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

ENTREGA 2 – OPTIMIZACIÓN DE CONSULTAS

OBJETIVOS

Integrar requerimientos de eficiencia a una aplicación transaccional desarrollada en una arquitectura de tres niveles, en una aplicación cliente/servidor y manejo de persistencia en base de datos relacionales.

PRERREQUISITOS

- Haber desarrollado los requerimientos funcionales de la entrega 1.
- Disponibilidad de los recursos computacionales de desarrollo del curso.
- Documento de descripción del caso de estudio.

HERRAMIENTAS

- 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.
- SQL Loader como herramienta sugerida para el cargue masivo de datos.
- Debe usar el IDE y framework de desarrollo indicados en el proyecto guiado de Coursera.

CASO DE ESTUDIO

La entrega se desarrolla utilizando el caso **Hotel de los Andes**, que se encuentra en Bloque Neón (BN) - Contenidos/Proyecto.

ARQUITECTURA DE LA APLICACIÓN

La definida en la entrega 1.

RECOMENDACIONES

- ✓ Revisen el caso de estudio de las entregas 1 y 2 en su totalidad, pues presenta el contexto particular sobre el cual se trabaja el proyecto de forma incremental. Revisen que comprenden bien la terminología, las reglas de negocio y los servicios que allí se describen.
- ✓ Tengan en cuenta el manejo de usuarios en la arquitectura, el modelo y en la documentación.

ACTIVIDADES A DESARROLLAR: DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN






(5 %) Análisis

A partir del diseño existente, analicen el impacto que representa la introducción de los nuevos requerimientos a nivel del modelo conceptual (diagrama de clases UML) propuesto en la entrega anterior. Tengan también en cuenta los comentarios recibidos por parte de los monitores en la sustentación de los talleres anteriores. Así, ajusten el modelo conceptual si se requiere e indiquen cuáles clases del modelo son actualizadas o creadas en esta entrega. Una vez hecho esto, actualicen el modelo relacional de la BD usando las reglas de transformación vistas en clase.



(30 %) Diseño de la aplicación

Analicen la aplicación completa resultante de la entrega anterior y los nuevos requerimientos para realizar el diseño físico correspondiente. En particular, diseñen los índices necesarios para el adecuado rendimiento global de la aplicación.

❖ (12%) Índices.

- Para cada uno de los requerimientos funcionales:
 -  Identifiquen si es necesario crear un índice.
 -  Justifiquen la selección del índice creado desde el punto de vista de la selectividad (concepto visto en clase). En caso de que para algún requerimiento funcional Uds determinen que un índice no es necesario, justifiquenlo también desde el punto de vista de la selectividad.
 -  Indiquen claramente cuál es el tipo de índice utilizado (B+, Hash, ..., primario, secundario).
- Según su modelo de datos, para los índices creados de forma automática por Oracle
 -  Incluyan una captura de pantalla con la información generada por Oracle asociada a los índices existentes.
 -  Analicen los índices encontrados. Específicamente, Analicen por qué fueron creados por Oracle y si ayudan al rendimiento de los requerimientos funcionales.

❖ (18%) Diseño de las consultas: para cada requerimiento funcional de consulta solicitado, documente:

- Sentencias SQL con la consulta o consultas que responden el requerimiento y que fueron analizadas. Anexe en el entregable un archivo en texto plano (.sql), con las sentencias SQL utilizadas.
- Distribución de los datos con respecto a los parámetros de entrada utilizados en el requerimiento funcional. En particular se quiere un análisis de distribución que permita ver cómo puede cambiar el tamaño de la respuesta según el valor de los parámetros utilizados y la configuración de los datos de prueba.
- Valores de los parámetros utilizados en el análisis y que constituyen diferenciadores en los planes de ejecución obtenidos.
- Planes de consulta obtenidos en Oracle para la ejecución de la consulta o consultas que implementan el requerimiento. Para ello, Documenten con una foto de pantalla los planes de consulta obtenidos en SQLDeveloper.
- Tiempos obtenidos con la ejecución de cada uno de los planes. Estos tiempos son medidos desde el núcleo de la aplicación, es decir, no incluyen la parte de interacción con el usuario, ingreso de datos ni despliegue de resultados.
- Análisis de eficiencia
 - Para cada requerimiento funcional, seleccione un escenario de análisis y diseñe un plan de ejecución de consulta propuesto por el grupo, de acuerdo con su conocimiento del modelo y de la aplicación.
 - Compare y Analicen el plan de ejecución propuesto por usted y el obtenido en Oracle.
 -  El porcentaje de evaluación correspondiente a cada uno de los requerimientos solicitados es proporcional al número de los requerimientos.
 -  La nota para cada uno de los requerimientos depende de los escenarios de ejecución definidos.

(25%) Diseño y cargue masivo de datos

- Ajuste las tablas creadas en Oracle de acuerdo a las decisiones del punto anterior.
- (20%) **Diseño de datos.** Cargue de datos necesarios para hacer el estudio de eficiencia de la aplicación.

- Diseñe los datos que le permitan verificar adecuadamente las reglas de negocio. Note que es importante generar adecuadamente los datos y para esta entrega lo es también el obtener un número muy grande de ellos.
- 👉 Se debe generar un volumen de datos tal que algunas tablas no quepan en la memoria principal de la máquina.
- 👉 Usualmente, alrededor de 750.000 registros en su base de datos puede ser adecuado. Demasiados datos sobre este valor pueden colapsar la infraestructura, que es compartida y se considera una mala práctica. Haga pruebas que le permitan experimentar con un valor adecuado.
- Puede escribir un programa de generación automática de datos acorde al diseño establecido para los mismos.
- Para la población de las tablas puede utilizar herramientas de carga masiva como, por ejemplo, `SQLLoader` o las disponibles en `SQLDeveloper`. Consulte el tutorial disponible en la página del curso sobre `SQLLoader`. También pueden optar por escribir un script para cargar datos en un lenguaje de programación.
- (5%) Documenten claramente el proceso de carga de datos: cómo fue realizado, cómo logró el volumen de datos solicitado, ...

(40%) Construcción de la aplicación, ejecución de pruebas y análisis de resultados

- Implemente en la aplicación que viene desarrollando desde la entrega 1, los requerimientos funcionales de consulta (RFC1 a RFC12) solicitados en el documento marco (debe tener una interfaz grafica sobre la cual se puedan realizar las consultas de los requerimientos solicitados). Para ello, utilice las tecnologías y frameworks que se vienen usando desde la entrega1.
- Implemente los escenarios de prueba que verifiquen el funcionamiento correcto de los requerimientos funcionales de consulta en la aplicación (RFC1 a RFC12), así como el aseguramiento de la calidad de los datos en la BD y los requerimientos no funcionales (RNF1-RNF3). No olvide considerar los roles de usuario, casos exitosos, casos de falla y casos de borde. Para probar la eficiencia de las operaciones, incluya las instrucciones de reloj correspondientes al iniciar y al finalizar la consulta.
- Todos los requerimientos tienen el mismo valor en la evaluación.

Completar todos los requerimientos funcionales de la primera entrega dará un bono de hasta el 0.5/5.0 sobre la nota de la segunda entrega.

ENTREGA

Fecha y hora límite: **5 de noviembre de 2023**

- 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 todos los entregables solicitados. El nombre del archivo debe seguir el formato de nombramiento `<Entrega1_L-NN_login1_login2_login3>.pdf`.
 - Proyecto Java con el código fuente y la documentación del proyecto de software.
 - La documentación del proyecto debe quedar en una carpeta `docs`, en la estructura principal de directorios del proyecto java.
 - En la carpeta `docs` del proyecto Java deben estar:
 - Los modelos desarrollados, tanto en el formato fuente como en `pdf`.
 - La documentación de los requerimientos funcionales.
 - Los archivos de scripts, sentencias y pruebas indicados en el enunciado (`.sql`).
 - 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>