

Práctica 4

DISEÑO

Temporización

Comienzo de la práctica	Duración
Semana del 22 de Mayo de 2017	3 semanas

Actividades a realizar

A partir de los **contratos proporcionados en este guión**, se llevarán a cabo las siguientes actividades:

1. Creación del **modelo de interacción de objetos** de todas las operaciones especificadas, representándolo mediante un conjunto de diagramas de comunicación de UML.
2. Obtención del **diagrama de clases del diseño** a partir del modelo de interacción de objetos obtenido anteriormente, representándolo mediante un diagrama de clases de UML.

Recomendaciones para realizar la práctica

El modelo de interacción de objetos debe hacerse aplicando las reglas explicadas en clase de teoría en el tema 3.3. Este modelo representa el conjunto de objetos que colaboran entre sí para llevar a cabo una operación, junto con las responsabilidades de cada uno de ellos. Para ello, sirven como base los contratos y, como guía, el modelo conceptual.

A continuación, se deberá crear el diagrama de clases del diseño, teniendo en cuenta todo el proceso explicado en el tema 3.4. En este caso, se usará como base el modelo de interacción de objetos y, como guía, el modelo conceptual.

Entrega de los documentos generados

Documentación a aportar (Tipo de entrega)	Fecha de entrega
<ul style="list-style-type: none">• Modelo de interacción de objetos (a repartir entre los integrantes del grupo, indicando el nombre del estudiante que ha realizado cada diagrama).• Diagrama de clases del diseño (Grupal)	11-Junio-2017

Contratos de las operaciones a diseñar

Administración de usuarios

Nombre	altaRegistro(nombreUsuario contraseña, direccionCorreo)
Responsabilidad	Crear un nuevo objeto UsuarioRegistrado que puede ser un Administrador o un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	- Existe un UsuarioRegistrado con el mismo nombreUsuario.
Salida	
Precondiciones	
Poscondiciones	<ul style="list-style-type: none"> Se creó el objeto de la clase <i>UsuarioRegistrado</i> (<i>Administrador o Usuario</i>), unUsuarioRegistrado, debidamente inicializado. Si el objeto unUsuarioRegistrado es un objeto de la clase Usuario, se modificó el atributo visibilidad a "falso".

Nombre	solicitarBajaRegistro(idUsuario)
Responsabilidad	Registrar la solicitud de baja de un Usuario del Sistema a petición del propio usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> Se modificó el atributo pendienteBaja a "verdadero".

Nombre	datosUsuario=obtenerDatosUsuario(idUsuario)
Responsabilidad	Obtener la información asociada a un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	datosUsuario= {nombreUsuario, contraseña, direccionCorreo} para el objeto de la clase Usuario identificado por idUsuario.
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	

Gestión de perfil

Nombre	idUsuario=introducirPerfil(idUsuario, nombre, telefono, añoNacimiento, breveDescPersonal, foto, direccionPostal)
Responsabilidad	Completar los datos de un Usuario y poner su perfil visible.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	- Existe ya un perfil para el usuario.
Salida	idUsuario
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> Se modificaron todos los atributos del objeto Usuario identificado por idUsuario especificados en la operación. Se modificó el atributo visibilidad a "verdadero".

Nombre	infoPerfil=consultarPerfil(idUsuario)
Responsabilidad	Proporcionar la información del perfil asociado a un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	infoPerfil = {nombre, telefono, añoNacimiento, breveDescPersonal, foto, direccionPostal, visibilidad, lista de preferenciaPago}} para el objeto de la clase Usuario identificado por idUsuario.
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	

Nombre	establecerFormaPago(idUsuario, listaTipoTransaccion)
Responsabilidad	Añadir nuevas formas de pago a un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> Se modificó el valor del atributo preferenciaPago del objeto de la clase Usuario, identificado por idUsuario, al valor del parámetro listaTipoTransaccion

Nombre	añadirVehiculo(idPropietario, matricula, marca, modelo, color, numeroPlazas, categoría, confor)
Responsabilidad	Crear un nuevo objeto Vehículo, y asignarlo a un UsuarioPropietario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	- Existe un objeto de la clase Vehículo con la misma matricula.
Salida	
Precondiciones	- El objeto de la clase UsuarioPropietario identificado por idPropietario debe existir.
Poscondiciones	<ul style="list-style-type: none"> Se creó un objeto de la clase Vehículo, miVehiculo, debidamente inicializado. Se creó un enlace entre el objeto de la clase UsuarioPropietario identificado por idPropietario y el objeto miVehiculo.

Nombre	eliminarVehiculo(idPropietario, idVehiculo)
Responsabilidad	Eliminar un objeto Vehículo propiedad de un UsuarioPropietario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	<ul style="list-style-type: none"> No existe un enlace entre el objeto de la clase UsuarioPropietario, identificado por idPropietario y el objeto de la clase Vehículo identificado por idVehiculo. El objeto Vehículo "formaParte" de un PlanViaje o PlanAlquiler con diaSalida de PlanViaje o primerDiaAlquiler de PlanAlquiler posterior a la fecha actual. /*Existe un plan de viaje o un plan de alquiler activo para ese vehículo */
Salida	
Precondiciones	- El objeto de la clase UsuarioPropietario identificado por idPropietario debe existir.
Poscondiciones	<ul style="list-style-type: none"> Se eliminó el enlace entre el objeto de la clase UsuarioPropietario identificado por idPropietario y el objeto de la clase Vehículo identificado por idVehiculo. Se eliminó el objeto de la clase Vehículo identificado por idVehiculo.

Gestión de alquileres

Nombre	definirPlanAlquiler(idPropietario, idVehiculo, fechaInicio, fechaFin, ciudadRecogida)
Responsabilidad	Crear un PlanAlquiler por parte de un UsuarioPropietario para un Vehículo que sea de su propiedad.
Tipo	Tapuntas(Sistema)
Notas	
Excepciones	<ul style="list-style-type: none"> - En el período comprendido entre fechaInicio y fechaFin el objeto Vehículo, identificado por idVehiculo, no está disponible, es decir, forma parte de otro plan de alquiler o plan de viaje. - El objeto Vehículo identificado por idVehiculo no es propiedad del UsuarioPropietario identificado por idPropietario.
Salida	
Precondiciones	<ul style="list-style-type: none"> - El objeto de la clase UsuarioPropietario identificado por idPropietario debe existir.
Poscondiciones	<ul style="list-style-type: none"> ▪ Se creó el objeto PlanAlquiler, mPlanAlquiler, debidamente inicializado, con el atributo visible a "falso". ▪ Se creó el enlace entre miPlanAlquiler y el objeto de la clase Vehículo identificado por idVehiculo. ▪ Se creó el enlace entre miPlanAlquiler y el objeto de la clase UsuarioPropietario identificado por idPropietario.

Nombre	misPlanesAlquiler=obtenerPlanesAlquiler(idPropietario)
Responsabilidad	Proporcionar todos los planes de alquiler de un UsuarioPropietario que estén vigentes y sin ofertar.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	misPlanesAlquiler = lista de {matricula de Vehículo, (primerDiaAlquiler, ultimoDiaAlquiler, costeAlquilerAIDia, ciudadRecogida) de PlanAlquiler} para todos los objetos de la clase PlanAlquiler del objeto de la clase UsuarioPropietario identificado por idPropietario que cumplan que el atributo visible sea "falso" y FECHA_ACTUAL < primerDiaAlquiler de PlanAlquiler
Precondiciones	<ul style="list-style-type: none"> - El objeto de la clase UsuarioPropietario identificado por idPropietario debe existir.
Poscondiciones	

Nombre	ofertarPlanAlquiler(idPropietario, idPlanAlquiler)
Responsabilidad	Hacer visible un plan de alquiler previamente definido.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	
Precondiciones	- El objeto de la clase UsuarioPropietario identificado por idPropietario debe existir.
Poscondiciones	▪ Se modificó a "verdadero" el atributo visible del objeto PlanAlquiler, identificado por idPlanAlquiler, ofertado por el objeto UsuarioPropietario identificado por idPropietario.

Nombre	listaOfertas=buscarOfertasAlquiler(ciudadRecogida, fechaInicio, fechaFin)
Responsabilidad	Proporcionar información de todas las ofertas de planes de alquiler que existan en una ciudad y para un determinado período de fechas.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	
Salida	listaOfertas = lista de {costeAlquilerAIDia de PlanAlquiler, (marca, modelo, confor, numeroPlazas,categoría) de Vehículo, (nombre, preferenciaCobro) de UsuarioPropietario} para todos los objetos de la clase PlanAlquiler que cumplan que ciudadRecogida de PlanAlquiler = ciudadRecogida y primerDiaAlquiler de PlanAlquiler <=fechaInicio y ultimoDiaAlquiler de PlanAlquiler >= fechaFin. /*Esta lista estará ordenada por orden ascendente de costeAlquilerAIDia de PlanAlquiler */
Precondiciones	
Poscondiciones	

Nombre	solicitarAlquilerVehiculo(idUsuario, idPlanAlquiler, fechaInicio, fechaFin, formaPagoElegida)
Responsabilidad	Crear un objeto Alquiler que refleje que el usuario idUsuario ha solicitado el plan de Alquiler idPlanAlquiler entre las fechas, fechaInicio y fechaFin, indicando que la forma de pago que ofrece es formaPagoElegida.
Tipo	Tapuntas (Sistema)
Notas	El objeto Alquiler creado en esta operación debe mantener el orden definido entre los objetos Alquiler de un PlanAlquiler. Estos objetos están ordenados de forma ascendente por el atributo fechaInicio de Alquiler.
Excepciones	<ul style="list-style-type: none"> - Existe un objeto Alquiler del objeto PlanAlquiler identificado por idPlanAlquiler, con estado "ACEPTADO" y alguna fecha, de las comprendidas entre fechaInicio y fechaFin, está incluida en el intervalo definido por fechaInicio y fechaFin de dicho objeto Alquiler. <p>/*Ya existe un alquiler aceptado que empieza o acaba dentro de las fechas solicitadas para ese PlanAlquiler*/</p>
Salida	
Precondiciones	<ul style="list-style-type: none"> - El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> ▪ Se creó un objeto Alquiler, nuevoAlquiler, debidamente inicializado y con estado a "PENDIENTE". ▪ Se creó un enlace entre nuevoAlquiler y el objeto Usuario identificado por idUsuario. ▪ Se creó un enlace entre nuevoAlquiler y el objeto PlanAlquiler identificado por idPlanAlquiler.

Nombre	aceptarSolicitudDeAlquiler(idUsuario, idAlquiler)
Responsabilidad	Indicar que la solicitud de alquiler, identificada por idAlquiler, y realizada por el Usuario identificado por idUsuario es aceptada.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	<ul style="list-style-type: none"> - No existe el objeto Alquiler identificado por idAlquiler. - Existe un objeto Alquiler relacionado con el objeto PlanAlquiler al que pertenece el objeto Alquiler identificado por idAlquiler, con estado "ACEPTADO" y, alguna fecha de las comprendidas entre fechaInicio y fechaFin del objeto Alquiler identificado por idAlquiler, está incluida en el intervalo definido por fechaInicio y fechaFin de dicho objeto Alquiler. <p>/*Ya existe un alquiler aceptado que empieza o acaba dentro de ese objeto Alquiler*/</p>
Salida	
Precondiciones	<ul style="list-style-type: none"> - El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> ▪ Se modificó a "ACEPTADO" el valor del atributo estado del objeto Alquiler identificado por idAlquiler correspondiente al objeto Usuario identificado por idUsuario.

Nombre	realizarPagoAlquiler(idUsuario, idAlquiler, detallesPago)
Responsabilidad	Realizar el pago de un Alquiler por parte de un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	- No existe el objeto de la clase Alquiler identificado por idAlquiler.
Salida	
Precondiciones	- El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> ▪ Se creó un objeto de la clase Pago, miPago, debidamente inicializado, con fecha a FECHA_ACTUAL e importe = (ultimoDiaAlquiler – primerDiaAlquiler) * costeAlquilerAlDia. ▪ Se creó un enlace entre miPago y e objeto de la clase Alquiler identificado por idAlquiler.

Nombre	valorarAlquiler(idAlquiler, puntuación, comentario)
Responsabilidad	Realizar la valoración de un Alquiler por parte de un Usuario.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	<ul style="list-style-type: none"> - No existe el objeto de la clase Alquiler identificado por idAlquiler. - El objeto de la clase Alquiler identificado por idAlquiler tiene su atributo estado a "RECHAZADO" o fechaInicio > FECHA_ACTUAL.
Salida	
Precondiciones	
Poscondiciones	<ul style="list-style-type: none"> ▪ Se creó un objeto de la clase ValoracionAlquiler, miValoracion, debidamente inicializado y con fecha a FECHA_ACTUAL. ▪ Se creó un enlace entre miValoracion y el objeto de la clase Alquiler identificado por idAlquiler.

Nombre	cancelarSolicitudAlquiler(idUsuario, idAlquiler)
Responsabilidad	Anular la solicitud que un Usuario ha realizado de un Alquiler.
Tipo	Tapuntas (Sistema)
Notas	
Excepciones	<ul style="list-style-type: none"> - No existe el objeto de la clase Alquiler identificado por idAlquiler. - EL objeto de la clase Alquiler identificado por idAlquiler tiene fechaInicio <= FECHA_ACTUAL y estado a "ACEPTADO".
Salida	
Precondiciones	<ul style="list-style-type: none"> - El objeto de la clase Usuario identificado por idUsuario debe existir.
Poscondiciones	<ul style="list-style-type: none"> ▪ Se eliminó el enlace entre el objeto de la clase Usuario identificado por idUsuario y el objeto de la clase Alquiler identificado por idAlquiler. ▪ Se eliminó el enlace entre el objeto de la clase Alquiler identificado por idAlquiler y el objeto de la clase PlanAlquiler del que forma parte el objeto Alquiler. ▪ Se eliminó el objeto Alquiler identificado por idAlquiler.