UNIVERSIDAD SAN FRANCISCO DE QUITO

PROYECTO FINAL - DISEÑO DE SISTEMAS

INTEGRANTES: CÉSAR CÁRDENAS, NICOLÁS MOINA, ALEX PÉREZ.

**Actores del sistema de pedidos de pizza:**

1. Cliente
2. Administrador
3. Servidor Principal (AuctionInterface)
4. Servidor de Subasta (por producto, AuctionServer)
5. Base de Datos (Singleton)

**Descripciones textuales de Casos de Uso**

1. **Se inicializa el servidor principal. Se despliega el view de Servidor Principal junto con información y datos sobre el proceso de subastas. Se mantiene conexión con las subastas disponibles y se reporta su estado.**

Nombre: Inicio del Servidor Principal

Flujo Normal:

* El administrador ejecuta el archivo AuctionInterface.java, encargado del servidor principal. Se despliega el view de Servidor Principal.
* El Servidor Principal establece conexión con la base de datos (singleton) y muestra en el view lo siguiente: número de clientes conectados, subastas disponibles (ligadas a los ítems/productos subastados y almacenados originalmente en la base de datos/singleton), servidores de subastas (con el botón INICIAR SERVIDOR), el estado de las subastas disponibles (servidor no iniciado, subasta iniciada, subasta terminada). Se actualiza el número de clientes conectados, en caso de que los haya.
* Tras iniciar el servidor de una subasta, haciendo click en el botón INICIAR SERVIDOR, se accede al view de Servidor de Subasta. El estado de la subasta pasa de “servidor no iniciado” a “subasta iniciada”.
* Tras finalizar el servidor de una subasta, luego de que el item/producto haya sido vendido a la mejor oferta, se muestra un pop-up/view de información: Subasta Terminada. El estado de la subasta pasa de “subasta iniciada” a “subasta terminada”.

Flujo Alternativo:

* Se ejecuta el archivo AuctionInterface.java, encargado del servidor principal. Se despliega el view de Servidor Principal.
  + Existe un error con las configuraciones de ejecución y no es posible realizar esta acción, por lo que no se inicializa el servidor principal y se muestran pop-ups/views de notificaciones de error de ejecución dadas por JavaIOExceptions.
* El Servidor Principal establece conexión con la base de datos (singleton) y muestra en el view lo siguiente: número de clientes conectados, subastas disponibles (ligadas a los ítems/productos subastados y almacenados originalmente en la base de datos/singleton), servidores de subastas (con el botón INICIAR SERVIDOR), el estado de las subastas disponibles (servidor no iniciado, subasta iniciada, subasta terminada). Se actualiza el número de clientes conectados, en caso de que los haya.
  + El singleton originalmente no tiene datos sobre ningún item/producto a subastar, por lo que el view de Servidor principal se mostrará vacío e incompleto.
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados se muestra como 0 en el view de Servidor Principal.
* Tras iniciar el servidor de una subasta, haciendo click en el botón INICIAR SERVIDOR, se accede al view de Servidor de Subasta. El estado de la subasta pasa de “servidor no iniciado” a “subasta iniciada”.
  + No es posible dar click en el botón de INICIAR SERVIDOR pues ya ha sido iniciado, junto con la subasta. Esto se verifica al ver el estado de la subasta (subasta iniciada).
  + No es posible dar click en el botón de INICIAR SERVIDOR pues ya ha terminado, junto con la subasta. Esto se verifica al ver el estado de la subasta (subasta terminada).
* Tras finalizar el servidor de una subasta, luego de que el item/producto haya sido vendido a la mejor oferta, se muestra un pop-up/view de información: Subasta Terminada. El estado de la subasta pasa de “subasta iniciada” a “subasta terminada”.

1. **Se inicializa el Servidor de Subasta. Se muestra el view Servidor de Subasta del item/producto respectivo, donde se despliega información del producto junto con el botón que da inicio al periodo de subasta.**

Nombre: Inicio del Servidor de Subastas

Flujo Normal:

* El servidor de subastas establece conexión con la base de datos (singleton) para mostrar información del item/producto subastado en el view Servidor de Subasta: nombre del item/producto, precio base, puerto del servidor (único también), el botón de INICIAR SUBASTA, un temporizador (cuyo estado inicial es “aún no iniciado”) y un log de progreso de la subasta que se actualizará con las ofertas de los clientes suscritos una vez que la subasta haya sido iniciada.
* Al hacer click en el botón INICIAR SUBASTA se permite que la subasta por el item/producto respectivo empiece, por lo que los clientes conectados podrán empezar a ofertar hasta que se termine el tiempo de la subasta. se cierra el view de Servidor de Subasta.
* Una vez terminada la subasta se muestra el pop-up/view de Subasta Terminada. En este view se notifica que la subasta ha terminado, se da información del cliente que ha ganado la subasta (junto con la cantidad ofertada y a pagar) y se muestra el botón de OK.
* Al hacer click en el botón de OK, se cierra el view Servidor de Subasta y se actualiza el view del Servidor principal, en el estado de la subasta, que pasa de subasta iniciada -> subasta terminada.

Flujo Alternativo:

* El servidor de subastas establece conexión con la base de datos (singleton) para mostrar información del item/producto subastado en el view Servidor de Subasta: nombre del item/producto, precio base, puerto del servidor (único también), el botón de INICIAR SUBASTA, un temporizador (cuyo estado inicial es “aún no iniciado”) y un log de progreso de la subasta que se actualizará con las ofertas de los clientes suscritos una vez que la subasta haya sido iniciada.
  + El singleton originalmente no tiene datos sobre ningún item/producto a subastar, por lo que el view de Servidor de Subasta se mostrará vacío e incompleto.
* Al hacer click en el botón INICIAR SUBASTA se permite que la subasta por el item/producto respectivo empiece, por lo que los clientes conectados podrán empezar a ofertar hasta que se termine el tiempo de la subasta.
  + Al no tener datos sobre ningún item/producto a subastar, los clientes no se podrán suscribir/conectar al servidor del item/producto respectivo y se muestra el pop-up/view de Error: No existen ítems/productos para subastar.
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados es de 0 y se mostrará un view de Servidor de Subasta donde el temporizador esté corriendo, pero no existirán registros/historial de ofertas hechas por el item/producto subastado.
* Una vez terminada la subasta se muestra el pop-up/view de Subasta Terminada. En este view se notifica que la subasta ha terminado, se da información del cliente que ha ganado la subasta (junto con la cantidad ofertada y a pagar) y se muestra el botón de OK.
  + No se ha realizado la inicialización de ningún cliente, por lo que el view de Subasta Terminada se mostrará incompleto y no se tendrá la información del cliente que ha ganado la subasta, pues no se han hecho ofertas.
* Al hacer click en el botón de OK, se cierra el view Servidor de Subasta y se actualiza el view del Servidor principal, en el estado de la subasta, que pasa de subasta iniciada -> subasta terminada.

1. **Se agrega un cliente con su respectivo ID (único) en el view Cliente (AuctionClientInterface). Se redirige al view de Cliente ID donde, tras conectarse con la base de datos (singleton), se muestran los ítems/productos subastados.**

Nombre: Creación de Clientes y oferta en subastas.

Flujo Normal:

* El cliente ejecuta el archivo AuctionClienteInterface.java, encargado de la creación de clientes. Se despliega el view de Cliente.
* Se establece conexión con la base de datos (singleton) y se muestra en el view de Cliente una petición de ingresar su ID (string) de cliente. Además, se muestran los botones de ACEPTAR y CANCELAR.
* Tras agregar un cliente, haciendo click en el botón de ACEPTAR, se muestra el view de Cliente ID donde se despliega una lista de los ítems/productos subastados con: nombre, precio y estado de la subasta (cuyos estados pueden ser: no iniciado, en progreso, terminado).
* Se muestra además un botón de ENTRAR A LA SUBASTA que al hacer click sobre el mismo se establece conexión con el puerto del servidor del item/producto por el cual se está ofertando (conexión con AuctionServer, Servidor de Subasta, que a su vez se conecta con AuctionInterface, Servidor Principal). Una vez establecida la conexión, se muestra el view de Cliente Subasta donde se tiene información del item/producto por el cual se está ofertando: nombre, precio base, puerto del servidor (con el que se estableció conexión y que es único del item/producto).
* Se muestra en el view de Cliente Subasta el botón de HACER OFERTA junto con tres botones de cantidades: 10 000, 100 000, 1 000 000 y el precio actual por el cual se está subastando el item/producto respectivo al puerto único. Es necesario hacer click en un botón de alguna de estas cantidades para poder hacer click en el botón de HACER OFERTA. Al hacer una oferta por el producto respectivo, el view de Cliente Subasta también muestra un log/registro del historial de ofertas hechas por el o los clientes conectados al puerto del servidor único para el item/producto respectivo.
* Se muestra en el view de cliente subasta un temporizador en cuenta regresiva que muestra el tiempo restante para la subasta del item/producto respectivo. Cada vez que un cliente hace click en HACER OFERTA este temporizador se actualiza en el o los views de Cliente Subasta (dependiendo de cuantos clientes estén conectados/suscritos al puerto del servidor único del item/producto) y en el view de Servidor Subasta.
* Una vez que la subasta haya terminado, que el temporizador haya llegado hasta 0, se cierra/termina la conexión al puerto del servidor y se muestra el pop-up/view de Subasta terminada. En este view se notifica a todos los clientes que estaban conectados/suscritos a la subasta del item/producto respectivo que la subasta ha terminado y se muestra el botón de OK.
* Al hacer click en el botón de OK, se cierra el view de Cliente Subasta y se actualiza el view de Cliente ID. La actualización se hace en el apartado de subasta item -> estado -> terminado.

Flujo Alternativo:

* Se ejecuta el archivo AuctionClienteInterface.java, encargado de la creación de clientes. Se despliega el view de Cliente.
  + Existe un error con las configuraciones de ejecución y no es posible realizar esta acción, por lo que no crea o agrega ningún cliente y se muestran pop-ups/views de notificaciones de error de ejecución dadas por JavaIOExceptions.
* Se establece conexión con la base de datos (singleton) y se muestra en el view de Cliente una petición de ingresar su ID (string) de cliente. Además, se muestran los botones de ACEPTAR y CANCELAR.
  + No se ingresó un ID de cliente (se dejó vacío) por lo que al intentar entrar en la subasta, se mostrará un view de Excepción diciendo “Ingresar un ID de cliente válido”.
* Tras agregar un cliente, haciendo click en el botón de ACEPTAR, se muestra el view de Cliente ID donde se despliega una lista de los ítems/productos subastados con: nombre, precio y estado de la subasta (cuyos estados pueden ser: no iniciado, en progreso, terminado).
  + El singleton originalmente no tiene datos sobre ningún item/producto a subastar, por lo que el view de Cliente ID se mostrará vacío e incompleto.
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados es de 0 y no se mostrará el view de Cliente ID.
* Se muestra además un botón de ENTRAR A LA SUBASTA que al hacer click sobre el mismo se establece conexión con el puerto del servidor del item/producto por el cual se está ofertando (conexión con AuctionServer, Servidor de Subasta, que a su vez se conecta con AuctionInterface, Servidor Principal). Una vez establecida la conexión, se muestra el view de Cliente Subasta donde se tiene información del item/producto por el cual se está ofertando: nombre, precio base, puerto del servidor (con el que se estableció conexión y que es único del item/producto).
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados es de 0 y no se mostrará un view de Cliente Subasta y no se podrán realizar ofertas.
* Se muestra en el view de Cliente Subasta el botón de HACER OFERTA junto con tres botones de cantidades: 10 000, 100 000, 1 000 000 y el precio actual por el cual se está subastando el item/producto respectivo al puerto único. Es necesario hacer click en un botón de alguna de estas cantidades para poder hacer click en el botón de HACER OFERTA. Al hacer una oferta por el producto respectivo, el view de Cliente Subasta también muestra un log/registro del historial de ofertas hechas por el o los clientes conectados al puerto del servidor único para el item/producto respectivo.
* Se muestra en el view de cliente subasta un temporizador en cuenta regresiva que muestra el tiempo restante para la subasta del item/producto respectivo. Cada vez que un cliente hace click en HACER OFERTA este temporizador se actualiza en el o los views de Cliente Subasta (dependiendo de cuantos clientes estén conectados/suscritos al puerto del servidor único del item/producto) y en el view de Servidor Subasta.
* Una vez que la subasta haya terminado, que el temporizador haya llegado hasta 0, se cierra/termina la conexión al puerto del servidor y se muestra el pop-up/view de Subasta terminada. En este view se notifica a todos los clientes que estaban conectados/suscritos a la subasta del item/producto respectivo que la subasta ha terminado y se muestra el botón de OK.
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados es de 0 y no se mostrará el view de subasta terminada para ningún cliente.
* Al hacer click en el botón de OK, se cierra el view de Cliente Subasta y se actualiza el view de Cliente ID. La actualización se hace en el apartado de subasta item -> estado -> terminado.
  + No se ha realizado la inicialización de ningún cliente, por lo que el número de clientes conectados es de 0 y no se actualizará el view de Cliente ID.