

## PROYECTO DEL CURSO - ITERACIÓN 2

### OBJETIVOS

- Implementar una aplicación informática de mediana complejidad que involucre bases de datos relacionales, siguiendo la arquitectura de referencia definida para el curso.

### METODOLOGÍA

- La iteración 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 Brightspace.

### HERRAMIENTAS

- Para la documentación UML del modelo se recomienda la utilización de diagrams.net (<https://www.diagrams.net/>). En los laboratorios del departamento también está disponible *Enterprise Architect*, en caso que tengan acceso.
- Oracle 19c o superior 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
- Ejemplo completo desarrollado por los profesores del curso – Parranderos-jdo

### CASO DE ESTUDIO

La iteración se desarrolla utilizando el caso **AforoCC-Andes**, que se encuentra en BrightSpace Unificada - Contenidos.

### ARQUITECTURA DE LA APLICACIÓN

Disponible en la documentación y ejemplo de Parranderos-jdo.

Explicada en la monitoría fuera de clase

### RECOMENDACIONES

- ✓ Revise el caso de estudio en su totalidad, pues presenta el contexto particular sobre el cual se trabaja el proyecto y, de forma incremental, las iteraciones siguientes. Revise que comprende la terminología, las reglas de negocio y los servicios que allí se describen.
- ✓ Tenga en cuenta el manejo de usuarios en la arquitectura, el modelo y en la documentación.

### ACTIVIDADES A DESARROLLAR: CONSTRUCCIÓN DE LA APLICACIÓN

1. Revise el caso de estudio propuesto. Identifique los elementos fundamentales que hacen parte del negocio que se describe: roles de usuario, entidades de negocio, funcionalidades principales, reglas de negocio. Modifique, si es necesario, el modelo entregado en la iteración 1.
2. Documente sus resultados en un archivo que DEBE seguir el estándar de reportes técnicos establecido para el curso, disponible en Brightspace Unificada.
3. **(10%)** Diseño y construcción de la base de datos
  - a. Asegúrese de que su modelo está en BCNF (Boyce-Codd Normal Form)
  - b. Utilizando SQL Developer, cree en Oracle las tablas correspondientes al modelo de datos, con sus restricciones de integridad. Para ello, utilice 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 tablas y restricciones.
- c. Anexe a los entregables los archivos .sql utilizados.
  4. **(18%)** Preencha las tablas con información de base con datos suficientes para poder realizar pruebas. Ver sección de Requerimientos funcionales de este documento (RF1, RF2, RF3 y RF5)
  5. **(72%) Implemente los requerimientos funcionales de modificación (RF4, RF6 y RF7) y los requerimientos funcionales de consulta (RFC1, RFC2 y RFC3) solicitados más adelante en este documento. Debe seguir la arquitectura propuesta para el curso.** Evaluación detallada: 17% para cada requerimiento funcional de modificación (3) y 7% para cada requerimiento funcional de consulta (3).
    - a. Desarrolle las clases que corresponden a la lógica y la conexión a la base de datos. Para el desarrollo, siga la arquitectura de la aplicación que se muestra en el ejemplo de desarrollo que utiliza `parranderos.jdo` y que encuentra en **Brightspace**.
    - b. Implemente los escenarios de prueba descritos en la sección correspondiente en este documento, para verificar el manejo de los datos persistentes y **verificación de restricciones por el SMBD**.
  6. **(10%)** Implementación y pruebas del bono

## DESARROLLO DEL CASO DE ESTUDIO - AFOROC-ANDES

### Requerimientos funcionales de modificación:

- **RF1, RF2, RF3 y RF5:** carga de las tablas con la información de base utilizando sentencias SQL INSERT. La información de base son los espacios, los tipos de visitantes, los visitantes y los lectores de carnet.
  - **RF4 y RF6:** implementar de forma **simple** las operaciones de creación y borrado de visitantes y para registrar la entrada/salida de un visitante a un espacio del centro comercial: para cada operación se supone que se cumplen todas las condiciones para poderla realizar.
  - **RF7:** Implementar una versión **simple** de la operación de cerrar un establecimiento: cuando no hay ningún visitante en el establecimiento
- ✓ Anexe en el entregable un archivo en texto plano (.sql), con las sentencias SQL utilizadas.

### Requerimientos funcionales de consulta:

Debe realizar los requerimientos **RFC1, RFC2 y RFC3**.

- ✓ Anexe en el entregable un archivo en texto plano (.sql), con las sentencias SQL utilizadas.

### Requerimientos no funcionales:

- ✓ Por facilidad de uso, cuando se solicite información al usuario, se le debe presentar únicamente las opciones que son válidas para el usuario y el requerimiento que se está desarrollando.
- ✓ Se debe minimizar la posibilidad de ingreso de datos incorrectos por parte del usuario.
- ✓ Los requerimientos solicitados sobre los datos **DEBEN** ser realizados utilizando **el menor número posible de sentencias SQL**.
- ✓ Deben garantizarse los requerimientos no funcionales propios al negocio, tal como están descritos en la descripción del caso de negocio (**RNF1 a RNF4**).

### Bono:

Los requerimientos **RFC4 y RFC5** son opcionales, se reconoce su desarrollo como bono en la iteración.

- ✓ Anexe en el entregable un archivo en texto plano (.sql), con las sentencias SQL utilizadas.

## ESCENARIOS DE PRUEBA

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

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.

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

## ENTREGA

Fecha y hora límite: **18 de octubre de 2020, 22:00**

- En la actividad correspondiente en Brightspace, notifique 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 los modelos solicitados, resultados logrados, resultados no logrados, balance del plan de pruebas, supuestos adicionales sobre las reglas de negocio encontradas en el caso de estudio. El nombre del archivo debe seguir el mismo formato de nombramiento del zip.
  - Proyecto Java con el código fuente, la documentación del proyecto de software 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.
  - 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 (`.sql`) y pruebas indicados en el enunciado (`.sql`).
    - El archivo de configuración de la interfaz (`.json`)
  - Archivo `readme.txt`, con todos los pasos a seguir para instalar la aplicación.
- 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 en el sitio Web del curso

Documentación en línea de Oracle 19c: <https://docs.oracle.com/en/database/oracle/oracle-database/19/index.html>