4. la persona confirma la inscripcion.
- 5. el sistema lo registra como socio o instructor.

Alternate. si el instructor no cumple con los requisitos no se registra.

Alternate. si el socio no cumple con los requisitos medicos no se registra.

#### CONSTRAINTS

Pre-condition. tener todos los documentos necesarios para la inscripción

[ Approved, Weight is 0. ]

Post-condition. en caso de ser instructor cobra sueldo.

[ Approved, Weight is 1. ]

Post-condition. En caso de ser socio paga la cuota.

[ Approved, Weight is 1. ]

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) socio

Target: Public (UseCase) Inscribirse

## **Imprimir horario**

UseCase in package 'Use Cases'

Imprime un horario actualizado con los horario de spinning y tambien los dias en que se dictan las clases, salón y qué instructor esta a clargo en cada clase de fitness.

Puede reprogramar las clases

Imprimir horario Version 1.0 Phase 1.0 Proposed Eduardo created on 7/10/2023. Last modified 8/10/2023

#### CONSTRAINTS

Pre-condition. se tiene que haber generado un horario.

[ Approved, Weight is 0. ]

Post-condition. se imprime el horario.

[ Approved, Weight is 1. ]

#### **CONNECTORS**

#### **CONNECTORS**

From: Imprimir horario : UseCase, Public
To: Generar horario : UseCase, Public

#### ASSOCIATIONS

/ Association (direction: Unspecified)

Source: Public (Actor) Recepcionista Target: Public (UseCase) Imprimir horario

## **Consultar Clases Asignadas**

UseCase in package 'Starter Use Case Model'

El instructor consulta el horario de las clases que tiene asignadas.

Consultar Clases Asignadas Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### CONSTRAINTS

Post-condition. El instructor tiene acceso a su horario de clases.

[ Approved, Weight is 0. ]

Pre-condition. El instructor ha iniciado sesión en el sistema.

[ Approved, Weight is 0. ]

#### CONNECTORS

✓ UseCaseLink Source -> Destination

From: Consultar Clases Asignadas : UseCase, Public

To: instructor : Actor, Public

## A) Buscar socio

UseCase in package 'Starter Use Case Model'

A) Buscar socio Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

**✓ Include** «include» Source → Destination

From: Aplicar descuento en abono : UseCase, Public

To: A) Buscar socio : UseCase, Public

#### CONNECTORS

Include «include» Source -> Destination
From: Contratar - renovar pase : UseCase, Public

To: A) Buscar socio : UseCase, Public

From: Asistir a clase: UseCase, Public
To: A) Buscar socio: UseCase, Public

## Aplicar descuento en abono

UseCase in package 'Starter Use Case Model'

Aplicar descuento en abono Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### CONNECTORS

**Extend** «extend» Source -> Destination

From: Aplicar descuento en abono : UseCase, Public To: H) Aplicar promocion de descuento : UseCase, Public

Include «include» Source -> Destination

From: Aplicar descuento en abono : UseCase, Public

To: A) Buscar socio : UseCase, Public

From: Contratar - renovar pase : UseCase, Public
To: Aplicar descuento en abono : UseCase, Public

## Armar clase de fitness

UseCase in package 'Starter Use Case Model'

Armar clase de fitness Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 9/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. la Recepcionista ingresa al sistema.
- 2. el sistema muestra las opciones para seleccionar el tipo de clase, instructor y horario.
- 3. <include> E) Consutar tipo de clase
- 4. selecciona la opción de instructores de fitness
- 5. <include> C) Consultar instructor
- 6. <include> Generar Listado de Presentismo de Instructores entre Fechas
- 7. la recepcionista completa los detallles en una planilla.
- 8. Confirma la creación de clase con el instructor.

#### **SCENARIOS**

Alternate: 8a. la recepcionista cancela la creación de la clase

9. el sistema registra la clase con el instructor y su horario.

10. <extend> Generar horario

Alternate. la recepcionista cancela la creación de la clase

#### **CONSTRAINTS**

Post-condition. se genera la clase con el instructor indicado.

[ Approved, Weight is 0. ]

Pre-condition. la recepcionista arma la clase.

[ Approved, Weight is 0. ]

#### CONNECTORS

From: Armar clase de fitness : UseCase, Public
To: Generar horario : UseCase, Public

Include «include» Source -> Destination
From: Armar clase de fitness: UseCase, Public

To: Generar Listado de Presentismo de Instructores entre Fechas : UseCase, Public

From: Armar clase de fitness: UseCase, Public
To: E) Consutar tipo de clase: UseCase, Public

Include «include» Source -> Destination
From: Armar clase de fitness : UseCase, Public
To: C) Consultar instructor : UseCase, Public

Include «include» Source -> Destination
From: Armar clase de fitness : UseCase, Public
To: Generar Listados : UseCase, Public

From: Generar horario : UseCase, Public
To: Armar clase de fitness : UseCase, Public

✓ UseCaseLink Source -> Destination From: Recepcionista : Actor, Public

To: Armar clase de fitness : UseCase, Public

#### Asistir a clase

UseCase in package 'Starter Use Case Model'

Asistir a clase
Version 1.0 Phase 1.0 Proposed
Valentin created on 7/10/2023. Last modified 9/10/2023

#### **SCENARIOS**

Basic Path. Basic Path

- 1. El socio ingresa al sistema.
- 2. <include> A) Buscar socio
- 3. se obtiene los datos del socio.

Alternate: 3a. La persona que ingresó no está registrado en el sistema

- 4. Se muestran los datos del socio
- 5. <include> B) Consultar pase
- 6. Se muestran la cantidad de entradas disponibles
- 7. <include> E) Consutar tipo de clase
- 8. Ingresa a la clase

Alternate: 8a. No tiene clases disponibles

- 9. <include> G) Aplicar descuento a bebidas energeticas
- 10. Compra bebidas

Alternate: 10a. No accede al descuento

- Alternate. La persona que ingresó no está registrado en el sistema
- Alternate. No tiene clases disponibles
- Alternate. No accede al descuento

#### CONSTRAINTS

Post-condition. Acceder a la clase

[ Approved, Weight is 0. ]

Pre-condition. No debe tener el pase vencido

[ Approved, Weight is 0. ]

The Pre-condition. Debe tener entradas disponibles

[ Approved, Weight is 0. ]

Pre-condition. Debe haber realizado el pago

[ Approved, Weight is 0. ]

#### CONNECTORS

From: Asistir a clase: UseCase, Public
To: B) Consultar pase: UseCase, Public

#### CONNECTORS

Include «include» Source -> Destination From: Asistir a clase : UseCase, Public

To: E) Consutar tipo de clase : UseCase, Public

Include «include» Source -> Destination
From: Asistir a clase: UseCase, Public

To: G) Aplicar descuento a bebidas energeticas : UseCase, Public

From: Asistir a clase: UseCase, Public
To: A) Buscar socio: UseCase, Public

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Asistir a clase : UseCase, Public

## B) Consultar pase

UseCase in package 'Starter Use Case Model'

B) Consultar pase Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

From: Asistir a clase: UseCase, Public
To: B) Consultar pase: UseCase, Public

From: contratar pase : UseCase, Public
To: B) Consultar pase : UseCase, Public

✓ Include «include» Source -> Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: B) Consultar pase : UseCase, Public

**Include** «include» Source → Destination

From: Dar de baja pases vencidos : UseCase, Public

To: B) Consultar pase: UseCase, Public

From: Contratar - renovar pase : UseCase, Public
To: B) Consultar pase : UseCase, Public

## C) Consultar instructor

UseCase in package 'Starter Use Case Model'

C) Consultar instructor Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

**✓ Include** «include» Source → Destination

From: Marcar Hora de llegada y salida : UseCase, Public To: C) Consultar instructor : UseCase, Public

From: Armar clase de fitness: UseCase, Public
To: C) Consultar instructor: UseCase, Public

From: Generar horario : UseCase, Public

To: C) Consultar instructor : UseCase, Public

**✓ Include** «include» Source → Destination

From: Generar Listado de Presentismo de Instructores entre Fechas : UseCase, Public

To: C) Consultar instructor: UseCase, Public

## **Cancelar Clases Asignadas**

UseCase in package 'Starter Use Case Model'

El instructor puede seleccionar una o varias clases asignadas y cancelarlas.

Cancelar Clases Asignadas Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### **CONSTRAINTS**

Post-condition. El instructor tiene clases asignadas.

[ Approved, Weight is 0. ]

Pre-condition. El instructor ha iniciado sesión en el sistema.

[ Approved, Weight is 0. ]

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Cancelar Clases Asignadas Target: Public (Actor) instructor

## Contratar - renovar pase

UseCase in package 'Starter Use Case Model'

Contratar - renovar pase Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

#### **SCENARIOS**

Basic Path. Contratar pase- inscripción única

- 1. Quiere contratar un pase mensual
- 2. <include> A) Buscar socio

Alternate: 2a. El socio ya se encuentra registrado en el sistema

- 3. El socio no esta registrado en el sistema
- 4. Pide los datos de inscripcion
- 5. debe entregar una foto y todos los datos correspondientes junto con el apto médico
- 6. se consultan los tipos de pases disponibles
- 7. se obtienen los precios de los pases
- 8. elige el pase que más se acomode a su gusto
- 9. Elige el método con que pagar
- 10. queda el socio registrado en el sistema
- 11. Se genera la cantidad de entradas (25)

Alternate. El socio ya se encuentra registrado en el sistema

- 1. Se obtiene la cantidad de meses que es socio
- 2. decide renovar el pase
- 3. Cada 3 meses se aplica un descuento
- 4. <extend> Aplicar descuento en abono

#### **CONSTRAINTS**

Post-condition. Pagar

[ Approved, Weight is 0. ]

Pre-condition. Estar inscripto

[ Approved, Weight is 0. ]

#### CONNECTORS

From: Contratar - renovar pase : UseCase, Public

To: Aplicar descuento en abono : UseCase, Public

From: Contratar - renovar pase : UseCase, Public
To: A) Buscar socio : UseCase, Public

Include «include» Source -> Destination
From: Contratar - renovar pase : UseCase, Public
To: B) Consultar pase : UseCase, Public

#### **CONNECTORS**

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: Contratar - renovar pase : UseCase, Public

#### contratar pase

UseCase in package 'Starter Use Case Model'

Tipos de pases disponibles: 25 clases desde 6000\$ Fitness 5500\$ solo Spinning 5000\$ solo aparatos

7000\$ el combo Aparatos + Spinning o Spinnig + Localizada.

contratar pase
Version 1.0 Phase 1.0 Proposed
Valentin created on 7/10/2023. Last modified 7/10/2023

#### **CONSTRAINTS**

Post-condition. Generar la cantidad de pases

[ Approved, Weight is 0. ]

Pre-condition. Estar inscripto

[ Approved, Weight is 0. ]

#### **CONNECTORS**

From: contratar pase : UseCase, Public
To: B) Consultar pase : UseCase, Public

UseCaseLink Source -> Destination

From: socio: Actor, Public

To: contratar pase : UseCase, Public

## D) Consultar clase vigente

UseCase in package 'Starter Use Case Model'

D) Consultar clase vigente Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### **CONNECTORS**

#### CONNECTORS

Include «include» Source -> Destination

From: Generar Estadísticas de Clases Asistidas : UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

Include «include» Source -> Destination From: Generar Listados : UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

**Include** «include» Source -> Destination

From: Generar listados

: UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

## Dar de baja pases vencidos

UseCase «temp» in package 'Starter Use Case Model'

Diariamente se dispara el proceso de limpieza que se encarga de dar de baja los pases vencidos

Dar de baja pases vencidos Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

#### CONSTRAINTS

Pre-condition. tienen que estar vencidos los pases.

[ Approved, Weight is 0. ]

Post-condition. se dan de baja.

[ Approved, Weight is 1. ]

#### CONNECTORS

UseCaseLink Source -> Destination

From: Dar de baja pases vencidos : UseCase, Public

To: Recepcionista: Actor, Public

Include «include» Source -> Destination

From: Dar de baja pases vencidos : UseCase, Public

To: B) Consultar pase: UseCase, Public

## E) Consutar tipo de clase

UseCase in package 'Starter Use Case Model'

#### **CONNECTORS**

From: Armar clase de fitness: UseCase, Public
To: E) Consutar tipo de clase: UseCase, Public

✓ Include «include» Source -> Destination From: Generar horario: UseCase, Public

To: E) Consutar tipo de clase : UseCase, Public

Include «include» Source -> Destination
From: Asistir a clase: UseCase, Public

To: E) Consutar tipo de clase : UseCase, Public

## F) Buscar insumo

UseCase in package 'Starter Use Case Model'

F) Buscar insumo Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### **CONNECTORS**

**✓ Include** «include» Source → Destination

From: G) Aplicar descuento a bebidas energeticas : UseCase, Public

To: F) Buscar insumo: UseCase, Public

**✓ Include** «include» Source → Destination

From: Registrar Venta de Productos : UseCase, Public

To: F) Buscar insumo : UseCase, Public

From: Registrar alquiler: UseCase, Public
To: F) Buscar insumo: UseCase, Public

## G) Aplicar descuento a bebidas energeticas

UseCase in package 'Starter Use Case Model'

Se venden productos regeneradores de sales minerales y proteínas tales como Powerade, Gatorade, Agua mineral y saborizadas,café, y barras de cereal a distintos precios. A veces, cuando una persona asiste a una clase, y es la tercera del día, el sistema ofrece promoción 2x1 en aguas o energizantes.

G) Aplicar descuento a bebidas energeticas Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

#### **CONNECTORS**

#### CONNECTORS

**Include** «include» Source -> Destination

From: G) Aplicar descuento a bebidas energeticas: UseCase, Public

F) Buscar insumo : UseCase, Public To:

**Extend** «extend» Source -> Destination

G) Aplicar descuento a bebidas energeticas: UseCase, Public H) Aplicar promocion de descuento: UseCase, Public

**Include** «include» Source -> Destination From: Asistir a clase: UseCase, Public

To: G) Aplicar descuento a bebidas energeticas: UseCase, Public

#### **Generar horario**

UseCase «temp» in package 'Starter Use Case Model'

Se genera un horario actualizado todos los meses y se borra el anterior Se pueden agregar nuevos instructores o modificarlos, como tambien agregar clases, cancelarlas o modificarlas. Imprime los horarios

> Generar horario Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 9/10/2023

#### CONSTRAINTS

Post-condition. Imprime el horario

[ Approved, Weight is 0. ]

Pre-condition. se selecciona al instructor y la clase.

[ Approved, Weight is 0. ]

Post-condition. se genera el horario.

[ Approved, Weight is 1. ]

#### CONNECTORS

Z Extend «extend»

Source -> Destination

Generar horario: UseCase, Public From: To: Armar clase de fitness : UseCase, Public

UseCaseLink Source -> Destination From: Generar horario: UseCase, Public To: Recepcionista: Actor, Public

#### CONNECTORS

Include «include» Source -> Destination From: Generar horario : UseCase, Public

To: E) Consutar tipo de clase : UseCase, Public

Include «include» Source -> Destination
 From: Generar horario: UseCase, Public
 To: C) Consultar instructor: UseCase, Public

From: Armar clase de fitness: UseCase, Public
To: Generar horario: UseCase, Public

From: Imprimir horario : UseCase, Public
To: Generar horario : UseCase, Public

#### Generar Listado de Presentismo de Instructores entre Fechas

UseCase in package 'Starter Use Case Model'

Generar Listado de Presentismo de Instructores entre Fechas Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023

#### **CONSTRAINTS**

Post-condition. Se genera un informe que muestra la asistencia de los instructores.

[ Approved, Weight is 0. ]

Pre-condition. Debe haber registros de asistencia de instructores en el sistema.

[ Approved, Weight is 0. ]

#### **CONNECTORS**

**Include** «include» Source → Destination

From: Generar Listado de Presentismo de Instructores entre Fechas : UseCase, Public

To: C) Consultar instructor : UseCase, Public

Include «include» Source -> Destination
From: Armar clase de fitness: UseCase, Public

To: Generar Listado de Presentismo de Instructores entre Fechas : UseCase, Public

#### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (UseCase) Generar Listado de Presentismo de Target: Public (Actor) Dueños del gimnasio

Instructores entre Fechas

#### ASSOCIATIONS

#### **Generar Listados**

UseCase in package 'Starter Use Case Model'

Generar Listados Version 1.0 Phase 1.0 Proposed Eduardo created on 8/10/2023. Last modified 8/10/2023

#### CONNECTORS

From: Armar clase de fitness: UseCase, Public
To: Generar Listados: UseCase, Public

#### **Generar listados**

UseCase in package 'Starter Use Case Model'

Permite generar diferentes listados (clases, socios, presentismo de instructores) según los filtros seleccionados.

Generar listados

Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### CONNECTORS

UseCaseLink Source -> Destination

From: Generar listados

: UseCase, Public

To: Recepcionista: Actor, Public

✓ Include «include» Source -> Destination

From: Generar listados

: UseCase, Public

To: D) Consultar clase vigente : UseCase, Public

## Generar un registro de asistencia

UseCase in package 'Starter Use Case Model'

Generar un registro de asistencia Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

#### **CONSTRAINTS**

#### CONSTRAINTS

Pre-condition. se registra la asistencia en el sistema cuando el socio ingresa a una clase.

[ Approved, Weight is 0. ]

Post-condition. se genera una lista de asistencia.

[ Approved, Weight is 1. ]

#### CONNECTORS

UseCaseLink

Source -> Destination

Generar un registro de asistencia : UseCase, Public From:

Recepcionista: Actor, Public To:

## H) Aplicar promocion de descuento

UseCase «temp» in package 'Starter Use Case Model'

se verifican los socios frecuentes, y cada 3 meses consecutivos, el sistema puede premiarlos arrojando un descuento que pueden ir entre un 10 al 50% en el abono de la cuota

> H) Aplicar promocion de descuento Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

#### CONSTRAINTS

Post-condition. se aplica el descuento.

[ Approved, Weight is 0. ]

Tre-condition. que sea socio frecuente

[ Approved, Weight is 0. ]

#### CONNECTORS

UseCaseLink Source -> Destination

H) Aplicar promocion de descuento : UseCase, Public From:

Recepcionista: Actor, Public To:

**Extend** «extend» Source -> Destination

Aplicar descuento en abono : UseCase, Public From:

To: H) Aplicar promocion de descuento : UseCase, Public

**Extend** «extend» Source -> Destination

G) Aplicar descuento a bebidas energeticas: UseCase, Public From:

H) Aplicar promocion de descuento : UseCase, Public To:

## Marcar Hora de llegada y salida

UseCase in package 'Starter Use Case Model'

El instructor utiliza su tarjeta magnética especial para marcar su hora de llegada o salida al gimnasio.

Marcar Hora de llegada y salida Version 1.0 Phase 1.0 Proposed Ro created on 8/10/2023. Last modified 8/10/2023

#### **CONSTRAINTS**

Pre-condition. Ingresa al sistema para marcar la hora.

[ Approved, Weight is 0. ]

Post-condition. El sistema almacena la hora de llegada o salida del instructor.

[ Approved, Weight is 1. ]

#### CONNECTORS

**✓ Include** «include» Source → Destination

From: Marcar Hora de llegada y salida : UseCase, Public To: C) Consultar instructor : UseCase, Public

**✓ UseCaseLink** Source → Destination

From: instructor : Actor, Public

To: Marcar Hora de llegada y salida : UseCase, Public

## Modificar o agregar instructores

UseCase in package 'Starter Use Case Model'

Modificar o agregar instructores Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### **CONNECTORS**

UseCaseLink Source -> Destination

From: Modificar o agregar instructores : UseCase, Public

To: Recepcionista : Actor, Public

## Registrar venta de insumos

UseCase in package 'Starter Use Case Model'

Registrar venta de insumos Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 8/10/2023

## Reprogramar una clase

UseCase in package 'Starter Use Case Model'

Estos cambios se ven reflejados en el horario impreso

Reprogramar una clase Version 1.0 Phase 1.0 Proposed Valentin created on 7/10/2023. Last modified 7/10/2023

#### CONNECTORS

✓ UseCaseLink Source -> Destination From: Recepcionista : Actor, Public

To: Reprogramar una clase : UseCase, Public