

Programación 4

Informe del Modelo de Casos de Uso

Grupo 66

Integrantes

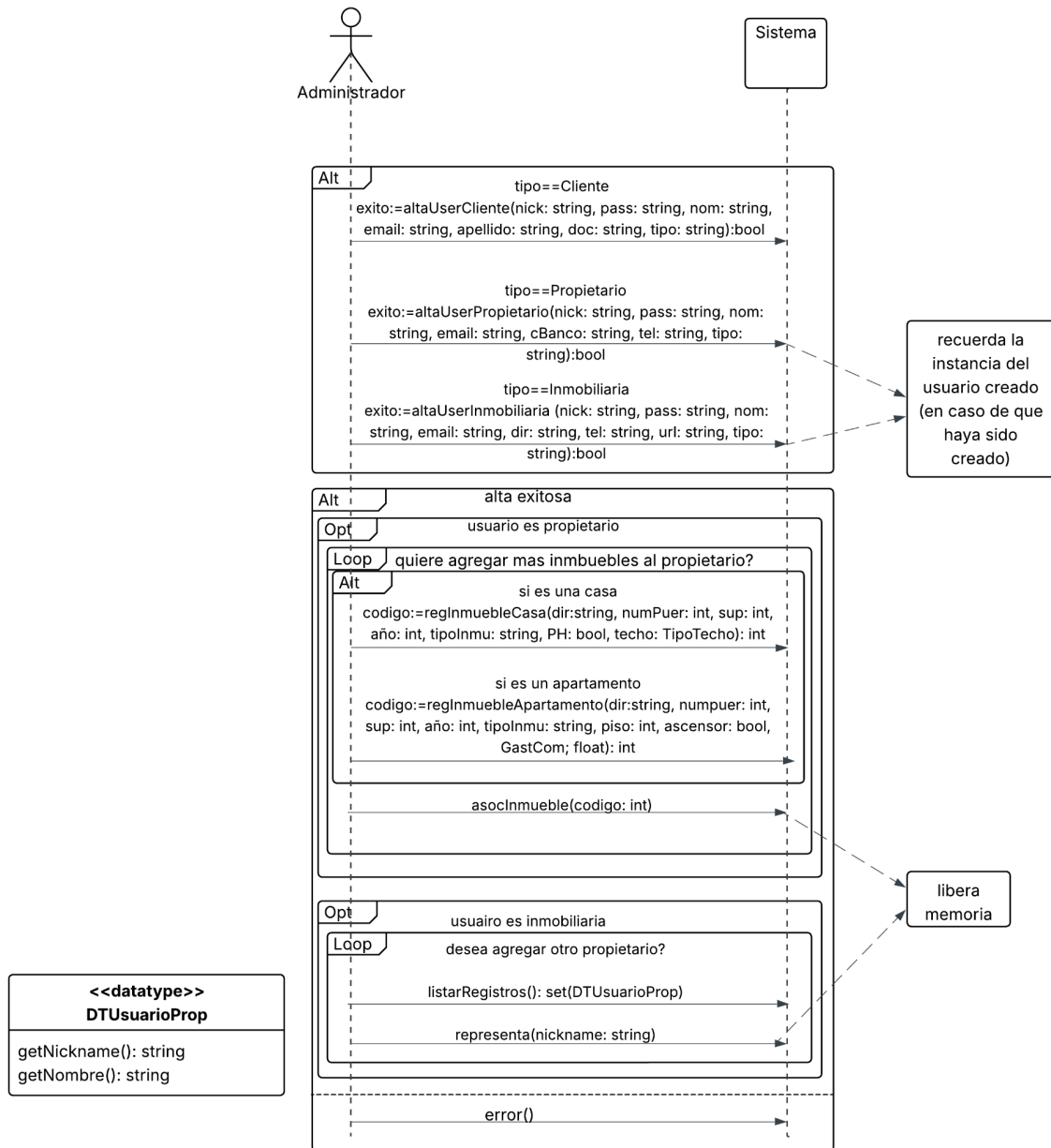
Valentin Pedragosa
Matias Torres
Tomás Martínez
Emiliano Esponda
Camilo Morganti

CI: 5.532.445-1
CI: 5.485.366-5
CI: 5.535.431-9
CI: 5.491.971-6
CI: 5.443.356-4

Docente: Federico Andrade

Alta de usuario con memoria

Diagramas de Secuencia del Sistema



Contratos

Nombre	Alta de cliente
Operación	<code>exito:=altaUserCliente(nick:string, pass:string, nom:string, email:string, apellido:string, doc:string, tipo:string):bool</code>
Entrada	nick: nickname del nuevo usuario. pass: contraseña del nuevo usuario. nom: nombre del nuevo usuario. email: email del nuevo usuario. apellido: apellido del nuevo usuario. doc: número de documento del nuevo usuario. tipo: tipo de usuario.
Salida	devuelve true si creo exitosamente, false en caso de que no
Descripción	Crea una instancia de cliente, carga sus datos correspondientes y lo da de alta si no existe otro usuario con el mismo nickname; en caso contrario devuelve false..

Precondiciones y Postcondiciones
Pre: pass debe contener al menos 6 caracteres. Pre: tipo debe ser cliente. Post: sí es creada, Existe una nueva instancia de cliente

Nombre	Alta de propietario
Operación	<code>altaUserPropietario(nick:string, pass:string, nom:string, email:string, cBanco:string, tel:string, tipo:string)</code>
Entrada	nick: nickname del nuevo usuario. pass: contraseña del nuevo usuario. nom: nombre del nuevo usuario. email: email del nuevo usuario. cBanco: cuenta bancaria del nuevo usuario. tel: teléfono de contacto del nuevo usuario. tipo: tipo de usuario.
Salida	Devuelve true si creo exitosamente, false en caso de que no
Descripción	Crea una instancia de propietario, carga sus datos correspondientes y, si no hay otra instancia propietario con el mismo nickname, es dado de alta y añadido al registro de propietarios; en caso contrario devuelve false..

Precondiciones y Postcondiciones
<p>Pre: pass debe contener al menos 6 caracteres.</p> <p>Pre: tipo debe ser propietario.</p> <p>Pre: tel debe ser un string de números.</p> <p>Post: sí es creada, Existe una nueva instancia de propietario</p> <p>Post: si la instancia es creada, se une al registro de propietarios.</p> <p>Post: sí es creada, se recuerda la nueva instancia de usuario.</p>

Nombre	Alta de inmobiliaria
Operación	altaUserInmobiliaria(nick:string, pass:string, nom:string, email:string, dir:string, tel:string, URL:string, tipo:string)
Entrada	<p>nick: nickname del nuevo usuario.</p> <p>pass: contraseña del nuevo usuario.</p> <p>nom: nombre del nuevo usuario.</p> <p>email: email del nuevo usuario.</p> <p>dir: dirección de la inmobiliaria.</p> <p>tel: teléfono de la inmobiliaria.</p> <p>URL: sitio web de la inmobiliaria.</p> <p>tipo: tipo de usuario.</p>
Salida	Devuelve true si creo exitosamente, false en caso de que no
Descripción	Crea una instancia de inmobiliaria, carga sus datos correspondientes y, si no hay otra instancia inmobiliaria con el mismo nickname, el usuario es dado de alta; en caso contrario devuelve false

Precondiciones y Postcondiciones
<p>Pre: pass debe contener al menos 6 caracteres.</p> <p>Pre: tipo debe ser inmobiliaria.</p> <p>Pre: tel debe ser un string de números.</p> <p>Post: sí es creada, Existe una nueva instancia de inmobiliaria</p> <p>Post: sí es creada, se recuerda la nueva instancia de usuario.</p>

Nombre	Registro de Inmuebles casa
Operación	<code>regInmuebleCasa(dir:string, numPuer:int, sup:int, año:int, tipoInmu:string, PH:bool, techo:TipoTecho):int</code>
Entrada	<p>dir: dirección del inmueble.</p> <p>numPuer: número de puerta del inmueble.</p> <p>sup: superficie del inmueble.</p> <p>año: año de construcción del inmueble.</p> <p>tipoInmu: tipo de inmueble.</p> <p>PH: si es propiedad horizontal.</p> <p>techo: tipo de techo.</p>
Salida	Código del inmueble.
Descripción	Crea una instancia inmueble, carga sus datos y la da de alta. Genera y devuelve un código numérico para el inmueble.

Precondiciones y Postcondiciones
<p>Pre: tipoInmu debe ser casa.</p> <p>Post: se crea una nueva instancia de inmueble.</p> <p>Post: el código generado debe ser único para cada instancia inmueble creada.</p>

Nombre	Registro de Inmuebles apartamento
Operación	<code>regInmuebleApartamento(dir:string, numPuer:int, sup:int, año:int, tipoInmu:string, piso:int, ascensor:bool, gc:float):int</code>
Entrada	<p>dir: dirección del inmueble.</p> <p>numPuer: número de puerta del inmueble.</p> <p>sup: superficie del inmueble.</p> <p>año: año de construcción del inmueble.</p> <p>tipoInmu: tipo de inmueble.</p> <p>piso: número de piso del apartamento.</p> <p>ascensor: el edificio posee ascensor.</p>

	gc: monto de gastos comunes.
Salida	Código del inmueble.
Descripción	Crea una instancia inmueble, carga sus datos y la da de alta. Genera y devuelve un código numérico para el inmueble.

Precondiciones y Postcondiciones	
Pre: tipoInmu debe ser apartamento.	
Post: se crea una nueva instancia de inmueble.	
Post: el código generado debe ser único para cada instancia inmueble creada.	

Nombre	Asociación de inmuebles
Operación	asoccInmueble(codigo:int)
Entrada	Recibe el código del inmueble.
Salida	-
Descripción	Asocia, mediante el código, al inmueble con su propietario..

Precondiciones y Postcondiciones	
Pre: el código debe estar asignado a un inmueble.	
Pre: existe en el sistema una instancia de propietario recordada.	
Post: existe un link entre el inmueble y el propietario.	
Post: libera memoria.	

Nombre	Lista de Propietarios
Operación	listarRegistros():set(DTUsuarioProp)
Entrada	-
Salida	Devuelve un set con los propietarios registrados.
Descripción	Muestra el nickname y nombre de los propietarios registrados.

Precondiciones y Postcondiciones	
Pre: hay al menos un propietario registrado.	

Nombre	Inmobiliaria representa propietario
Operación	representa(nickname:string)
Entrada	Recibe el nickname del propietario que la inmobiliaria representa.

Salida	-
Descripción	Asocia a la inmobiliaria con su representado..

Precondiciones y Postcondiciones

Pre: el nickname debe estar asignado a un propietario registrado.

Pre: existe en el sistema una instancia de inmobiliaria recordada.

Post: se crea una nueva asociación entre la inmobiliaria y el propietario.

Post: libera memoria.

Nombre	Error
Operación	<code>error()</code>
Entrada	No aplica
Salida	No aplica
Descripción	Informa que el sistema ha tenido un error en la alta de usuario

Precondiciones y Postcondiciones

Pre: el nickname debe estar asignado a un propietario registrado.

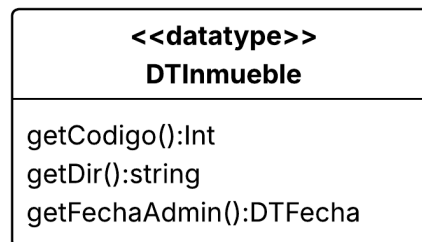
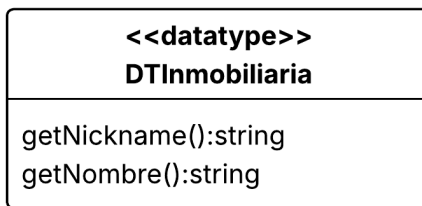
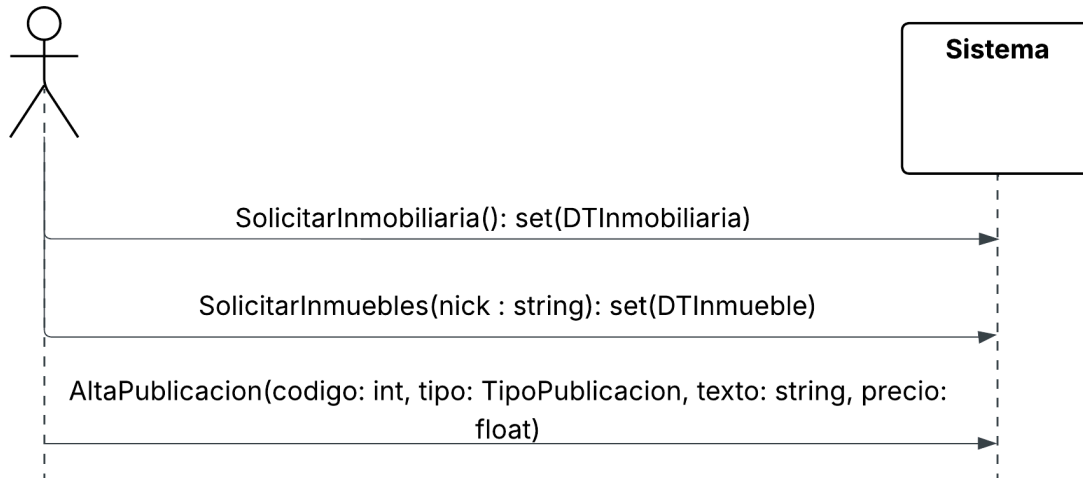
Pre: existe en el sistema una instancia de inmobiliaria recordada.

Post: se crea una nueva asociación entre la inmobiliaria y el propietario.

Post: libera memoria.

Alta de Publicación

Diagramas de Secuencia del Sistema



Contratos

Nombre	SolicitarInmobiliaria
Operación	<code>solicitarInmobiliaria():set(DTInmobiliaria)</code>
Entrada	No aplica.
Salida	Set con todas las inmobiliarias.

Descripción	Devuelve un set con todas las inmobiliarias
--------------------	---

Precondiciones y Postcondiciones
Pre: Existe al menos una inmobiliaria cargada en el sistema.

Nombre	SolicitarInmuebles
Operación	<code>solicitarInmuebles(nick:string):set(DTInmueble)</code>
Entrada	<code>nick</code> : Nickname de la inmobiliaria a listar los inmuebles
Salida	Set de DTInmuebles administrados por la inmobiliaria con nickname nick
Descripción	Muestra los inmuebles administrados por la Inmobiliaria con nickname nick

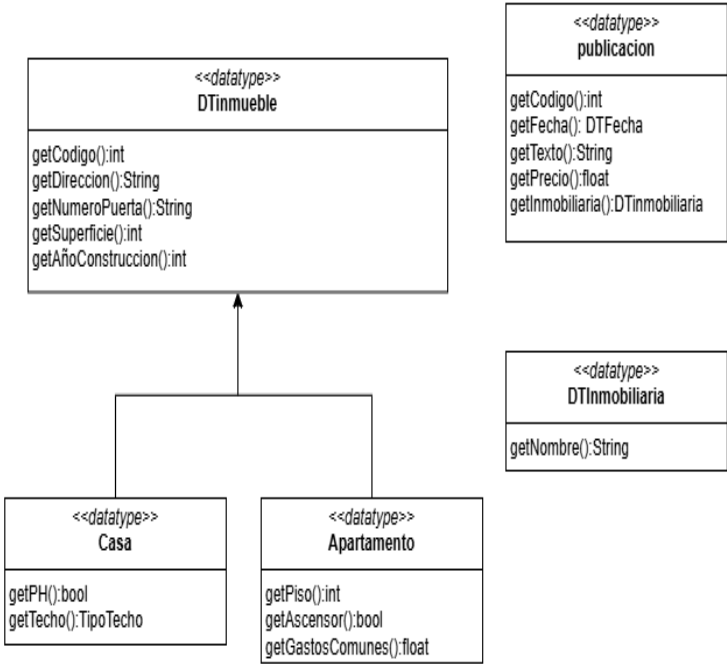
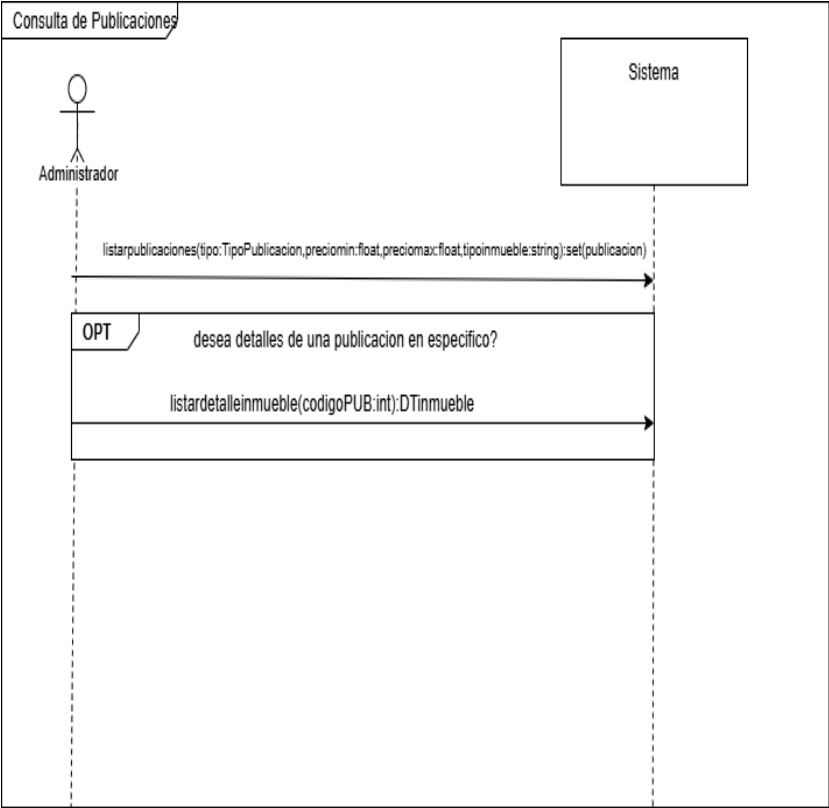
Precondiciones y Postcondiciones
Pre: Existe una inmobiliaria con nickname nick

Nombre	AltaPublicacion
Operación	<code>altaPublicacion(codigo:int, tipo:TipoPublicacion, texto:string, precio:float)</code>
Entrada	<code>codigo</code> : Código del inmueble a crear la publicación. <code>tipo</code> : Tipo de publicación. <code>texto</code> : Texto de la publicación. <code>precio</code> : Precio de la publicacion
Salida	-
Descripción	Si existe una publicación para la fecha y tipo no se da el alta y muestra mensaje de error, si no, da el alta con los datos dados, le asigna un código incremental generado por el sistema, la fecha del sistema y se pone como publicación activa. Por último crea el link con la instancia de AdministraPropiedad del inmueble con código codigo .

Precondiciones y Postcondiciones
Pre: Existe un inmueble con código codigo
Post: Existe una nueva instancia de Publicacion
Post: Existe un link entre la publicación creada y AdministraPropiedad

Consulta de publicaciones

Diagramas de Secuencia del Sistema



Contratos

Nombre	ListarPublicaciones
Operación	<code>ListarPublicaciones(tipo:TipoPublicacion,preciomin:float,preciomax:float,tipoinmueble:string):set(publicacion)</code>
Entrada	<p>tipo: es el tipo de la publicación</p> <p>preciomin: es el precio mínimo de las publicaciones</p> <p>preciomax: es el precio máximo de las publicaciones</p> <p>tipoinmueble: es el tipo del inmueble</p>
Salida	Set de publicaciones que cumplen con los filtros
Descripción	Filtrar y mostrar las publicaciones activas que coincidan con los intereses del administrador, tipo de publicación, rango de precio, y tipo de inmueble en caso de que se especifique

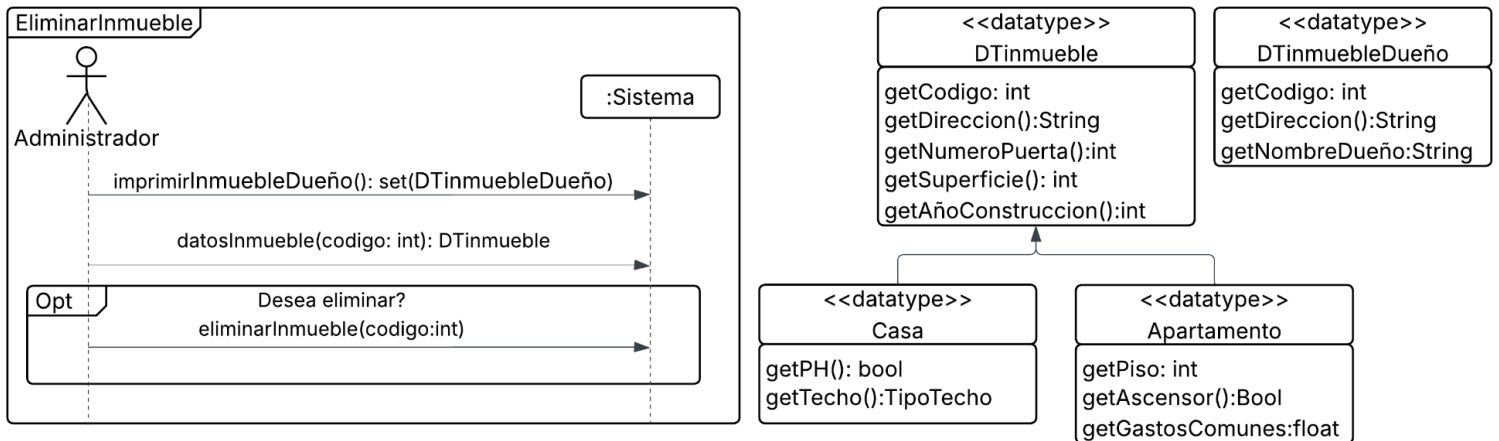
Precondiciones y Postcondiciones
<p>Pre: Existe al menos una publicación activa.</p> <p>Pre: <code>preciomin ≤ preciomax</code>.</p> <p>Pre: <code>tipoinmueble</code> debe ser exactamente "Casas", "Apartamentos" o "Todos".</p>

Nombre	ListarDetalleInmueble
Operación	<code>ListarDetalleInmueble(codigo:int):DTInmueble</code>
Entrada	codigo: es el código de una publicación
Salida	Retorna toda la información del inmueble
Descripción	Muestra el detalle completo del inmueble asociado a la publicación de dicho código

Precondiciones y Postcondiciones
Pre: Existe una publicación activa con ese código

Eliminar Inmueble

Diagramas de Secuencia del Sistema



Contratos

Nombre	Imprimir Inmuebles
Operación	<code>imprimirInmuebleDueño(): set (DTinmuelleDueño)</code>
Entrada	-
Salida	Set de DTinmuelleDueño
Descripción	Retorna el código, la dirección, el nombre del propietario para cada inmueble en el sistema.

Nombre	Listar Detalle Del Inmueble
Operación	<code>datosInmueble(codigo: int): DTinmuelle</code>
Entrada	codigo: Es el codigo del inmueble
Salida	Devuelve un DTinmuelle con información del inmueble
Descripción	Retorna toda la información del inmueble

Nombre	Eliminar Inmuebles
Operación	<code>eliminarInmueble(codigo: Int)</code>
Entrada	codigo: es el codigo de un inmueble
Salida	-
Descripción	Debe dejar inactivas todas las publicaciones del inmueble, las agendas asociadas al inmueble y borrar el inmueble.

Precondiciones y Postcondiciones
Pre: El inmueble debe existir Post: Se borra agenda Post: Se cambian los estados de las publicaciones activas Post: se borra inmueble