
 Universidad de los Andes Facultad de Ingeniería	Ingeniería de Sistemas y Computación Pregrado ISIS2304 – Sistemas Transaccionales http://sistemas.uniandes.edu.co/~isis2304 Semestre: 2014-1 Horario: Miércoles y Viernes – 15:30 a 16:50	
---	---	---

ITERACIÓN 2 - DISEÑO E IMPLANTACIÓN DE UNA APLICACIÓN TRANSACCIONAL

OBJETIVOS

- Realizar el proceso de diseño de una aplicación de características transaccionales, que utiliza un sistema manejador de bases de datos relacional.
- Desarrollar los elementos de arquitectura de software que permiten integrar un SMBD relacional con un ambiente de objetos Java
- Incorporar elementos de calidad del modelo de datos, con respecto a integridad de la información.

PRERREQUISITOS

- El taller se realiza siguiendo los lineamientos indicados en el documento de “Herramientas, recursos, logística de trabajo y evaluación” y el documento de reglas de juego del proyecto del semestre, disponibles en Sicua+.
- Disponibilidad de los recursos computacionales de desarrollo del curso.
- Documento de descripción del caso de estudio.

HERRAMIENTAS

- Oracle 10g instalado en el servidor `prod.oracle.virtual.uniandes.edu.co`
- SQL Developer, disponible en los laboratorios del Departamento.
- Para la documentación UML del modelo se recomienda la utilización de *Enterprise Architect*, disponible en los laboratorios.
- Ejemplo resuelto, disponible en la wiki del curso (sección Ejemplo Resuelto).
- Plantilla de desarrollo de aplicaciones Web, disponible en Sicua+.

REQUERIMIENTOS SOBRE EL AMBIENTE DE DESARROLLO

Los requerimientos descritos a continuación se complementan con los indicados en el enunciado del caso de estudio. Constituyen requerimientos independientes del caso de estudio a desarrollar.

- Lenguaje de programación: Java 1.5 o superior
- La interfaz de la aplicación debe ser Web, utilizando servlets. Si utiliza un programa diferente, debe informarlo al inicio del taller. En cualquier caso, debe ser un ambiente disponible en los laboratorios del Departamento.
- Sistema de bases de datos: Oracle 10g.
- La plataforma de ejecución es JBoss, en la versión disponible en los laboratorios. Si utiliza una tecnología diferente debe informarlo al inicio del taller. En cualquier caso, debe ser un ambiente disponible en los laboratorios del Departamento.
- La conexión a la base de datos debe hacerse a través de JDBC. Esto permite desacoplar la aplicación del sistema de base de datos específico.

ENUNCIADO

RECOMENDACIONES

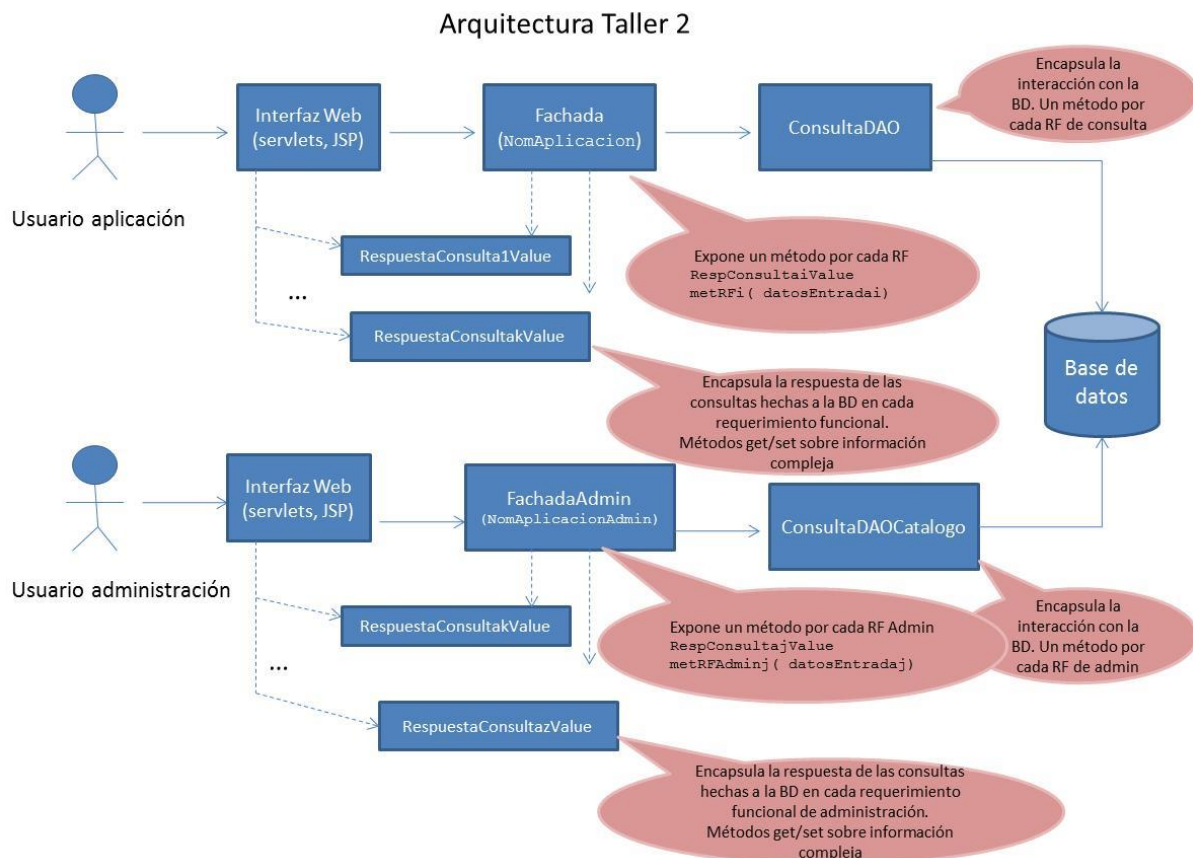
1. Revise el caso de estudio en su totalidad. El caso de estudio presenta el contexto particular sobre el cual se trabaja este taller y, de forma incremental, los talleres siguientes. Revise que comprende la terminología, las reglas de negocio y los servicios que allí se describen.
2. Tenga en cuenta el manejo de usuarios en la arquitectura, modelo y documentación.

CASO DE ESTUDIO

El taller se desarrolla utilizando el caso “**Central de Abastos de Los Andes**”. Los documentos de descripción del caso se encuentran en Sicua+.

ARQUITECTURA DE LA APLICACIÓN

La aplicación debe seguir la siguiente arquitectura.



REQUERIMIENTOS NO INTERACTIVOS

RB1. Cargue la información básica de arranque el sistema, poblando las tablas **a partir de archivos** para los tipos de producto, productos, bodegas, locales de venta, usuarios, pedidos a proveedores, propuestas de proveedores.

REQUERIMIENTOS INTERACTIVOS

- Los siguientes requerimientos **DEBEN** ser realizados utilizando **el menor número de sentencias SQL posibles**.
- Se debe minimizar la posibilidad de ingreso de datos incorrectos por parte del usuario. Por ejemplo, **las solicitudes de productos hechas por un local de productos perecederos pueden seleccionar los productos dinámicamente**

según los productos perecederos que se encuentran registrados en el sistema y los pedidos pendientes de un usuario se seleccionan entre aquellos que aparecen registrados a su nombre.

- En esta iteración NO es necesario desarrollar autenticación ni registro de usuarios.

REQUERIMIENTOS FUNCIONALES DE CONSULTA

RF1. Consultar las existencias de productos en CabAndes, clasificados según un criterio deseado por quien realiza la consulta (usuario o administración de CabAndes). Los filtros son producto, tipo de producto, presentación, fecha de expiración, identificador de bodega, identificador de local de ventas. Debe ofrecerse la posibilidad de ordenamiento de las respuestas según los intereses del usuario que consulta.

RF2. Consultar los pedidos de un usuario. Deben poder ser filtrados por satisfechos, pendientes de entrega, en cierto rango de fechas, costo, (producto, presentación). Se debe mostrar la fecha, los productos y cantidades solicitadas y el costo.

REQUERIMIENTOS FUNCIONALES DE MODIFICACIÓN

RF3. Registrar movimiento de producto. Por facilidad de uso, deben poder seleccionarse los productos, presentaciones y demás información para los diferentes casos especificados dentro de las existentes.

RF4. Registrar/satisfacer pedido. Por facilidad de uso, el sistema debe ofrecer la información para que el usuario pueda hacer su pedido de manera ágil y eficiente.

RF5. Escoger proveedor para una licitación de productos. Por facilidad de uso, el sistema sólo debe ofrecer la información de las licitaciones que ya están cerradas pero aún sin asignar.

REQUERIMIENTOS NO FUNCIONALES

RNF1. PERSISTENCIA: La información registrada en el sistema debe mantenerse en el tiempo.

RNF2. DISTRIBUCIÓN: La aplicación se puede utilizar desde cualquier sitio vía web. **Para esta iteración, la base de datos de la aplicación está centralizada.**

BONO

RF6. Consultar el (los) producto(s) con mayor movimiento en el sistema, dado un rango de fechas. El resultado debe incluir la información de los productos, presentación, el promedio de peso, el costo promedio, la cantidad de veces que ha sido solicitado, la cantidad de veces que ha sido vendido. El resultado debe estar ordenado por identificador del tipo de producto y nombre del producto.

RF7. Obtener todos los datos del puesto de ventas que ha realizado el mayor número de ventas de un cierto tipo de producto o producto específico.

ESCENARIOS DE PRUEBA

Desarrolle los escenarios de prueba que le permiten asegurar el correcto funcionamiento de la aplicación y la corrección y calidad de los datos en la base de datos.

En particular, para probar la corrección y calidad del modelo en la base de datos, incluya las siguientes pruebas:

1. Pruebas de unicidad de tuplas. Para cada tabla,
 - a. Inserte una tupla 1 con una PK conocida y nueva
 - b. Inserte una tupla 2, con la misma PK que la tupla 1
 - c. Haga las pruebas de resultados de la inserción del primer registro y del segundo registro.
2. Pruebas de integridad con FK
 - a. Inserte una tupla 1 que tenga una FK que se encuentra en la tabla referenciada
 - b. Inserte una tupla 1 que tenga una FK que no se encuentra en la tabla referenciada
 - c. Haga las pruebas de inserción para cada caso
 - d. Haga pruebas de borrado de tuplas maestras y dependientes.

3. Pruebas de integridad de acuerdo con restricciones de chequeo
 - a. Inserte tuplas que cumplen con las restricciones de chequeo establecidas
 - b. Inserte tuplas que violan las restricciones de chequeo establecidas
 - c. Haga las pruebas de inserción y borrado correspondientes.

ACTIVIDADES A DESARROLLAR: CONSTRUCCIÓN DE LA APLICACIÓN

1. (5%) Cree en Oracle las tablas correspondientes al modelo de datos.
2. (5%) Poble las tablas con información suficiente para poder realizar pruebas.
3. (25%) Desarrolle las clases correspondientes al caso de estudio. Para el desarrollo, siga la arquitectura de la aplicación que se muestra en este documento. Esto significa la clase fachada, todas las clases XXXValue y las respectivas clases `ConsultaDAO`. En la plantilla de aplicación adjunta a este enunciado, encuentra un ejemplo de la clase `ConsultaDAO`. El desarrollo de los requerimientos funcionales tiene ponderación uniforme.
4. (20%) Desarrolle los servlets correspondientes a la interfaz de la aplicación.
5. Automatice los procesos asociados al desarrollo y despliegue utilizando `Ant`. Haga el despliegue de la aplicación en JBoss.
6. (25%) Ejecute los casos de uso desde un navegador de internet. Verifique el estado de la base de datos utilizando el cliente Oracle para Windows.
7. (20%) Implemente los escenarios de prueba descritos en la sección escenarios de prueba de este documento, para verificar el manejo de los datos persistentes.

ENTREGA

Fecha y hora límite: **9 de marzo de 2014, 22:00**

Archivo de la entrega:

- Archivo comprimido en formato .zip. Nombre del archivo: **<Taller2_NN_login1_login2>.zip**. NN es el número del grupo y login1 y 2 son los correspondientes a los integrantes del grupo.
- Contenido del archivo:
 - Proyecto Java con el código fuente, la documentación del proyecto y el bono si fue desarrollado. La documentación del proyecto debe quedar en una carpeta `docs`, en la estructura principal de directorios del proyecto java. Los documentos que entregue deben estar en formato .pdf.
 - Archivo `integrantes.txt`, con los nombres y códigos de estudiante de los integrantes del grupo
 - Archivo `readme.txt`, con todos los pasos a seguir para instalar la aplicación. La calificación del proyecto no incluye necesariamente una sustentación. Este archivo debe describir la forma de crear los ejecutables de la aplicación y de ejecutar la aplicación. Incluya un ejemplo para obtener resultados al ejecutar cada uno de los requerimientos funcionales implementados en esta entrega.
- El no seguimiento del formato de entrega del taller tiene una penalización de 0.5/5.0 puntos en la nota final, **por cada elemento faltante**.

DOCUMENTACIÓN

Tutoriales de acceso a los servicios de Oracle: <http://sistemas.uniandes.edu.co/~isis2304>

Documentación en línea de Oracle 10g: <http://sistemas.virtual.uniandes.edu.co/~manuales/oracle10g/>