
 Universidad de los Andes Facultad de Ingeniería	Ingeniería de Sistemas y Computación Pregrado ISIS2304 – Sistemas Transaccionales	
---	--	---

PROYECTO DEL CURSO – ENTREGA 2 DEL PROYECTO

OBJETIVOS

- Desarrollar habilidades en el proceso de generación de un modelo de datos relacional a partir de un modelo conceptual
- Incorporar elementos de calidad del modelo de datos, con respecto a integridad de la información.
- Implementar una aplicación informática de mediana complejidad que involucre bases de datos relacionales.

HERRAMIENTAS

- Para la documentación del modelo UML puede utilizar GenMyModel u otro
- Para la documentación del modelo E/R puede utilizar Oracle DataModeler
- Para la documentación del modelo relacional debe utilizar la plantilla Excel dispuesta en BN
- Oracle 12c disponible en la infraestructura computacional del Departamento
- SQL Developer, disponible en los laboratorios del Departamento y disponible para descarga en el sitio Web de Oracle
- Debe usar el IDE y framework de desarrollo indicados en el proyecto guiado de Coursera

CASO DE ESTUDIO

La iteración se desarrolla utilizando el caso **Banco de los Andes**, que se encuentra en Bloque Neón (BN) - Contenidos.

ARQUITECTURA DE LA APLICACIÓN

Disponible en la documentación y archivos de base disponibles en BN.

ACTIVIDADES A DESARROLLAR: DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN

Revise el caso de estudio propuesto. Identifique los elementos fundamentales que hacen parte del negocio que se describe: entidades de negocio, funcionalidades principales, y reglas de negocio.

1. **(15%)** Análisis y revisión del modelo de datos. Analicen la retroalimentación dada por parte del monitor en la Entrega 1 - diseño. A partir de esto:
 - a. Ajusten el modelo conceptual UML e indiquen cuáles clases del modelo son actualizadas o creadas en esta entrega.
 - b. Actualicen el modelo relacional de la BD usando las reglas de transformación vistas en clase. Este debe quedar documentado en el formato Excel disponible para ello en BN.
 - c. Verifique que cada relación del modelo esté en la Forma Normal de Boyce-Codd (BCNF). De no estarlo, realice los ajustes necesarios hasta cumplir con esta forma normal y documente dichos cambios.
 - d. Anexe la documentación y modelos de este entregable.
2. **(15%)** Diseño de la base de datos
 - e. Utilizando SQL Developer, cree en Oracle las tablas correspondientes al modelo de datos, con sus restricciones de integridad. Para ello, utilicen archivos de scripts con las sentencias SQL correspondientes (archivos .sql, generados por SQL Developer). Siga los estándares de nombramiento y buenas prácticas de definición de restricciones.

- f. Anexe a los entregables los archivos .sql utilizados.
3. **(63%)¹** Desarrolle una aplicación usando Java Spring y siguiendo la arquitectura de aplicación de ejemplo (ver proyecto guiado de coursera) que implemente los requerimientos funcionales de modificación (RF1 a RF9) y los requerimientos de consulta (RFC1 a RFC3) descritos en el documento marco. Para ello, desarrolle las clases correspondientes a:
 - a. La conexión a la base de datos
 - b. La lógica para la implementación de los RF. Estas deben contener las sentencias SQL de inserción, eliminación y modificación de datos necesarias para la implementación de los RF.
 - c. Las solicitudes SQL para las consultas descritas en los RFC.
 - d. Las interfaces de usuario correspondientes.

Evaluación detallada: 5% para cada requerimiento funcional de modificación (RF1 a RF9) y 6% para cada requerimiento funcional de consulta (RFC1 a RFC3)
4. **(2%)** Población de Tablas: pueblen las tablas con datos suficientes a través de las interfaces para poder verificar los requerimientos funcionales y los requerimientos funcionales de consulta.
5. **(5%)** Escenarios de prueba: realice las inserciones descritas abajo para poder realizar pruebas para los RF. Ver sección “Escenarios de prueba” de este documento.
 - a. Implemente los escenarios de prueba para cada requerimiento funcional (ver secciones de Requerimientos funcionales del documento marco del caso de estudio (RF1-RF6)), con el fin de verificar la persistencia de los datos y el manejo de las restricciones de integridad por parte el SMBD.
 - b. Incluya la traza de la ejecución de las pruebas (scripts de inserción y pantallazos).
 - c. Anexe los scripts SQL para estas operaciones y pantallazos de lo obtenido.

ESCENARIOS DE PRUEBA

Diseñe y desarrolle los escenarios de prueba que le permiten asegurar el correcto funcionamiento de la aplicación, así como la corrección y calidad de los datos en la base de datos.

A continuación, damos algunos ejemplos para probar la corrección y calidad del modelo en la base de datos:

1. Pruebas de unicidad de tuplas: para una de las tablas, por ejemplo, Oficinas o Usuarios, trate de insertar una tupla 1 con una PK nueva. Enseguida trate de insertar una tupla 2, con la misma PK que la tupla 1.
2. Pruebas de integridad referencial con FK: trate de insertar una tupla 1 en una tabla (por ejemplo, la tabla Cuentas), que tenga una FK cuyo valor se encuentra previamente en la tabla referenciada (por ejemplo, en la tabla Clientes, ya que toda cuenta pertenece a un cliente). De igual manera trate de insertar una tupla 1 que tenga una FK que no se encuentra en la tabla referenciada. Haga las pruebas de inserción para cada caso.
3. Pruebas de integridad de acuerdo con restricciones de chequeo: trate de insertar tuplas que violan las restricciones de chequeo de valores establecidas.

Para probar la corrección de la aplicación, incluya pruebas habituales de funcionalidad que revisen las reglas de negocio. No olvide considerar los roles de usuario, casos exitosos, y casos de falla.

ENTREGA

Fecha y hora límite: **17 de marzo de 2024**

- En la actividad correspondiente en BN, envíe la URL del *commit* de la entrega en el repositorio GitHub asignado a su grupo.
- Contenido de la entrega (*commit* de GitHub):
 - Documento de informe, debe estar en formato .pdf. Debe incluir:

¹ La aplicación debe tener interfaces que permitan al usuario interactuar con los datos desde un navegador web (ver proyecto guiado de coursera). A través de estas interfaces se debe poder poblar la BD y modificar los datos (RFs) y acceder a las consultas (RFCs)..

- Análisis y revisión del modelo de datos: modelos UML y relacional y su descripción, indicando las modificaciones hechas. Comprobación de la forma normal de cada relación.
 - Diseño de la base de datos: archivos .sql con los scripts utilizados para la creación del a BD
 - Documentación de las clases de Java Spring y de la arquitectura de la aplicación
 - Traza de la ejecución del plan de pruebas (scripts de inserción y pantallazos). Balance del plan de pruebas (que se logró hacer y qué no).
 - Proyecto Java con el código fuente y la documentación del proyecto de software descrita antes.
 - La documentación del proyecto debe quedar en una carpeta `docs`, en la estructura principal de directorios del proyecto java.
 - Archivo `readme.txt`, con todos los pasos a seguir para instalar la aplicación.
- El no seguimiento de los lineamientos de formato de entrega tiene una penalización de 0.5/5.0 puntos en la nota, por cada elemento faltante.

DOCUMENTACIÓN

Tutoriales de acceso a los servicios de Oracle en el sitio Web del curso

Documentación en línea de Oracle 12c: <http://www.oracle.com/technetwork/database/enterprise-edition/documentation/index.html>