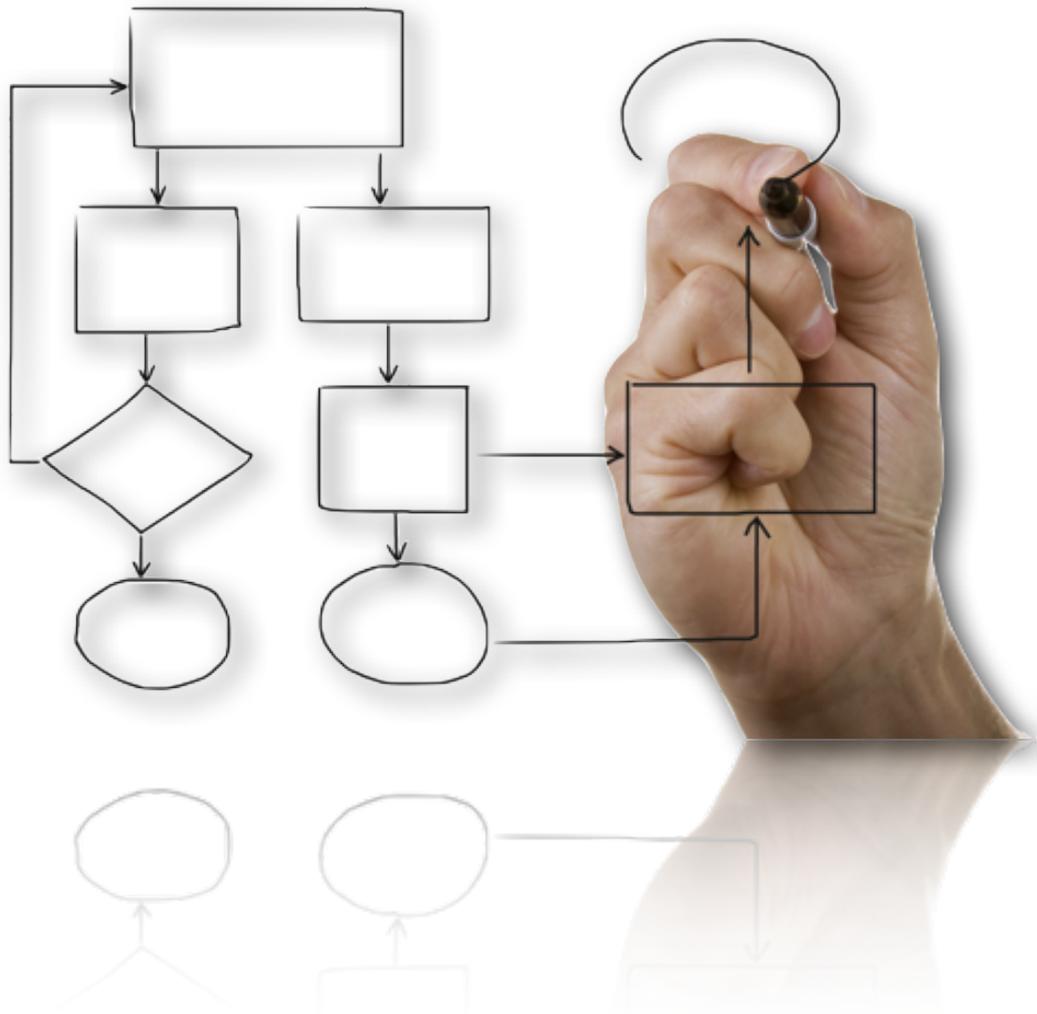




Antonio Martín Moreno



MATERIAL MARKET

TFG GRADO INFORMÁTICA - ÁREA JAVA J2EE - FECHA 17 DE ABRIL 2014

ESTUDIANTE: ANTONIO MARTÍN MORENO

CONSULTOR: ALBERT GRAU

Análisis de Requisitos	3
Visión general	3
Actores del sistema	3
Requisitos funcionales	4
Requisitos de la interfaz de usuario: aspectos técnicos	5
Requisitos de la interfaz de usuario: aspectos de diseño y usabilidad	5
Casos de uso	6
Casos de uso: autenticación y gestión de cuentas	7
Login	8
Alta de usuario	9
Baja de cuenta de usuario	10
Modificación de cuenta de usuario	11
Recuperar contraseña	12
Casos de uso: gestión de ofertas y demandas	13
Alta de oferta	14
Consulta de ofertas publicadas	15
Modificación de oferta	16
Baja de oferta	17
Casos de uso de demandas	18
Casos de uso: suscripción de ofertas y demandas	19
Búsqueda de ofertas	20
Detalle de oferta	21
Suscripción de ofertas	22

Añadir ofertas a favoritos	24
Consulta de ofertas favoritas	25
Eliminar ofertas de favoritos	26
Casos de uso de demandas	27
Subscripción de demandas	28
Casos de uso: histórico de ofertas y demandas	29
Consulta de histórico de ofertas	30
Detalle de histórico de la oferta	31
Valoración del usuario emisor de la oferta	32
Casos de uso de demandas	33
Diseño técnico	34
Arquitectura del sistema	35
Diagrama entidad Relación	37
Diagrama de clases: entidades	39
Diagrama de clases: servicios	40
Diagrama de clases: front-end	42
Especificaciones	44
Entregables software	45
Conclusiones	46
Bibliografía y referencias	47

Análisis de Requisitos

Visión general

Tal y como se había comentado, se implementará un sistema capaz de proporcionar las funcionalidades suficientes como para los usuarios puedan realizar sus tareas de comercio electrónico. La usabilidad de la interfaz debe ser lo mas y sencilla de utilizar, para que la curva de aprendizaje del usuario sea mínima, de tal forma que la aplicación sea una herramienta atractiva, funcional y eficaz para aquellas personas que la usen.

Actores del sistema

Principalmente van a existir dos actores: los usuarios por un lado y el administrador por otro. Los usuarios harán uso de la plataforma con el fin de realizar tareas de comercio electrónico, tanto en cuando, el administrador realizará tareas de supervisión de que el sistema funciona correctamente, tanto técnicamente como funcionalmente. Velará que la información que existe en el sistema es la adecuada.

Requisitos funcionales

La aplicación tratará de cubrir, en un principio, las funcionalidades básicas que cualquier plataforma de este tipo debería tener:

Para el perfil o rol de usuario:

- ✦ Funcionalidades relacionadas con el tema de la seguridad, esto implica la gestión de cuentas de usuario: alta, baja, modificación de la cuenta, además de la recuperación de la contraseña en caso de olvido y el login del sistema.
- ✦ Funcionalidades relacionadas con la publicación y gestión de las ofertas y demandas emitidas por un usuario: alta, baja, modificación y consulta de ofertas y demandas.
- ✦ Funcionalidad relacionadas con la suscripción de ofertas y demandas por un usuario: búsqueda, detalle y suscripción de ofertas y demandas. A tener en cuenta, que en el caso de las demandas, un usuario puede publicar una demanda y otros pueden suscribir ofertas para estas demandas. El usuario que ha publicado la demanda podrá ver que ofertas suscritas que tiene asociadas esta demanda. A parte, un usuario podrá suscribirse a ofertas de otros usuarios, y en ese proceso, realizará el pago del producto ofertado.
- ✦ Funcionalidad relacionadas con la gestión de ofertas y demandas favoritas por un usuario: búsqueda, detalle, añadir y eliminar ofertas y demandas favoritas.
- ✦ Funcionalidad relacionadas con la gestión del histórico de ofertas y demandas por un usuario: búsqueda, detalle, y valoración de ofertas y demandas.

Para el perfil o rol de administrador:

- ✦ Funcionalidades relacionadas con el tema de la seguridad: el sistema validará las credenciales del administrador para dejarle ingresar en el sistema con ese rol.
- ✦ Funcionalidad relacionadas con la suscripción de ofertas y demandas por los usuarios: búsqueda, detalle y modificación ofertas y demandas en el caso de que el administrador lo considere correspondiente. Esto implica que puede eliminar las ofertas y demandas, o simplemente las valoraciones de los usuarios en el caso que lo estime oportuno, para velar por un correcto funcionamiento del sistema.

Requisitos de la interfaz de usuario: aspectos técnicos

Uno de los aspectos mas importantes a la hora de utilizar un aplicación web es el tiempo de respuesta. Al utilizar una arquitectura cliente servidor (web), cada vez que se produce una petición por parte del usuario, se produce un retardo hasta que el servidor procesa la petición, obtiene los datos, renderiza el resultado de la pagina a mostrar al usuario y se la envía para que este pueda ver la información solicitada. Con el fin, de paliar este efecto y que se minimice el tiempo de espera, se diseñará las paginas de tal forma que su contenido sea sencillo de procesar y servir por parte del servidor web. Se harán uso de peticiones asíncronas siempre que sea posible para agilizar las comunicaciones y que el usuario no tenga la sensación de estar esperando. El objetivo es que la navegación por la aplicación sea fluida.

Requisitos de la interfaz de usuario: aspectos de diseño y usabilidad

Otro de los aspectos mas importantes, y clave del éxito de una aplicación, reside en su usabilidad. Para ello, el uso de una interfaz gráfica sencilla y clara es requisito indispensable. Los textos de las opciones que implementan las funcionalidades deben ser claros y descriptivos, es decir, no deben llevar a confusión. La navegación entre paginas tiene que ser lógica y natural, de tal forma que no sea necesario aprender nada, simplemente dejandose guiar por el flujo de la aplicación, un usuario puede llegar a realizar las tareas sin esfuerzo.

Por otro lado los colores y los estilos empleados para diseñar la aplicación deben ser sencillos y usados en contraste, para que no causen fatiga visual. Una apariencia sencilla y funcional es un factor clave para éxito de una aplicación.

Tal y como se indicaba en el punto anterior, para que la carga de las paginas sea rápida, se limitará el uso de imágenes pesadas con el fin de que el renderizado de las paginas sea rápido y el usuario no tenga que estar esperando.

Casos de uso

Una vez determinados los requisitos funcionales que va a realizar la aplicación, la siguiente tarea es realizar la especificación de los mismos utilizando los diagramas de casos de uso.

Primeramente se han definido los casos de uso de “grano grueso”, es decir, aquellos casos que engloban varios casos de uso de “grano fino”, para seguidamente especificar los casos de uso de un nivel inferior o de “grano fino”. La siguiente imagen ilustra los casos de uso a nivel general o de “grano grueso”:

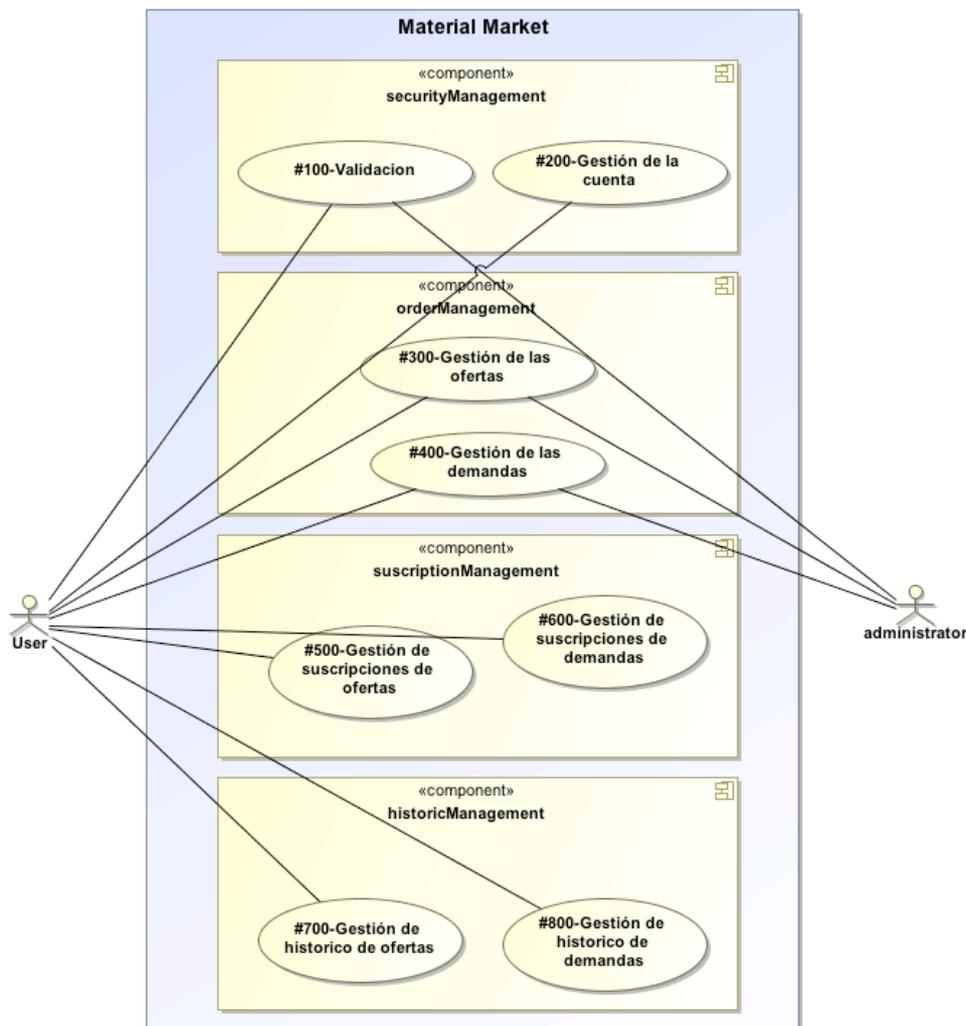


Diagrama de casos de uso del sistema a nivel general

Casos de uso: autenticación y gestión de cuentas

Tal y como se ha especificado en la figura anterior, se han definido los casos de uso de validación de usuario y de gestión de la cuenta del usuario como casos de uso de “grano grueso”. A continuación se muestran los casos de uso de “grano fino” en los que se descomponen los anteriores:

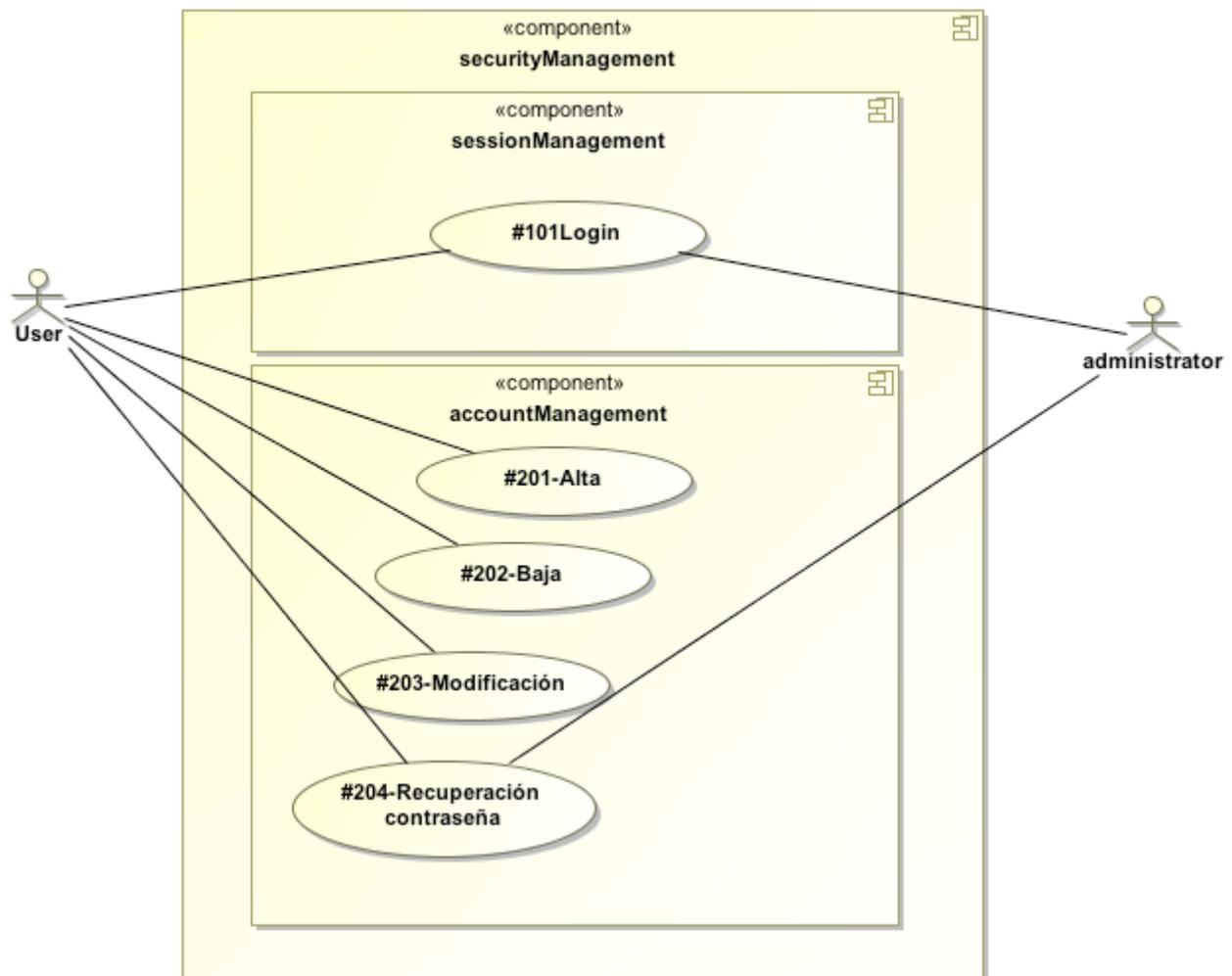


Diagrama de casos de uso de autenticación y gestión de cuentas

Login

Caso de uso: #101-Login

Dependencia de casos de uso: ninguno.

Actor principal: usuario, administrador

Ambito: Componente de sesión del sistema (sessionManagement).

Nivel de objetivo: usuario, administrador

Stakeholders e intereses:

Usuario: quiere poder entrar en el sistema.

Administrador: quiere poder entrar en el sistema.

Precondición: la cuenta debe existir para poder entrar en el sistema.

Garantías mínimas:

Garantías en caso de éxito: el usuario o el administrador pueden entrar en el sistema.

Escenario principal de éxito:

1. El usuario/administrador accede al sistema.
2. El sistema procesa la petición y valida la sesión. Comprueba que no existe ninguna. A continuación muestra un formulario a rellenar para acceder al sistema.
3. El usuario/administrador rellena el formulario con el nombre de usuario y password y lo envía al sistema.
4. El sistema procesa la petición, valida que el usuario existe en el sistema y que los datos son correctos. Seguidamente muestra la ventana principal de la aplicación.

Extensiones:

2a.- El sistema valida la sesión y comprueba que existe una creada previamente. El sistema muestra la ventana principal de la aplicación y termina el caso.

4a.- El sistema procesa la petición, valida los datos y muestra un mensaje indicando que los datos recibidos no son correctos. (Se vuelve al paso #3).

Alta de usuario

Caso de uso: #201-Alta de cuenta de usuario

Dependencia de casos de uso: ninguno

Actor principal: usuario

Ambito: Componente de gestión de cuentas del sistema (accountManagement).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder darse de alta en el sistema para poder usarlo.

Precondición: el usuario no debe existir en el sistema.

Garantías mínimas:

Garantías en caso de éxito: el sistema permitirá dar de alta un usuario.

Escenario principal de éxito:

5. El usuario accede al sistema y selecciona la opción de crear una cuenta.
6. El sistema procesa la petición y le muestra un formulario para indicar los datos de la cuenta.
7. El usuario rellena el formulario y se lo envía al sistema.
8. El sistema procesa la petición, valida que el usuario no existe en el sistema y muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

3a.- El usuario decide no rellenar el formulario y sale de la aplicación. No se crea la cuenta.

4a.- El sistema procesa la petición, valida que el usuario existe en el sistema y muestra un mensaje indicando tal circunstancia. (Se vuelve al paso #3).

Baja de cuenta de usuario

Caso de uso: #202-Baja de la cuenta de usuario

Dependencia de casos de uso: ninguno.

Actor principal: usuario

Ambito: Componente de gestión de cuentas del sistema (accountManagement).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder dar de baja su cuenta.

Precondición: la cuenta debe existir para poder ser dada de baja.

Garantías mínimas:

Garantías en caso de éxito: la cuenta de usuario se dará de baja.

Escenario principal de éxito:

1. El usuario accede al sistema y selecciona la opción dar de baja la cuenta.
2. El sistema procesa la petición y le muestra los datos actuales de la cuenta.
3. El usuario envía un petición de baja de la cuenta.
4. El sistema procesa la petición y muestra un mensaje indicando que se va a proceder a dar de baja a la cuenta
5. El usuario envía una petición al sistema confirmando que desea eliminar la cuenta.
6. El sistema procesa la petición, valida que el usuario existe en el sistema. Seguidamente muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

5a.- El usuario decide no dar de baja la cuenta. Se vuelve al paso #2.

Modificación de cuenta de usuario

Caso de uso: #203-Modificación de cuenta de usuario

Dependencia de casos de uso: ninguno.

Actor principal: usuario

Ambito: Componente de gestión de cuentas del sistema (accountManagement).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder modificar los datos de su cuenta.

Precondición: la cuenta debe existir para poder ser modificada

Garantías mínimas:

Garantías en caso de éxito: la cuenta de usuario se modificará.

Escenario principal de éxito:

1. El usuario accede al sistema y selecciona la opción modificar cuenta.
2. El sistema procesa la petición y le muestra un formulario con los datos actuales de la cuenta.
3. El usuario modifica el formulario y se lo envía al sistema.
4. El sistema procesa la petición, valida que el usuario existe en el sistema y que los datos son correctos. Seguidamente muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

3a.- El usuario decide no rellenar el formulario. Se termina el caso.

4a.- El sistema procesa la petición, valida los datos y muestra un mensaje indicando que los datos recibidos no son correctos. (Se vuelve al paso #3).

Recuperar contraseña

Caso de uso: #204-Recuperar contraseña

Dependencia de casos de uso: ninguno.

Actor principal: usuario, administrador

Ambito: Componente de gestión de cuentas del sistema (accountManagement).

Nivel de objetivo: usuario, administrador

Stakeholders e intereses:

Usuario: quiere poder recuperar su contraseña en caso de olvido.

Administrador: quiere poder recuperar su contraseña en caso de olvido.

Precondición: la cuenta debe existir para poder recuperar la contraseña.

Garantías mínimas:

Garantías en caso de éxito: se recuperará la contraseña.

Escenario principal de éxito:

1. El usuario/administrador accede al sistema y selecciona la opción recuperar contraseña.
2. El sistema procesa la petición y le muestra un formulario para rellenar el nombre de la cuenta.
3. El usuario rellena el formulario y se lo envía al sistema.
4. El sistema valida que existe la cuenta y muestra un mensaje indicando que se le envía la contraseña a la cuenta de correo del usuario.

Extensiones:

4a.- El sistema valida que no existe la cuenta y muestra un mensaje indicando que la cuenta de usuario no es válida. Se vuelve al paso #2.

Casos de uso: gestión de ofertas y demandas

A continuación se muestran los casos de uso de “grano fino” para los casos de uso de “grano grueso” gestión de ofertas y demandas:

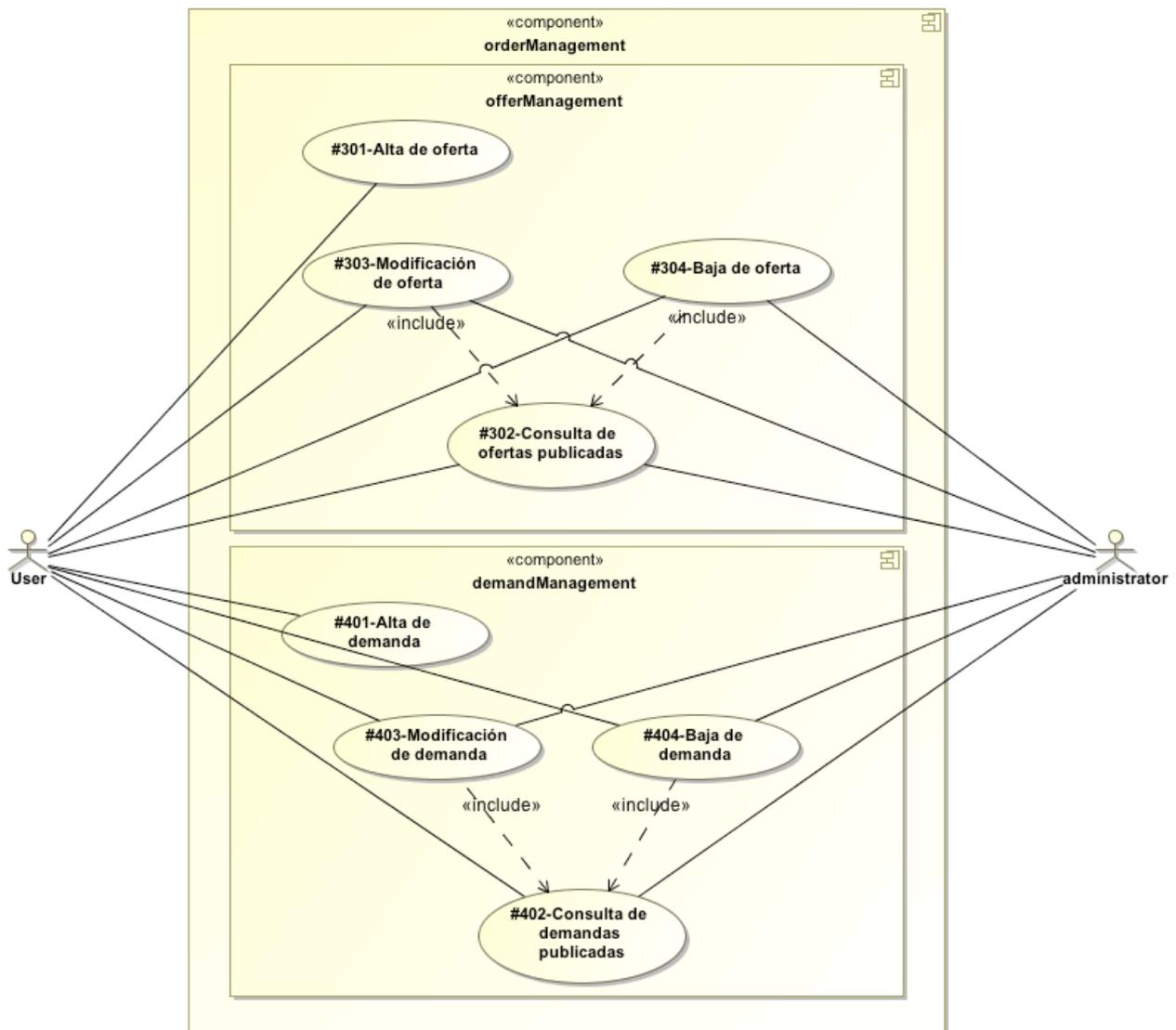


Diagrama de casos de uso gestión de ofertas y demandas

Alta de oferta

Caso de uso: #301-Alta de oferta

Dependencia de casos de uso: ninguno.

Actor principal: usuario

Ambito: Componente de gestión de ofertas del sistema (OfferManagement).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder dar de alta una oferta.

Precondición: el usuario debe estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario da de alta una oferta.

Escenario principal de éxito:

1. El usuario accede al sistema y selecciona la opción dar de alta una oferta.
2. El sistema procesa la petición y le muestra un formulario para indicar los datos de la oferta.
3. El usuario rellena el formulario y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

4a.- El sistema procesa la petición, valida los datos recibidos y muestra un mensaje indicando que los datos recibidos no son correctos. Se vuelve al paso #2.

Consulta de ofertas publicadas

Caso de uso: #302-Consulta de ofertas publicadas (por el mismo)

Dependencia de casos de uso: ninguno.

Actor principal: usuario/administrador

Ambito: Componente de gestión de ofertas del sistema (OfferManagement).

Nivel de objetivo: usuario/administrador

Stakeholders e intereses:

Usuario: quiere poder buscar sus ofertas publicadas.

Administrador: quiere poder buscar ofertas publicadas para controlar las publicaciones.

Precondición: el usuario/administrador debe estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario/administrador obtiene una la lista de ofertas publicadas.

Escenario principal de éxito:

1. El usuario/administrador accede al sistema y selecciona la opción mostrar ofertas publicadas.
2. El sistema procesa la petición y muestra una lista con las ultimas ofertas publicadas diferenciando si es el usuario o el administrador quien las solicita. Para el primero muestra las suyas propias, mientras que para el segundo muestra las de todos los usuarios. Además de mostrar las ofertas, el sistema muestra un formulario con los campos necesarios para que el usuario/administrador pueda realizar una búsqueda y filtrar las ofertas.
3. El usuario/administrador rellena el formulario para filtrar las ofertas y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra una lista con las ofertas encontradas. Se vuelve al paso #2.

Extensiones:

3a.- El usuario/administrador no realiza ninguna búsqueda y se finaliza el caso.

Modificación de oferta

Caso de uso: #303-Modificación de oferta

Dependencia de casos de uso: #302-consulta de ofertas publicadas

Actor principal: usuario/administrador

Ambito: Componente de gestión de ofertas del sistema (OfferManagement).

Nivel de objetivo: usuario/administrador

Stakeholders e intereses:

Usuario: quiere poder modificar sus ofertas publicadas.

Administrador: quiere poder modificar ofertas publicadas para el correcto funcionamiento del sistema.

Precondición: Se parte del listado de ofertas mostrado en el caso de uso de consulta de ofertas publicadas.

Garantías mínimas:

Garantías en caso de éxito: el usuario/administrador modifica los datos de una oferta.

Escenario principal de éxito:

1. El usuario/administrador selecciona una oferta publicada para modificarla y envía una petición al sistema para realizar dicha acción.
2. El sistema procesa la petición y le muestra un formulario para modificar los datos de la oferta.
3. El usuario/administrador rellena el formulario y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

4a.- El sistema procesa la petición, valida los datos recibidos y muestra un mensaje indicando que los datos recibidos no son correctos. Se vuelve al paso #2.

Baja de oferta

Caso de uso: #304-Baja de oferta

Dependencia de casos de uso: #302-consulta de ofertas publicadas

Actor principal: usuario/administrador

Ambito: Componente de gestión de ofertas del sistema (OfferManagement).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder dar de baja sus ofertas publicadas.

Administrador: quiere poder dar de baja ofertas publicadas para el correcto funcionamiento del sistema.

Precondición: Se parte del listado de ofertas mostrado en el caso de uso de consulta de ofertas publicadas.

Garantías mínimas:

Garantías en caso de éxito: el usuario/administrador da de baja una oferta.

Escenario principal de éxito:

1. El usuario/administrador selecciona una oferta publicada para darla de baja y envía una petición al sistema para realizar dicha acción.
2. El sistema procesa la petición y le muestra un formulario con los datos de la oferta.
3. El usuario/administrador envía un petición de baja de la cuenta.
4. El sistema procesa la petición y muestra un mensaje indicando que se va a proceder a dar de baja a la oferta.
5. El usuario/administrador envía una petición al sistema confirmando que desea eliminar la oferta.
6. El sistema procesa la petición. Seguidamente muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

5a.- El usuario/administrador decide no dar de baja la cuenta. Se vuelve al paso #2.

Casos de uso de demandas

Con el fin de no extender la especificación de los casos de uso y repetir lo redactado, los casos uso de las demandas #4xx se omitirán en su redacción, puesto que son exactamente iguales a los de las ofertas e indicarlos no aportan mas que lo que con estas lineas se indica. Aunque se omite su redacción, se deben tener en cuenta a la hora del diseño técnico y de la implementación del sistema. En cualquier caso se pueden ver que están especificados en las figuras aportadas y se deben de tener en consideración.

Casos de uso: suscripción de ofertas y demandas

A continuación se muestran los casos de uso de “grano fino” para los casos de uso de “grano grueso” suscripción de ofertas y demandas:

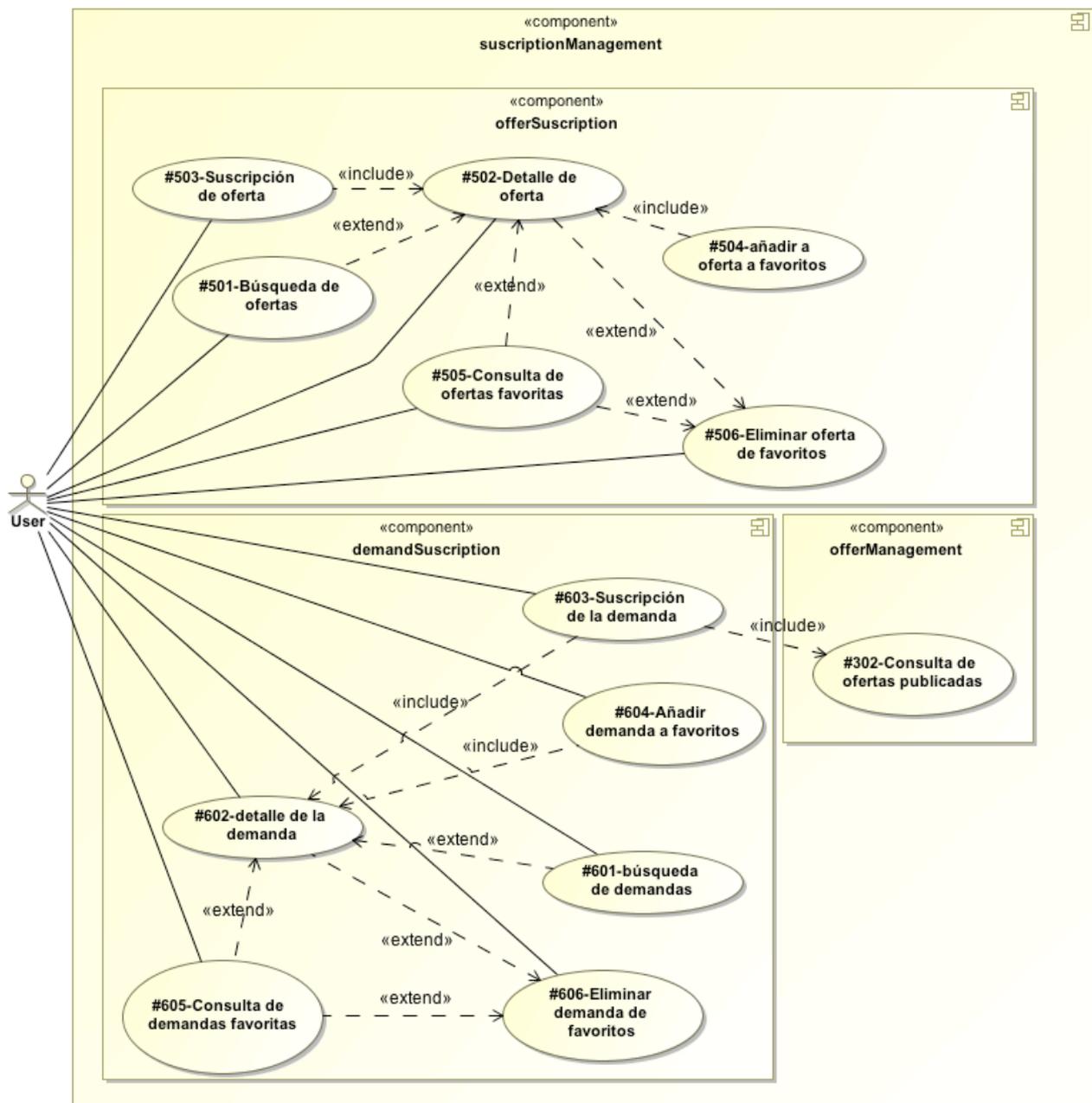


Diagrama de casos de uso suscripción de ofertas y demandas

Búsqueda de ofertas

Caso de uso: #501-Búsqueda de ofertas

Dependencia de casos de uso: ninguno

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder buscar las ofertas del sistema.

Precondición: ninguna.

Garantías mínimas:

Garantías en caso de éxito: el usuario busca las ofertas del sistema.

Escenario principal de éxito:

1. El usuario accede al sistema y selecciona la opción buscar ofertas publicadas.
2. El sistema procesa la petición y muestra una lista con las ultimas ofertas publicadas por los demás usuarios de la aplicación. Además de mostrar las ofertas, el sistema muestra un formulario con los campos necesarios para que el usuario pueda realizar una búsqueda y filtrar las ofertas.
3. El usuario rellena el formulario para filtrar las ofertas y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra una lista con las ofertas encontradas. Se vuelve al paso #2.

Extensiones:

3a.- El usuario no realiza ninguna búsqueda y se finaliza el caso.

Detalle de oferta

Caso de uso: #502-Detalle de oferta

Dependencia de casos de uso: #501-Búsqueda de ofertas, #505-Consulta de ofertas favoritas.

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder ver el detalle las ofertas del sistema.

Precondición: Se parte del listado de ofertas mostrado en el caso de uso de búsqueda de ofertas publicadas o de consulta de ofertas favoritas.

Garantías mínimas:

Garantías en caso de éxito: el usuario ve el detalle de las ofertas del sistema.

Escenario principal de éxito:

1. El usuario selecciona una oferta de la lista y envía una petición al sistema para que le muestre el detalle.
2. El sistema procesa la petición y le muestra un formulario con los datos de la oferta.

Subscripción de ofertas

Caso de uso: #503-Suscripción de ofertas

Dependencia de casos de uso: #501-Detalle de oferta, #302-consulta de ofertas publicadas.

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder suscribirse a las ofertas del sistema.

Precondición: Se parte del detalle de la oferta seleccionada en el caso de uso de Detalle de oferta. El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario se suscribe a las ofertas del sistema.

Escenario principal de éxito:

1. El usuario lee la oferta y desea suscribirse. Para ello envía una petición al sistema para realizar tal acción.
2. El sistema procesa la petición y muestra al usuario los datos de la suscripción: oferta y datos del usuario: direcciones y método de facturación.
3. El usuario selecciona las direcciones y el método de facturación que desea aplicar a la suscripción y se las envía al sistema.
4. El sistema procesa la petición, captura los datos y los valida. Seguidamente le muestra un mensaje indicando que se va a realizar la suscripción.
5. El usuario confirma que desea realizar la suscripción enviándole una petición al sistema.
6. El sistema procesa la petición y seguidamente muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

5a.- El usuario decide no subscribirse a la oferta porque no está seguro, y no confirma la operación. Se vuelve al paso #2.

6a.- El sistema procesa la petición y seguidamente muestra un mensaje de error en indicando que la operación se ha realizado con éxito, en el caso de que no funcione la pasarela de pago. Se vuelve al paso#2

Añadir ofertas a favoritos

Caso de uso: #504-Añadir ofertas a favoritos

Dependencia de casos de uso: #501-Detalle de oferta

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder añadir una oferta a su lista de ofertas favoritas a modo de recordatorio.

Precondición: Se parte del detalle de la oferta seleccionada en el caso de uso de Detalle de oferta. El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario se añade una oferta a su lista de favoritos.

Escenario principal de éxito:

1. El usuario lee la oferta y desea añadirla a favoritos. Para ello envía una petición al sistema para realizar tal acción.
2. El sistema procesa la petición y muestra un mensaje indicando que se ha añadido la oferta a la lista de favoritos.

Consulta de ofertas favoritas

Caso de uso: #505-Consulta de ofertas favoritas

Dependencia de casos de uso: ninguno.

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder consultar su lista de ofertas favoritas.

Precondición: El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario consulta su lista de ofertas favoritas.

Escenario principal de éxito:

1. El usuario entra en el sistema y selecciona la opción de ver su lista de ofertas favoritas.
2. El sistema muestra una lista con las ultimas ofertas favoritas. Además de mostrar las ofertas, el sistema muestra un formulario con los campos necesarios para que el usuario pueda realizar una búsqueda y filtrar las ofertas.
3. El usuario rellena el formulario para filtrar las ofertas y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra una lista con las ofertas encontradas. Se vuelve al paso #2.

Extensiones:

3a.- El usuario no realiza ninguna búsqueda y se finaliza el caso.

Eliminar ofertas de favoritos

Caso de uso: #506-Eliminar ofertas de favoritos

Dependencia de casos de uso: #502-Detalle de oferta, #505-Consulta de ofertas favoritas

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder eliminar una oferta a su lista de ofertas favoritas.

Precondición: Se parte del detalle de la oferta seleccionada en el caso de uso de Detalle de oferta, o se parte del listado de ofertas favoritas en el caso de consultar lista de ofertas favoritas. El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario se elimina una oferta a su lista de favoritos.

Escenario principal de éxito:

1. El desea eliminar una oferta de favoritos. Para ello, ya desde el detalle de la oferta, o desde el listado de ofertas favoritas, envía una petición indicando tal acción con la oferta seleccionada.
2. El sistema procesa la petición y muestra un mensaje indicando que se ha eliminado la oferta de la lista de favoritos.

Casos de uso de demandas

Con el fin de no extender la especificación de los casos de uso y repetir lo redactado, los casos uso de las demandas #6xx se omitirán en su redacción, excepto para el caso #603, puesto que son exactamente iguales a los de las ofertas e indicarlos no aportan mas que lo que con estas lineas se indica. Aunque se omite su redacción, se deben tener en cuenta a la hora del diseño técnico y de la implementación del sistema. En cualquier caso se pueden ver que están especificados en las figuras aportadas y se deben de tener en consideración.

Subscripción de demandas

Caso de uso: #503-Suscripción de ofertas

Dependencia de casos de uso: #501-Detalle de oferta, #302-Consulta de ofertas publicadas.

Actor principal: usuario

Ambito: Componente de suscripciones de ofertas del sistema (offerSubscription).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder suscribirse a las demandas del sistema.

Precondición: Se parte del detalle de la demanda seleccionada en el caso de uso de #602-Detalle de de la demanda. El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario se suscribe a las demandas del sistema.

Escenario principal de éxito:

1. El usuario lee la demanda que desea suscribirse. Para ello envía una petición al sistema para realizar tal acción.
2. El sistema le muestra un listado de ofertas publicadas por el mismo (caso de uso #302).
3. El usuario selecciona una oferta y envía una petición al sistema indicando la oferta seleccionada que desea suscribir a la demanda.
4. El sistema procesa la petición y muestra un mensaje indicando que se va a proceder a la suscripción de la oferta por parte del usuario.
5. El usuario envía una petición al sistema confirmando que desea suscribirse a la oferta.
6. El sistema procesa la petición. Seguidamente muestra un mensaje indicando que la operación se ha realizado con éxito.

Extensiones:

5a.- El usuario decide no suscribirse a la oferta. Se vuelve al paso #2.

Casos de uso: histórico de ofertas y demandas

A continuación se muestran los casos de uso de “grano fino” para los casos de uso de “grano grueso” histórico de ofertas y demandas:

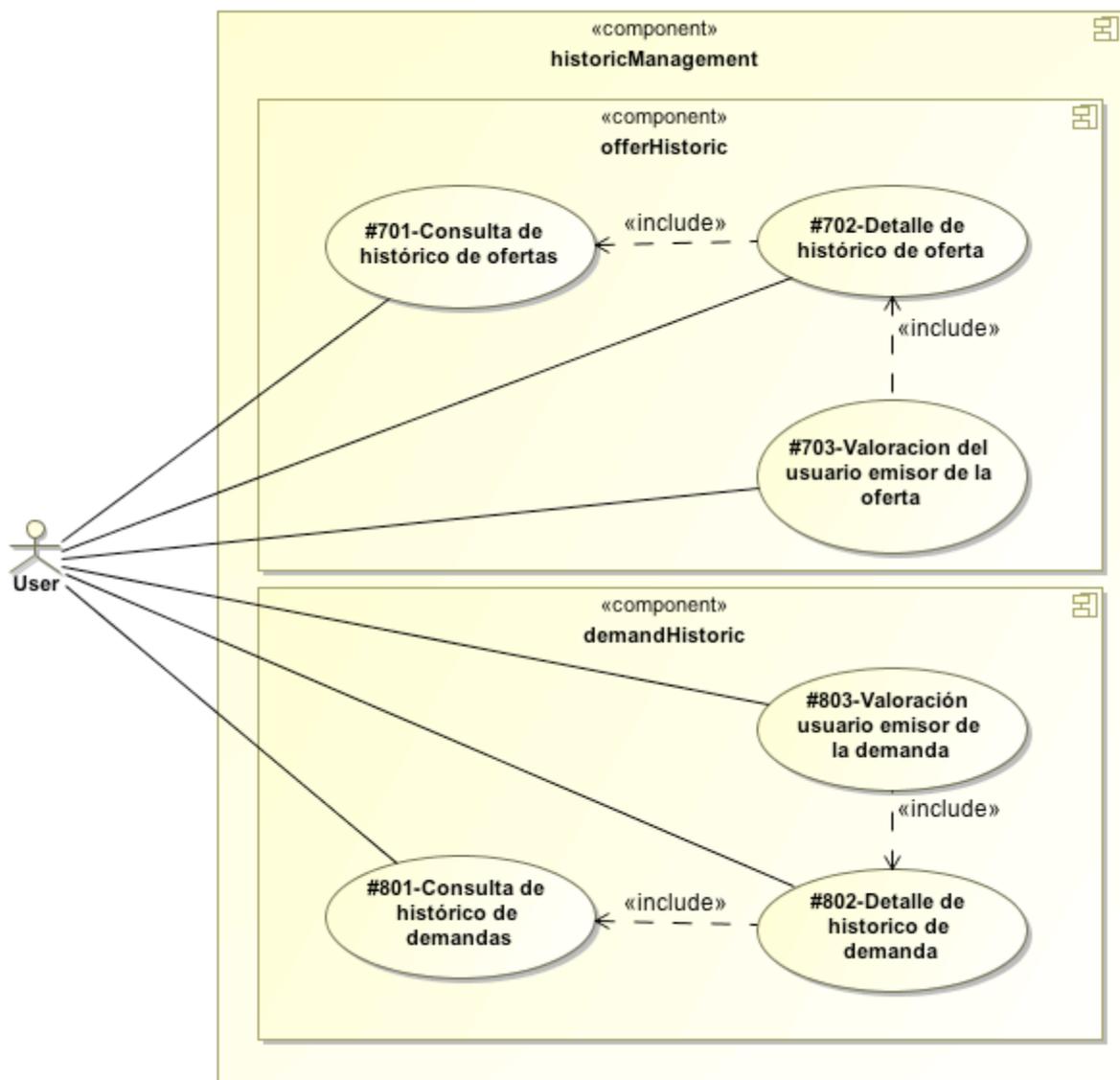


Diagrama de casos de uso histórico de ofertas y demandas

Consulta de histórico de ofertas

Caso de uso: #701-Consulta de histórico de ofertas

Dependencia de casos de uso: ninguno.

Actor principal: usuario

Ambito: Componente del histórico de ofertas suscritas por el usuario (offerHistoric).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder consultar su lista de ofertas suscritas.

Precondición: El usuario tiene que estar logado.

Garantías mínimas:

Garantías en caso de éxito: el usuario consulta su lista de ofertas suscritas.

Escenario principal de éxito:

1. El usuario entra en el sistema y selecciona la opción de ver su histórico de ofertas.
2. El sistema muestra una lista con las ultimas ofertas suscritas. Además de mostrar las ofertas, el sistema muestra un formulario con los campos necesarios para que el usuario pueda realizar una búsqueda y filtrar las ofertas.
3. El usuario rellena el formulario para filtrar las ofertas y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y muestra una lista con las ofertas encontradas. Se vuelve al paso #2.

Extensiones:

3a.- El usuario no realiza ninguna búsqueda y se finaliza el caso.

Detalle de histórico de la oferta

Caso de uso: #702-Detalle de histórico de la oferta

Dependencia de casos de uso: #701-Consulta de histórico de ofertas (subscritas).

Actor principal: usuario

Ambito: Componente del histórico de ofertas subscritas por el usuario (offerHistoric).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder ver el detalle la oferta subscrita.

Precondición: Se parte del listado de ofertas mostrado en el caso de uso de consulta de histórico de ofertas subscritas.

Garantías mínimas:

Garantías en caso de éxito: el usuario ve el detalle de las oferta subscrita.

Escenario principal de éxito:

1. El usuario selecciona una oferta de la lista y envía una petición al sistema para que le muestre el detalle.
2. El sistema procesa la petición y le muestra un formulario con los datos de la oferta.

Valoración del usuario emisor de la oferta

Caso de uso: #703-Valoración del usuario emisor de la oferta

Dependencia de casos de uso: #702-Detalle de histórico de la oferta (subscritas).

Actor principal: usuario

Ambito: Componente del histórico de ofertas subscritas por el usuario (offerHistoric).

Nivel de objetivo: usuario

Stakeholders e intereses:

Usuario: quiere poder publicar una opinión/valoración sobre la oferta subscrita.

Precondición: Se parte del detalle de una oferta del histórico.

Garantías mínimas:

Garantías en caso de éxito: el usuario publica una opinión del oferta subscrita.

Escenario principal de éxito:

1. Sobre el detalle de la oferta del histórico, el usuario decide publicar una opinión/valoración. Para ello envía una petición al sistema.
2. El sistema procesa la petición y le muestra un formulario para rellenar con la opinión/valoración del usuario.
3. El usuario rellena el formulario y se lo envía al sistema.
4. El sistema procesa la petición, valida los datos recibidos y guarda la valoración. Se vuelve al paso #2 por si el usuario decide cambiar la opinión.

Extensiones:

3a.- El usuario no realiza ninguna valoración y se finaliza el caso.

Casos de uso de demandas

Con el fin de no extender la especificación de los casos de uso y repetir lo redactado, los casos uso de las demandas #8xx se omitirán en su redacción, puesto que son exactamente iguales a los de las ofertas e indicarlos no aportan mas que lo que con estas lineas se indica. Aunque se omite su redacción, se deben tener en cuenta a la hora del diseño técnico y de la implementación del sistema. En cualquier caso se pueden ver que están especificados en las figuras aportadas y se deben de tener en consideración.

Diseño técnico

Una vez abordado el análisis del sistema, se pasa a especificar el diseño técnico del mismo. Principalmente abordaremos esta sección indicando el diagrama de clases y el modelo entidad del sistema, como piezas mas importantes del diseño del sistema.

Indicar que como decisión técnica se va a usar Hibernate como ORM, por lo que las clases contendrán un identificador que se utilizará como campo que actúa de enlace con la columna de la PK en las tablas que representan. Es hartos sabido, que las claves compuestas, en los modelos entidad relación, aportan complejidad en las consultas y mas bajo rendimiento del motor de base de datos al tener que realizar “árboles sintácticos¹” mas complejos. Se tienen que crear indices mas complejos y al final todo son problemas tanto de uso del modelo entidad relación. Hibernate, consciente de este problema, nos sugiere que usemos como claves primaria un campo simple de tipo integer o long². Siguiendo este principio nuestras clases seguirán esta norma.

Comentar que los nombres utilizados en las clases han sido definidos de tal forma que no sea necesario documentar su función, con lo que, echando un vistazo al diagrama de clases queda claro cual es su función dentro del sistema. Los beneficios que esto aporta son principalmente claridad a la hora de leer el diagrama, claridad a la hora de programar y claridad a la hora mantener el sistema, ya que la documentación de los sistemas no siempre está a mano del que lo necesita.

¹ véase REF#02

² véase REF#01 apartado “Advanced: Composite_Primary_Keys”

Arquitectura del sistema

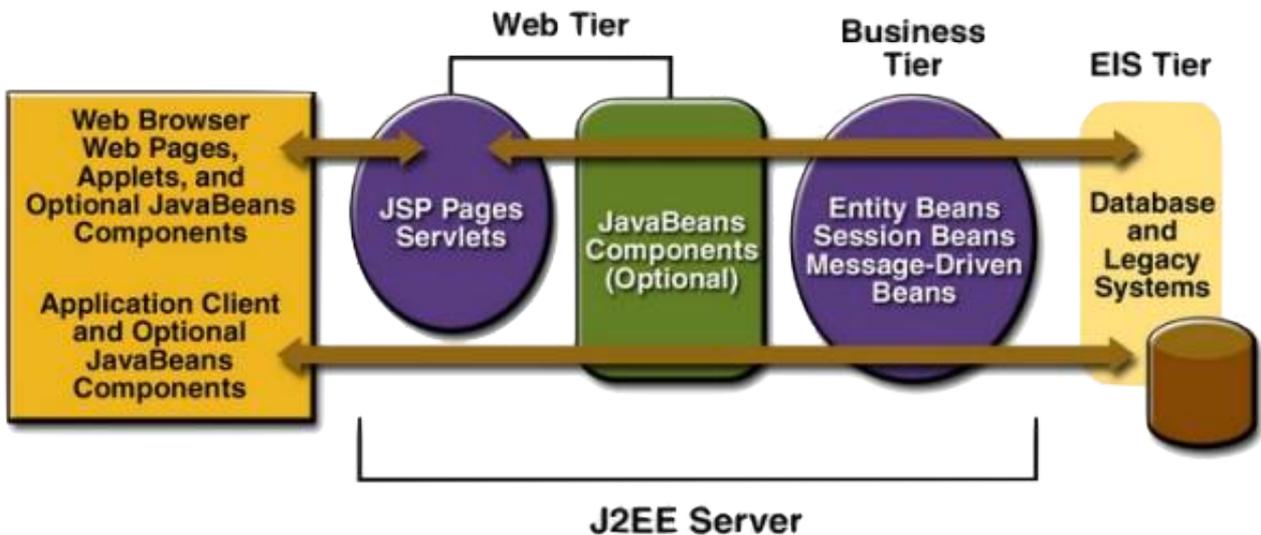
La arquitectura del sistema será una arquitectura cliente-servidor de 3 capas:

- ✦ Cliente ligero: navegador del cliente.
- ✦ Servidor de aplicaciones J2EE: procesamiento de la aplicación. Usaremos el servidor JBOSS en su versión 7.1.1
- ✦ Servidor de base de datos: administración de datos. Usaremos el SGBD PostgreSQL en su versión 9.2

En cuanto a la representación de la arquitectura software, utilizaremos el patrón de estilo arquitectónico MVC para realizar la división lógica en capas: Modelo-Vista-Controlador

- ✦ Capa presentación: esta capa contendrá todos los elementos relacionados con la gestión de la presentación o “frontend”. Se utilizará el API de JSF como tecnología para implementar esta capa.
- ✦ Capa de dominio: esta capa contendrá todos los elementos relacionados con la lógica de negocio de la aplicación. Se utilizarán EJB's para modelar la capa de dominio o “backend”.
- ✦ Capa de servicios técnicos: esta capa contendrá todos los elementos relacionados con el modelo. Se utilizarán POJO's como entities de Hibernate para modelar la capa de servicios técnicos del sistema. Se utilizará el API de JPA como tecnología para implementar esta capa.

A grosso modo la arquitectura de nuestro sistema quedaría como sigue:



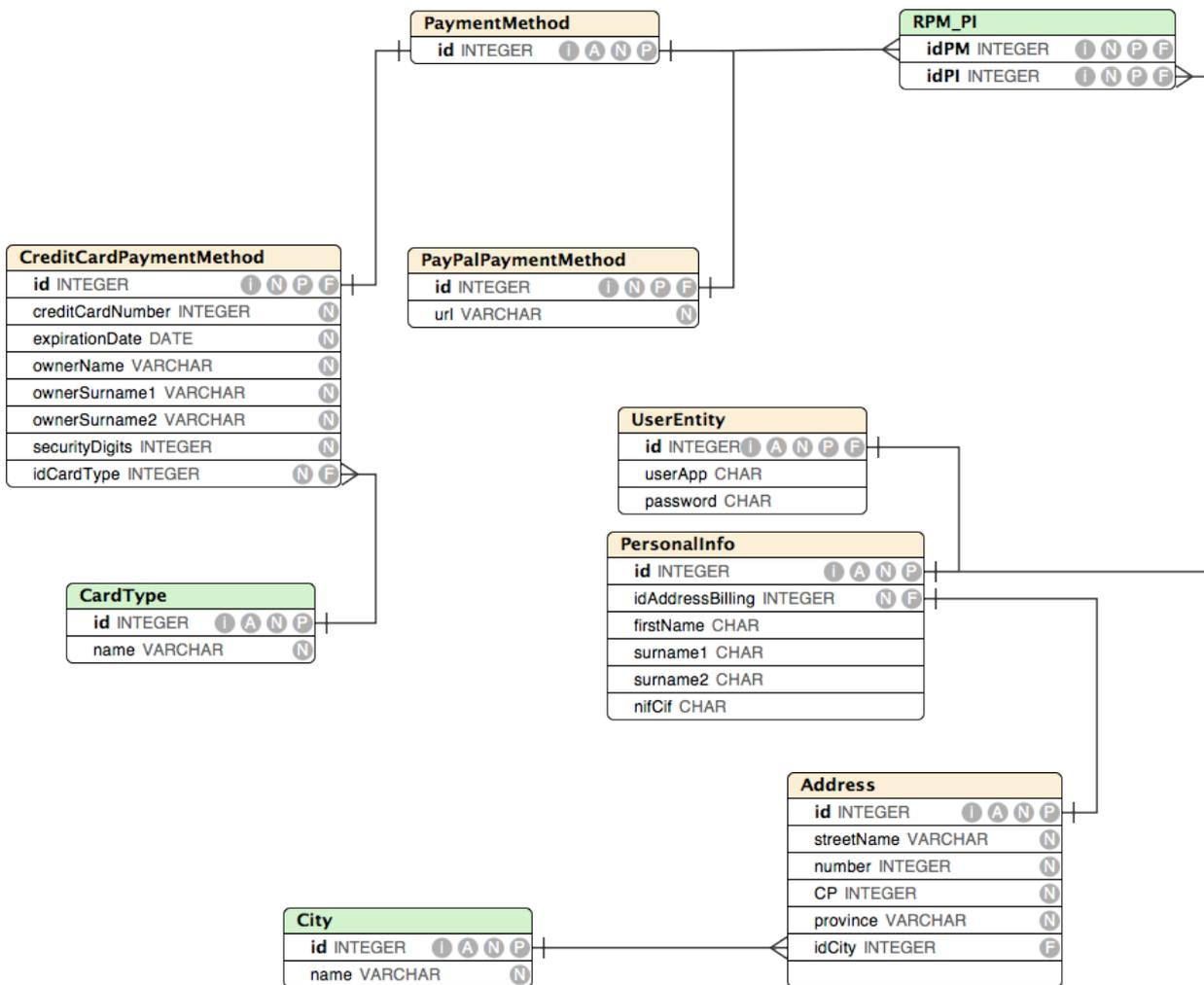
Arquitectura J2EE

En cuanto a las dependencias entre capas se ha de tener en cuenta lo siguiente:

- ♦ La capa mas inferior, o capa de presentación, tendrá una dependencia de la capa del dominio. Ningún elemento de la capa de servicios técnicos estará disponible para la capa de presentación.
- ♦ La capa de dominio solo tendrá dependencia de la capa de servicios técnicos.
- ♦ La capa de servicios técnicos no tendrá ninguna dependencia con respecto a las capas anteriores.

Diagrama entidad Relación

Basándonos en el diagrama de clases, el modelo entidad relación quedaría como sigue:



Modelo entidad relación de la parte de datos de usuario y seguridad

Diagrama de clases: servicios

El diseño de los servicios es similar en todos los casos, así pues se expondrá el caso del servicio de gestión de cuentas de usuario, a fin de ilustrar un caso.

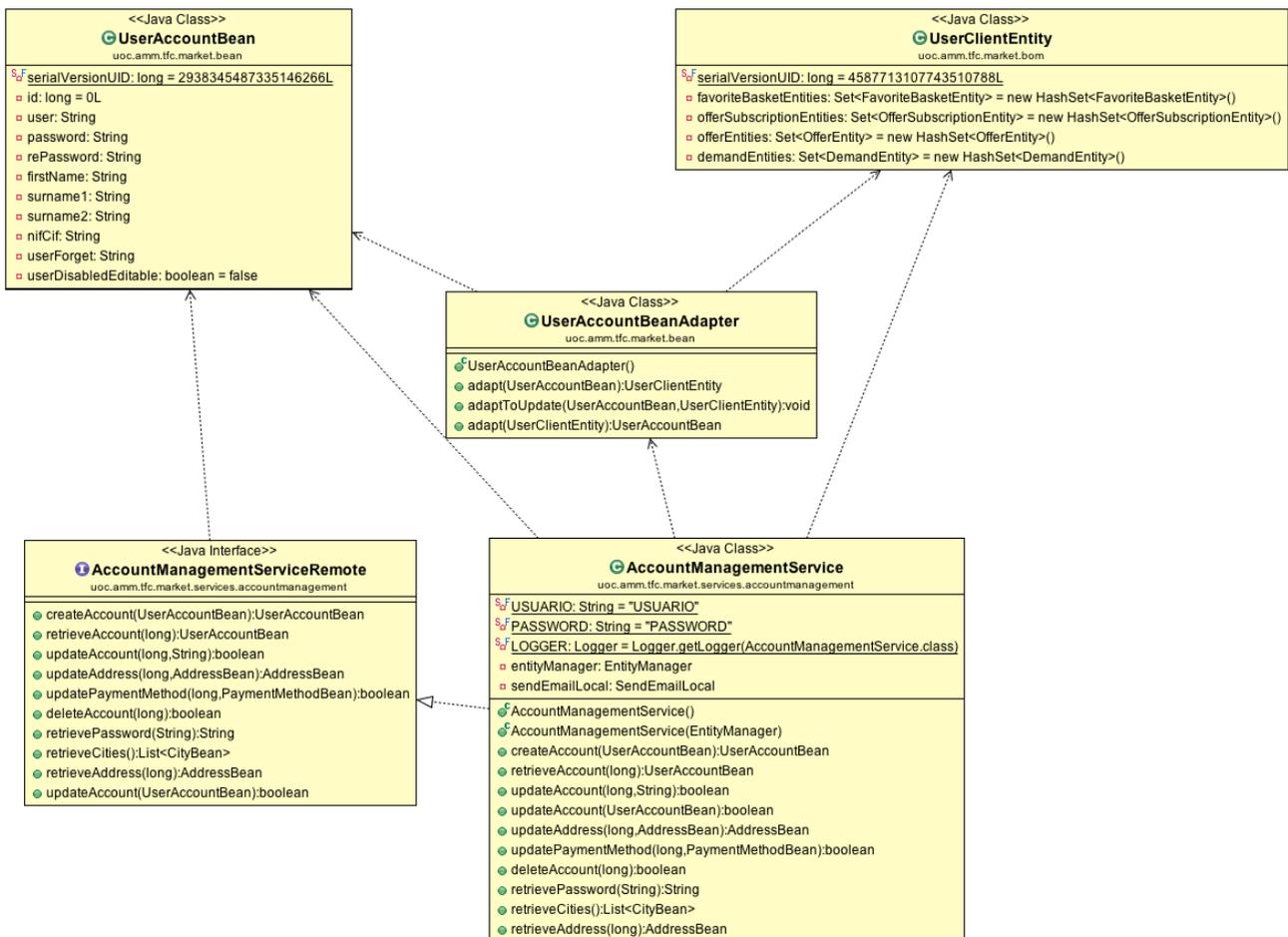


Diagrama de clases del servicio de gestión de cuentas del usuario

Los servicios implementados se corresponden con la funcionalidad de los casos de usos expuestos en los apartados anteriores. Los nombres de los servicios y los paquetes se corresponden con los usados en los diagramas de caso de uso.

En cualquier caso, el diseño está basado en un EJB y su interfaz Remota. El EJB usa el API de JPA para obtener los datos de la base de datos haciendo uso de los entities. El resultado es adaptado a un objeto que hemos denominado Bean que sirve a la capa de presentación para construir la vista.

Diagrama de clases: front-end

El diseño de los controladores es similar en todos los casos, así pues se expondrá el caso del controlador de gestión de cuentas de usuario, a fin de ilustrar un caso:

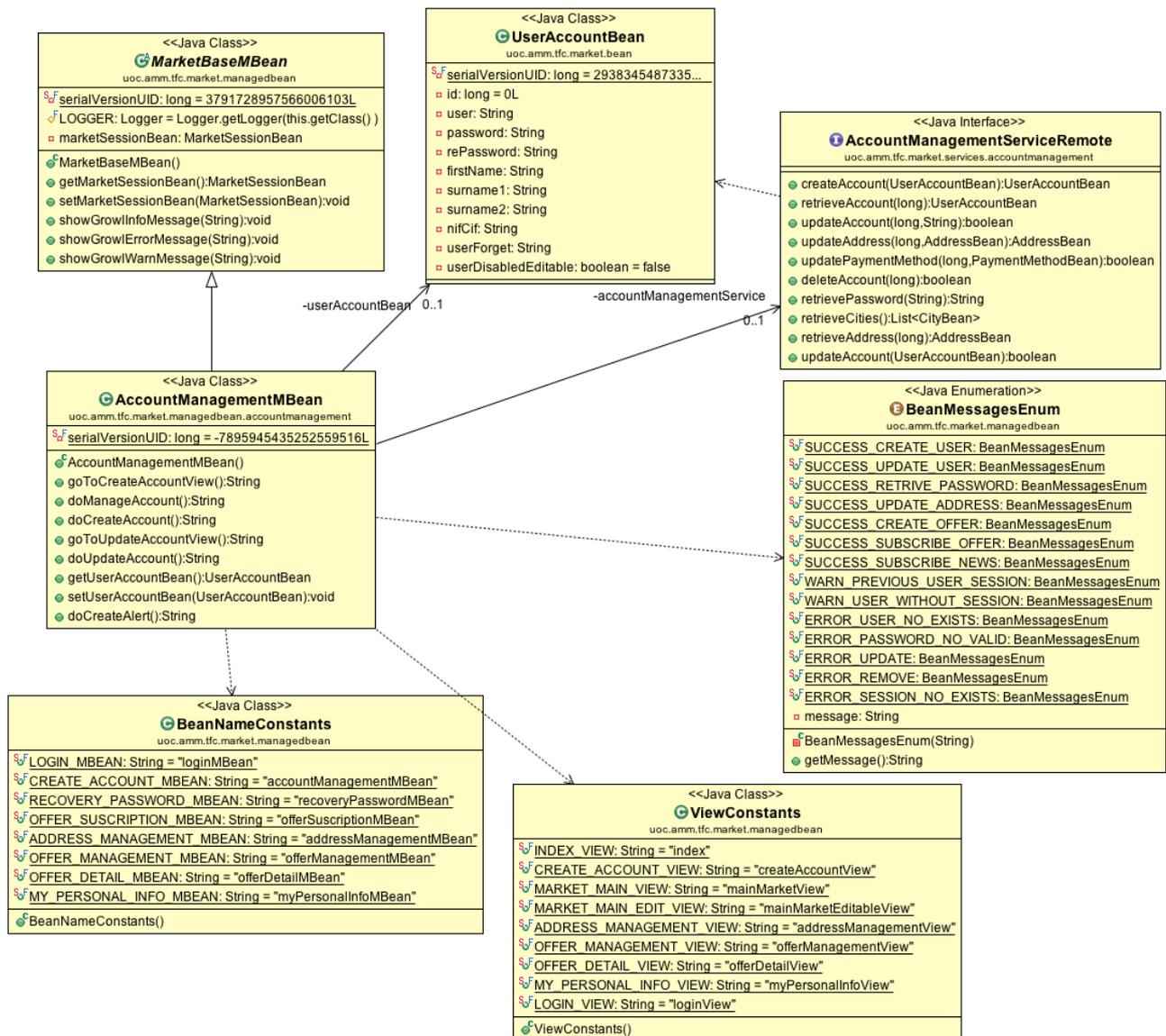


Diagrama de clases de la capa de presentación para el controlador o managedBean de gestión de cuentas de usuario

La estrategia habitual seguida para el desarrollo será “un controlador por cada vista, y un servicio para cada controlador.”, aunque esta estrategia puede variar por temas de navegación y reutilización de código.

En cualquier caso, el diseño está basado en un Controlador o managedBean que recibe la petición de una vista. Este solicita al EJB el servicio que necesita. El retorno de la llamada al EJB será un Bean que se utilizará para construir la vista que se sirve al navegador.

Especificaciones

Para construir la aplicación se utilizarán los siguientes componentes:

- ✦ Se utilizará el Api de JPA v2
- ✦ Se utilizará Hibernate v4 como ORM
- ✦ Se utilizará maven con herramienta de gestión de dependencias y generación de artefactos.
- ✦ Se utilizará EJB v3
- ✦ Se utilizará Java J2SE v1.6
- ✦ Se utilizará Java J2EE v1.6
- ✦ Se utilizará JSF v2 y Primefaces v5

Servidores

- ✦ Se utilizará JBoss v7.1.1 como servidor de aplicaciones J2EE
- ✦ Se utilizará postgresSQL v9.2 como servidor de base de datos

Herramientas de desarrollo:

- ✦ Se utilizará Eclipse Juno version J2EE
- ✦ SQLEditor como editor de diagrama entidad relación
- ✦ Magic Draw como editor de diagramas de clases UML

Entregables software

Para construir la aplicación se utilizará:

- ♦ Se entregará el desplegable de la aplicación: market.EAR.
- ♦ Se entregaran los scripts de generación de las tablas del sistema: creacionTablas.sql
- ♦ Se entregará un fichero con la configuración del servidor JBoss que contiene el datasource utilizado en el proyecto y el mail-session para que se puedan enviar correos electrónicos. El fichero se ubicará en la carpeta de instalación de JBoss, en el directorio \$directorioInstalacionJBoss/standalone/configuration. El fichero se denominará standalone.xml

Conclusiones

Llevo años trabajando como desarrollador y me apetecía probar algo nuevo, innovador y diferente, con respecto a lo que suelo hacer en el trabajo diario. He disfrutado utilizando herramientas nuevas, que de otro modo, seguro que no utilizaría. El trabajo del día a día, todo el desarrollo está controlado y gestionado. Este TFG ha sido un soplo de aire fresco con respecto a lo que conocía. He aprendido a usar herramientas y tecnologías nuevas, y además de eso, me ha servido para probarme o medirme, para demostrarme a mi mismo que puedo hacer un proyecto por mi mismo. Desde cero. Ha sido una experiencia gratificante y espero que me sirva como iniciativa y punto de partida para futuros proyectos personales.

Bibliografía y referencias

REF#01 - http://en.wikibooks.org/wiki/Java_Persistence/Identity_and_Sequencing#Composite_Primary_Keys

REF#02 - Material oficial de referencia de la asignatura de Sistemas Gestores de Base de Datos 3, código 81031

REF#03 - Material oficial de referencia de la asignatura de Ingeniería del software de componentes y sistemas distribuidos, código 75.587

REF#04 - Material oficial de referencia de la asignatura de Introducción a la ingeniería de requisitos, código 75.593

REF#05 - Material oficial de referencia de la asignatura de Análisis y diseño de patrones, código 75.586