

Práctica 3 Parte 2

Sistema de Gestión de Alquileres



[Autores:](#)

Aida Martín Cuéllar

Claudia de la Vieja Lafuente

José Teodosio Lorente Vallecillos

María África Ureña Legaza

Contrato 1.1: consultarContratoAlquiler (TEO)

Nombre	consultarContratoAlquiler (idContratoAlquiler)
Responsabilidad	Solicitar la información de un contrato de alquiler mediante el número de idContratoAlquiler
Tipo	-----
Notas	Se necesita verificar el número de idContratoAlquiler pra consultar la información
Excepciones	-Si el idContratoAlquiler o el contrato de alquiler no existen
Salida	-Información del contrato de alquiler
Precondiciones	Tener un contrato de alquiler creado con la id contrato alquiler
Postcondiciones	-----

Contrato 1.2: altaAveria (TEO)

Nombre	altaAvería (idViviendaAlquiler, descripciónAveria, fechaNotificación)
Responsabilidad	Crear una nueva avería registra en el sistema con el número de idViviendaAlquiler, con la descripción de avería y su fecha de notificación
Tipo	-----
Notas	-----
Excepciones	-Si el id de la vivienda de alquiler no existe o no es correcto
Salida	-----
Precondiciones	-----
Postcondiciones	<ul style="list-style-type: none">Se crea un nuevo objeto avería registrada con un idAvería y su correspondiente idViviendaAlquiler, descripción y fecha de notificación.

Contrato 1.3: altaContratoAlquiler (TEO)

Nombre	altaContratoAlquiler (idViviendaAlquiler, idInquilino, fechaInicioContrato, fechaFinContrato, fianza, cuotaMensual)
Responsabilidad	Crear un nuevo contrato de alquiler registrado en el sistema con el id de la vivienda, el id del inquilino, la fecha de inicio, la fecha de fin, la fianza y la cuota mensual
Tipo	-----
Notas	Se necesita verificar los id de la vivienda y del inquilino
Excepciones	-Si el id de la vivienda no es correcto -Si el id del inquilino no es correcto o no existe -Si fecha fin no es posterior a fecha inicio
Salida	-----
Precondiciones	-El idInquilino existe y es válido
Postcondiciones	<ul style="list-style-type: none">Se ha creado un objeto contrato alquiler registrado con un idContratoAlquiler identificado por idViviendaAlquiler, idInquilino, fechaInicioContrato, fechaFinContrato.

Contrato 1.4: bajaContratoAlquiler(TEO)

Nombre	bajaContratoAlquiler (idContratoAlquiler)
Responsabilidad	Se procede a la baja del contrato de alquiler actualmente vigente con su idContratoAlquiler
Tipo	-----
Notas	-----
Excepciones	-Si el id del contrato de alquiler no es correcto
Salida	-----
Precondiciones	-El contrato de alquiler con ese id debe haberse creado previamente
Postcondiciones	<ul style="list-style-type: none">Se eliminó el contrato de alquiler vigente

Contrato 1.5: altaInquilino (TEO)

Nombre	altaInquilino (dni, nombre, apellido, direccion, numeroCuentaBancaria)
Responsabilidad	Se procede a la alta de un nuevo usuario inquilino en el sistema con el dni, su nombre, apellidos, direccion y numero de cuenta bancaria
Tipo	-----
Notas	-----
Excepciones	-Si el dni del usuario ya existe o no es correcto -Si el nombre y apellidos ya existen -Si el número de cuenta bancaria ya está registrado o no es correcto
Salida	-----
Precondiciones	-----
Postcondiciones	<ul style="list-style-type: none">Se creó un nuevo objeto inquilino registrado con un idInquilino e identificado con su dni, nombre apellidos, direccion y numero de cuenta bancaria

Contrato 1.6: bajaInquilino (TEO)

Nombre	bajaInquilino (idInquilino)
Responsabilidad	Se procede a eliminar al usuario del sistema mediante su idInquilino
Tipo	-----
Notas	-----
Excepciones	-Si el idInquilino no es correcto
Salida	-----
Precondiciones	-El inquilino debe estar dado de alta en el sistema
Postcondiciones	<ul style="list-style-type: none">Se eliminó el usuario del inquilino del sistema

Contrato 1.7: devolverFianzaInquilino (TEO)

Nombre	devolverFianzaInquilino (idContratoAlquiler, fecha)
Responsabilidad	Se procede a devolver la fianza cobrada en el alta de el contrato de alquiler
Tipo	-----
Notas	-----
Excepciones	-Si el idContratoAlquiler no es correcto -Si las condiciones necesarias para devolver la fianza no se han cumplido
Salida	-----
Precondiciones	-El contrato de alquiler se debe haber dado de baja
Postcondiciones	<ul style="list-style-type: none">• Se ha devuelto la fianza al inquilino• Se puede dar de baja el inquilino si lo desea

Contrato 1.8: pagarReciboMetalico (TEO)

Nombre	pagarReciboMetalico(idInquilino, idContratoAlquiler, mensualidad, fecha)
Responsabilidad	Realizar el pago del alquiler
Tipo	-----
Notas	-----
Excepciones	-Si el idInquilino no es correcto -Si el idContratoAlquiler no es correcto -Si la mensualidad no coincide con lo preestablecido -Si la fecha no es correcta
Salida	-----
Precondiciones	-Debe estar registrado previamente el inquilino y coincidir con su id -Debe haber un contrato de alquiler enlazado con su idInquilino
Postcondiciones	<ul style="list-style-type: none">• Se creó un idReciboPagado identificado con el idInquilino, la mensualidad y la fecha concreta

Contrato 1.9:
añadirPagoCuentaCliente(idCliente,idContratoAlquiler,mensualidad,fecha) MARÍA

Nombre	añadirPagoCuentaCliente (idCliente,idContratoAlquiler,mensualidad,fecha)
Responsabilidad	Se va a añadir un nuevo pago que quedará registrado en la cuenta del cliente cuyo id sea idCliente
Tipo	GestionAlquileres
Notas	-----
Excepciones	-Si no existe ningún id que coincida con idCliente -Si no existe ningún contrato coincidente con idContratoAlquiler
Salida	-----
Precondiciones	- Debe estar registrado el cliente en el sistema y su id debe coincidir con idCliente -Debe estar registrado el alquiler en el sistema y su id debe coincidir con idContratoAlquiler
Postcondiciones	<ul style="list-style-type: none"> Se creó un nuevo pago en la cuenta de un cliente, en una fecha y con una mensualidad concretas

Contrato 1.10: listadRecibosDevueltos =
obtenerRegistrosDevueltos(idInquilino,fecha) MARÍA

Nombre	obtenerRegistrosDevueltos (idInquilino,fecha)
Responsabilidad	Se obtendrá una lista de recibos devueltos por parte de un inquilino cuyo id es idInquilino
Tipo	GestionAlquileres
Notas	-----
Excepciones	-No existe ningún inquilino cuyo id coincida con idInquilino registrado en el sistema
Salida	listadRecibosDevueltos
Precondiciones	-Debe estar registrado el inquilino con idInquilino en el sistema
Postcondiciones	<ul style="list-style-type: none"> Se obtiene una lista de recibos devueltos por parte de un inquilino en una fecha concreta

Contrato 1.11:**listarRecibosPagados=obtenerRecibosPagados(idInquilino,fecha)****MARÍA**

Nombre	obtenerRecibosPagados (idInquilino,fecha)
Responsabilidad	Se obtendrá una lista de recibos pagados por parte de un inquilino cuyo id es idInquilino
Tipo	GestionAlquileres
Notas	-----
Excepciones	-No existe ningún inquilino cuyo id coincida con idInquilino registrado en el sistema
Salida	listarRecibosPagados
Precondiciones	-Debe estar registrado el inquilino con idInquilino en el sistema
Postcondiciones	<ul style="list-style-type: none">• Se obtiene una lista de recibos pagados por parte de un inquilino en una fecha concreta

Contrato 1.12:**iniciarTramiteMoroso(idInquilino,listarRecibosDevueltos(fecha)****MARÍA**

Nombre	iniciarTramiteMoroso (idInquilino,listarRecibosDevueltos)
Responsabilidad	Se iniciará el tramite contra un moroso cuyo id coincide con idInquilino
Tipo	GestionAlquileres
Notas	-----
Excepciones	-El idInquilino no se encuentra registrado en el sistema actualmente
Salida	-----
Precondiciones	-El inquilino debe estar anteriormente registrado en el sistema con un id coincidente con idInquilino
Postcondiciones	<ul style="list-style-type: none">• Se inició el trámite contra un moroso

Contrato 1.13:**listadRecibos=generarRecibosAlquiler(idCliente,fecha) MARÍA**

Nombre	generarRecibosAlquiler (idCliente,fecha)
Responsabilidad	Se generará una lista de recibos de alquiler generada por un cliente cuyo id coincide con idCliente en una fecha concreta
Tipo	GestionAlquileres
Notas	-----
Excepciones	-El cliente con id idCliente no está registrado con anterioridad en el sistema
Salida	listadRecibos
Precondiciones	-El cliente debe estar registrado anteriormente en el sistema y además su id debe coincidir con idCliente
Postcondiciones	<ul style="list-style-type: none">• Se creó una lista de recibos de alquiler generados por un cliente en una fecha concretamente

Contrato 1.14: enviarReciboBanco(idRecibo) MARÍA

Nombre	enviarReciboBanco(idRecibo)
Responsabilidad	Va a enviarse un recibo al banco cuyo id coincide con idRecibo
Tipo	GestionAlquileres
Notas	-----
Excepciones	-No existe ningún recibo en el sistema con id coincidente con idRecibo
Salida	-----
Precondiciones	-Deberá estar registrado el recibo en el sistema con anterioridad y además su id debe coincidir con idRecibo
Postcondiciones	<ul style="list-style-type: none">• Se enviará un recibo a un banco concreto, cuyo id es idRecibo

Contrato 2.1:

idCliente=nuevoCliente(dni,nombre,apellidos,direccion,cuentaDelBanco) MARÍA

Nombre	nuevoCliente (dni,nombre,apellidos,direccion,cuentaDelBanco)
Responsabilidad	Se registrará en el sistema un nuevo cliente, cuyos datos asociados serán dni, nombre, apellidos, dirección y una cuenta del banco
Tipo	GestionClientes
Notas	-----
Excepciones	-El dni introducido no es correcto (tiene más/menos dígitos) -El dni introducido ya está registrado en el sistema -El nombre y apellidos coinciden (ambos) con los de otro cliente registrado anteriormente -La dirección es inválida -La cuenta del banco introducida no existe -Faltan alguno de los parámetros mencionados
Salida	idCliente
Precondiciones	-El cliente no puede registrarse con un dni inválido o usado anteriormente por otro cliente -Debe ser válida la cuenta del banco introducida -No puede estar registrado en el sistema alguien con el mismo nombre y apellidos
Postcondiciones	<ul style="list-style-type: none"> Se registrará un nuevo cliente válido en el sistema

Contrato 2.2:

modificarCliente(idCliente,dni,nombre,apellidos,direccion,cuentaDeIBanco) MARÍA

Nombre	modificarCliente (idCliente,dni,nombre,apellidos,direccion,cuentaDelBanco)
Responsabilidad	Se modificará alguno/algunos de los datos asociados a un cliente registrado en el sistema
Tipo	GestionClientes
Notas	-----
Excepciones	-Se modifica el dni por uno que no es válido o que ya está registrado en el sistema -Se modifica la dirección y se introduce una inválida
Salida	-----
Precondiciones	-Debe existir el cliente con idCliente registrado anteriormente en el sistema para poder modificarlo
Postcondiciones	<ul style="list-style-type: none"> Se cambió algún dato asociado a un cliente registrado en el sistema

Contrato 2.3: eliminarCliente(idCliente) MARÍA

Nombre	eliminarCiente (idCliente)
Responsabilidad	Se va a eliminar un cliente que ya está registrado en el sistema con id coincidente con idCliente
Tipo	GestionClientes
Notas	-----
Excepciones	-No existe ningún cliente registrado en el sistema con id idCliente
Salida	-----
Precondiciones	-Debe existir el cliente registrado anteriormente en el sistema con id coincidente con idCliente -Se deben tener los permisos necesarios para poder eliminar a un cliente del sistema
Postcondiciones	<ul style="list-style-type: none">• Se eliminó un cliente del sistema

Contrato 2.3: eliminarCliente(idCliente) AIDA

Nombre	eliminarCliente(idCliente)
Responsabilidad	Eliminar el cliente indicado
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	
Precondiciones	<ul style="list-style-type: none">• Existe el objeto <i>Cliente</i> identificado por <i>idCliente</i>
Postcondiciones	<ul style="list-style-type: none">• Se eliminó cualquier enlace entre el objeto de la clase <i>Cliente</i> identificado por <i>idCliente</i> y cualquier objeto de cualquier clase relacionado con él y también se eliminó ese objeto de esa clase• Se eliminó el objeto <i>Cliente</i> identificado por <i>idCiente</i>

Contrato 2.4: infoCliente = consultarCliente(idCliente) AIDA

Nombre	infoCliente = consultarCliente(idCliente)
Responsabilidad	Consultar la información de un cliente indicado
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	infoCliente = {idCliente, dni, nombre, apellidos, direccion, cuentaDelBanco} para el objeto de la clase <i>Cliente</i> identificado por <i>idCliente</i>
Precondiciones	<ul style="list-style-type: none">Existe el objeto de la clase <i>Cliente</i> identificado por <i>idCliente</i>
Postcondiciones	

Contrato 2.5. listaldViviendasAlquiler = viviendasAlquiladasCliente(idCliente) AIDA

Nombre	listaldViviendasAlquiladas = viviendasAlquiladasCliente(idCliente)
Responsabilidad	Consultar las viviendas que el cliente indicado tiene alquiladas
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	listaldViviendasAlquiladas = lista de {idViviendaAlquiler} que el cliente identificado por <i>idCliente</i> tiene alquiladas
Precondiciones	<ul style="list-style-type: none">Existe el objeto de la clase <i>Cliente</i> identificado por <i>idCliente</i>
Postcondiciones	

Contrato 2.6: recibos = cobrarAlquiler(idCliente, idViviendaAlquiler, mensualidad, fecha) AIDA

Nombre	recibos = cobrarAlquiler(idCliente, idViviendaAlquiler, mensualidad, fecha)
Responsabilidad	Cobrar el alquiler de la vivienda indicada de un cliente determinado
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	recibos = {idCliente, idViviendaAlquiler, mensualidad, fecha, IngresoInquilino:cargoBancario, PagoCliente:ingresoCuenta}
Precondiciones	<ul style="list-style-type: none"> ● Existe el objeto <i>Cliente</i> identificado por <i>idCliente</i> ● Existe el objeto <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> ● Existe un enlace entre el objeto <i>Cliente</i> identificado por <i>idCliente</i> y el objeto <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> ● Existe un objeto de la clase <i>ContratoAlquiler</i> enlazado con el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i>
Postcondiciones	<ul style="list-style-type: none"> ● Se creó un objeto, recibo, de la clase <i>RecibosAlquiler</i> inicializado con <i>mensualidad</i> ● Se creó un enlace entre recibo y el objeto de la clase <i>ContratoAlquiler</i> que esté enlazado con el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> ● Se creó un objeto, ingrInq, de la clase <i>IngresoInquilino</i> inicializado correctamente (con <i>fecha</i>) ● Se creó un enlace entre ingrInq y recibo ● Se creó un objeto, pagoCli, de la clase <i>PagoCliente</i> inicializado correctamente (con <i>fecha</i>) ● Se creó un enlace entre pagoCli y recibo

Contrato 3.1: idViviendaAlquiler = altaViviendaAlquiler(idCliente, esCasa, direccion, cuotaMensualBase, variacionAdmitidaCuota, descripcion, foto) AIDA

Nombre	idViviendaAlquiler = altaViviendaAlquiler(idCliente, esCasa, direccion, cuotaMensualBase, variacionAdmitidaCuota, descripcion, foto)
Responsabilidad	Dar de alta una vivienda para alquiler con las características indicadas
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	idViviendaAlquiler = identificador del objeto de la clase <i>ViviendaAlquiler</i> creado
Precondiciones	<ul style="list-style-type: none"> • Existe el <i>Cliente</i> identificado por <i>idCliente</i>
Postcondiciones	<ul style="list-style-type: none"> • Se creó un objeto, vivAl, de la clase <i>ViviendaAlquiler</i> inicializado con <i>direccion</i>, <i>cuotaMensualBase</i>, <i>variacionAdmitidaCuota</i>, <i>descripcion</i> • Se creó un objeto de la clase <i>Casa</i> o <i>Local</i> en función de lo indicado por <i>esCasa</i> debidamente inicializado y se creó un enlace entre este objeto y vivAl • Se creó un objeto de la clase <i>Foto</i> inicializado correctamente (con <i>foto</i>) y se creó un enlace entre este objeto y vivAl • Se creó un enlace entre el objeto <i>Cliente</i> identificado por <i>idCliente</i> y vivAl

Contrato 3.2: modificarDatosViviendaAlquiler(idViviendaAlquiler, idCliente, esCasa, direccion, cuotaMensualBase, variacionAdmitidaCuota, descripcion, foto) AIDA

Nombre	modificarDatosViviendaAlquiler(idViviendaAlquiler, idCliente, esCasa, direccion, cuotaMensualBase, variacionAdmitidaCuota, descripcion, foto)
Responsabilidad	Modificar los datos de una vivienda de alquiler determinada
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	
Precondiciones	<ul style="list-style-type: none"> • Existe un objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> • Existe un objeto de la clase <i>Cliente</i> identificado por <i>idCliente</i> • Existe un enlace entre el objeto de <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> y el objeto de <i>Cliente</i> identificado por <i>idCliente</i>
Postcondiciones	<ul style="list-style-type: none"> • Se modificaron los atributos del objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> con los proporcionados en la operación

Contrato 3.3: bajaViviendaAlquiler(idViviendaAlquiler) AIDA

Nombre	bajaViviendaAlquiler(idViviendaAlquiler)
Responsabilidad	Dar de baja la vivienda de alquiler indicada
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	
Precondiciones	<ul style="list-style-type: none"> • Existe el objeto de <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i>
Postcondiciones	<ul style="list-style-type: none"> • Se eliminó el enlace entre el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> y el objeto de la clase <i>Cliente</i> asociado con esa vivienda • Se eliminó el enlace entre el objeto de <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> y el objeto de la clase <i>Casa</i> o <i>Local</i> relacionado con esa vivienda y también se eliminó ese objeto de la clase <i>Casa</i> o <i>Local</i>

	<ul style="list-style-type: none"> Se eliminó el enlace entre el objeto de <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> y el objeto de la clase <i>Foto</i> relacionado con esa vivienda y también se eliminó ese objeto de la clase <i>Foto</i> Se eliminó el enlace entre el objeto de <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> y cualquier otro objeto de cualquier otra clase relacionado con esa vivienda y también se eliminó el objeto de esa clase Se eliminó el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i>
--	---

Contrato 3.4: infoViviendaAlquiler = consultarDatosViviendaAlquiler(idViviendaAlquiler) AIDA

Nombre	infoViviendaAlquiler = consultarDatosViviendaAlquiler(idViviendaAlquiler)
Responsabilidad	Consultar la información de la vivienda de alquiler indicada
Tipo	Alquintervivac
Notas	
Excepciones	
Salida	infoViviendaAlquiler = {idViviendaAlquiler, direccion, cuotaMensualBase, variacionAdmitidaCuota, descripcion} para el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i> + {imagen, descripcion} de la foto asociada + si es una casa {superficieUtil, numeroHabitaciones, numeroBaños} + si es un local {superficie} + si existe una avería asociada {descripciónAveria, fechaNotificacion, solventad} si se encuentra en ese momento alquilada o no
Precondiciones	<ul style="list-style-type: none"> Existe el objeto de la clase <i>ViviendaAlquiler</i> identificado por <i>idViviendaAlquiler</i>
Postcondiciones	

Contrato 3.5: idOfertaAlquiler = altaOfertaViviendaAlquiler(idViviendaAlquiler, descripcion, fechalnicio, fechaFin) CLAUDIA

Nombre	idOfertaAlquiler = altaOfertaViviendaAlquiler(idViviendaAlquiler, fechalnicio, fechaFin, condicionesAlquiler)
Responsabilidad	Crear una nueva oferta de vivienda que se va a poner en alquiler en el sistema con el identificador de la vivienda, la fecha de inicio y fin del alquiler y con las condiciones del alquiler.
Tipo	Alquintervivac
Notas	-----
Excepciones	- Si el id de la vivienda ya existe o no es correcto.
Salida	idOfertaAlquiler.
Precondiciones	<ul style="list-style-type: none"> • Existe el objeto ViviendaAlquiler identificado por idViviendaAlquiler. • La fechalnicio y fechaFin están en un rango no existente.
Postcondiciones	<ul style="list-style-type: none"> • Se creó un nuevo objeto, <i>miOferta</i>, de la clase OfertaAlquiler debidamente inicializado. • Se creó un enlace entre <i>miOferta</i> y el objeto ViviendaAlquiler, identificado por idViviendaAlquiler.

Contrado 3.6: idPublicidad = altaPublicidad(idViviendaAlquiler, descripcion, fechalnicio, fechaFin) CLAUDIA

Nombre	idPublicidad = altaPublicidad(idViviendaAlquiler, descripcion, fechalnicio, fechaFin)
Responsabilidad	Crear publicidad sobre una vivienda que se va ha puesto en oferta en el sistema con el identificador de la vivienda, una descripción de ella, la fecha de inicio y de fin de alquiler.
Tipo	Alquintervivac.
Notas	-----
Excepciones	<ul style="list-style-type: none"> • Si el id de la vivienda ya existe o no es correcto. • La fechalnicio y fechaFin están en un rango no existente.
Salida	idPublicidad.
Precondiciones	La vivienda tiene que estar previamente dada de alta en oferta de vivienda de alquiler
Postcondiciones	<ul style="list-style-type: none"> • Se creó un nuevo objeto, <i>miPublicidad</i>, de la clase Publicidad debidamente inicializado. • Se creó un enlace entre <i>miPublicidad</i> y el objeto ViviendaAlquiler, identificado por idViviendaAlquiler.

Contrato 3.7: bajaOfertaAlquilerVivienda(idOfertaAlquiler) CLAUDIA

Nombre	bajaOfertaAlquilerVivienda(idOfertaAlquiler)
Responsabilidad	Solicitar la baja de una vivienda que está en alquiler.
Tipo	Alquintervivac
Notas	-----
Excepciones	-----
Salida	-----
Precondiciones	<ul style="list-style-type: none">• Se puede eliminar el alquiler de la vivienda del sistema.
Postcondiciones	<ul style="list-style-type: none">• Se eliminó todos los objetos de la clase OfertaAlquiler identificado por idOferta.

Contrato 3.8: bajaPublicidad(idPublicidad) CLAUDIA

Nombre	bajaPublicidad(idPublicidad)
Responsabilidad	Solicitar la baja de la publicidad de una vivienda.
Tipo	Alquintervivac
Notas	-----
Excepciones	-----
Salida	-----
Precondiciones	<ul style="list-style-type: none">• Se puede eliminar la publicidad de una vivienda que se haya dado de baja previamente en el sistema.
Postcondiciones	<ul style="list-style-type: none">• Se eliminó todos los objetos de la clase Publicidad identificado por idPublicidad

Contrato 3.9: modificarOfertaAlquilerVivienda(idOfertaAlquiler, fechaInicio, fechaFin, condicionesAlquiler) CLAUDIA

Nombre	modificarOfertaAlquilerVivienda(idOfertaAlquiler, fechaInicio, fechaFin, condicionesAlquiler)
Responsabilidad	Permite al administrativo modificar los datos sobre el alquiler de una vivienda identificada como idOfertaAlquiler.
Tipo	Alquintervivac
Notas	-----
Excepciones	<ul style="list-style-type: none">• La fechaInicio y fechaFin están en un rango no existente.
Salida	-----
Precondiciones	<ul style="list-style-type: none">• Se puede modificar la oferta de alquiler de una vivienda que esté dada de alta en el sistema.
Postcondiciones	<ul style="list-style-type: none">• Se modificaron los atributos del objeto de la clase OfertaAlquiler identificado por idOfertaAlquiler.

Contrato 3.10: modificarPublicidad(idPublicidad, descripcion, fechaInicio, fechaFin) CLAUDIA

Nombre	modificarPublicidad(idPublicidad, descripcion, fechaInicio, fechaFin)
Responsabilidad	Permite al administrativo modificar los datos sobre la publicidad de una vivienda identificada como idPublicidad.
Tipo	Alquintervivac
Notas	-----
Excepciones	-----
Salida	-----
Precondiciones	
Postcondiciones	<ul style="list-style-type: none">• Se modificaron los atributos de la clase Publicidad identificado por idPublicidad.

**Contrato 3.11: infoOfertaAlquiler =
consultarOfertaAlquiler(idOfertaAlquiler) CLAUDIA**

Nombre	infoOfertaAlquiler = consultarOfertaAlquiler(idOfertaAlquiler)
Responsabilidad	Permite al administrativo consultar los datos de las ofertas en alquiler identificadas como idOfertaAquiler
Tipo	Alquintervivac
Notas	-----
Excepciones	-----
Salida	infoOfertaAlquiler = consultarOfertaAlquiler(idOfertaAlquiler)
Precondiciones	<ul style="list-style-type: none">• La vivienda ofertada identificada por idOfertaAlquiler tiene que estar en el sistema
Postcondiciones	-----