

**Carrera:** INGENIERIA EN INFORMATICA

**Asignatura:** 3641 – Bases de Datos Aplicada.

**Tema:** Práctico Integrador

**Unidad:** 1 a 6

**Objetivo:** Que el alumno demuestre su comprensión de los conceptos vertidos a lo largo de la materia en un escenario real de aplicación. Que encuentre algunas de las dificultades reales a las que se enfrentará en su ejercicio profesional y sea capaz de resolverlas utilizando las herramientas provistas en la materia así como otras que obtenga como fruto de su propio estudio.

**Competencia/s a desarrollar:**

- Especificación, proyecto y desarrollo de sistemas de información.
- Especificación, proyecto y desarrollo de sistemas de comunicación de datos.
- Especificación, proyecto y desarrollo de software.
- Proyecto y dirección en lo referido a seguridad informática.
- Establecimiento de métricas y normas de calidad de software.
- Procedimientos y certificaciones del funcionamiento, condición de uso o estado de sistemas de información, sistemas de comunicación de datos, software, seguridad informática y calidad de software.
- Dirección y control de la implementación, operación y mantenimiento de sistemas de información, sistemas de comunicación de datos, software, seguridad informática y calidad de software.
- Identificación, formulación y resolución de problemas de ingeniería en sistemas de información/informática.
- Concepción, diseño y desarrollo de proyectos de ingeniería en sistemas de información / informática.
- Gestión, planificación, ejecución y control de proyectos de ingeniería en sistemas de información / informática.
- Utilización de técnicas y herramientas de aplicación en la ingeniería en sistemas de información / informática.
- Generación de desarrollos tecnológicos y/o innovaciones tecnológicas.
- Desempeño en equipos de trabajo.
- Comunicación efectiva.

**Descripción de la actividad:**

1. Tiempo estimado de resolución: cuatrimestral
2. Metodología: En computadora.
3. Forma de entrega: Obligatoria. Coloquio grupal.
4. Metodología de corrección y feedback al alumno: Presencial y por Miel.

### **Pautas para las entregas**

Se hará disponible un archivo plantilla de Word con el formato que deberán respetar. El documento se completará de forma iterativa incremental en base a las correcciones y sucesivas entregas. Al momento de entregarlo el archivo deberá renombrarse (manteniendo la extensión docx) siguiendo la forma: "ComXXXX\_GrupoYY\_EntregaZZ". Si se trata de la comisión 2900, el grupo 8 y la segunda entrega, el archivo se denominará "Com2900\_Grupo08\_Entrega02". Note que se usan ceros a la izquierda para mantener el ancho del nombre de archivo.

Cuando la entrega incluya otros archivos (por ejemplo scripts, proyectos, etc.) cada archivo deberá mencionarse en el documento en la sección que corresponda y luego todos los archivos deberán entregarse en formato ZIP en un archivo denominado de la misma forma que el documento ("ComXXXX\_GrupoYY\_EntregaZZ").

Se espera que presten cuidadosa atención a las pautas y formatos. Las entregas que no cumplan las mismas no se aceptarán.

Si un grupo debe realizar correcciones el numerador de entrega se incrementará (no puede haber dos entregas distintas con el mismo número). Esto significa que dos grupos pueden estar entregando la misma consigna con un número de entrega distinto.

Las entregas se harán en todo caso por la plataforma MIEL donde uno de los miembros del grupo enviará los archivos en nombre del equipo utilizando la sección de "Entregas".

### **Código fuente**

Si la entrega incluye scripts o código fuente, se espera que esté organizado en un proyecto. Solo se incluirán los archivos en la versión actual correspondiente a la entrega, y aquellos que se requieran como dependencias (no deben incluirse versiones anteriores).

Los scripts deben prepararse de tal forma que en cualquier entrega se pueda probar lo requerido en la consigna. Por tanto si para ello se deben crear objetos, dependencias, cargar juegos de prueba, etc., los scripts deben incluir todo lo necesario.

Todos los archivos fuentes (de cualquier tipo) que se entreguen deben comenzar con un comentario donde conste el enunciado (la parte que se está cumplimentando), fecha de entrega, número de comisión, número de grupo, nombre de la materia, nombres y DNI de los alumnos.

### **Juegos de prueba**

Siempre que se entreguen módulos de código fuente deben acompañarse de scripts de testing. Los juegos de prueba deben entregarse en un archivo separado al script del fuente, aunque se incluya en el mismo proyecto. Todo módulo ejecutable (SP, función), debe ser utilizado en al menos una prueba.

Deben utilizarse comentarios para indicar el resultado esperado de cada prueba. Por ejemplo, si un juego de prueba pretende demostrar que un dato se valida y por fallar la validación no

se completa la transacción, el comentario debe indicar la validación, por qué falla y qué evidencia se presenta (mensaje por consola, error, etc.).

### **Normas para variables, objetos, etc.**

Cada grupo deberá determinar una norma de nomenclatura para las variables, los store procedures, tablas, etc. Se espera que la norma cumpla las buenas prácticas (por ejemplo las tablas se denominan en singular). La norma que generen y adopten debe constar en la documentación.

### **Pautas para los coloquios**

Todas las entregas pueden acompañarse de coloquios. No necesariamente todos los grupos rindan el coloquio el mismo día. Tampoco es necesario que todos los integrantes participen el mismo día, pero será condición necesaria para la aprobación que para el final de la cursada todos los miembros del grupo hayan demostrado su participación en el trabajo interviniendo en algún coloquio.

Los coloquios podrán realizarse en forma presencial o virtual, dependiendo de la clase con la que coincidan. Cuando se trate de coloquios virtuales se espera que cada participante que quiera intervenir cuente con equipo para hablar en vivo (no sería suficiente utilizar el chat) y de ser posible activar su video (no es requisito, pero es deseable).

### **Revisión de código fuente**

Los coloquios que consistan en entrega/defensa de scripts, se presentarán de la siguiente manera: cada grupo pasará al frente por turnos. Cuando sea su turno se presentará el archivo fuente ante toda la clase y el docente les marcará aspectos positivos y negativos del mismo. El grupo deberá tomar nota para realizar las correcciones del caso. El docente le indicará si deben reentregar bajo la misma consigna o si pueden adoptar las correcciones como parte de la siguiente entrega.

Aunque cada grupo pasará la revisión de código por unos minutos (por lo tanto será imposible revisarlo en su totalidad), todos los grupos deben tomar nota de los aspectos positivos y negativos que se mencionen. De esta forma todos los grupos deberán incorporar los cambios requeridos, sea que el docente se los haya mencionado directamente o no.

En consonancia con las buenas prácticas, se les solicita que apliquen el proceso de integración continua para el desarrollo de software; para ello se utilizará la herramienta Git.

### **Pautas para la denominación y creación de la base de datos**

Cada grupo deberá generar una DB con un nombre distinto. Para ello usarán el nombre de la comisión y del grupo como denominador de la DB. Por ejemplo "Com3900G02". Note que es una versión abreviada del nomenclador de las entregas. El formato es ComXXXXGYG donde XXXX es el código de comisión e YY es el numero de grupo con cero a la izquierda de ser necesario.

# Proyecto

## Introducción

Un supermercado de venta minorista solicita la creación de un sistema que le permita obtener reportes sobre sus ventas.

Hoy cuenta con tres sucursales cuya información se encuentra descentralizada, cada sucursal maneja su propia información. Solo se cuenta con un reporte de ventas, ya que el cliente no cuenta con toda la información de forma digital.

## Objetivo

Generar un sistema de bases de datos que registre las ventas realizadas en cada sucursal.

## Limite

Desde que se realiza la inserción del catálogo de productos hasta que registra la venta.

## Alcance

Actual (AS-IS)

En reporte de ventas se visualiza de la siguiente manera:

Factura ID	Tipo de Factura	Ciudad	Tipo de cliente	Genero	Línea de producto	Precio unitario	cantidad	Total	Fecha	Hora	Medio de pago
765-26-6951	A	Yangon	Normal	Male	Sports and travel	72.61	6	457.443	1/1/2019	10:39	Credit card
530-90-9855	A	Yangon	Member	Male	Home and lifestyle	47.59	8	399.756	1/1/2019	14:47	Cash
891-01-7034	B	Mandalay	Normal	Female	Electronic accessories	74.71	6	470.673	1/1/2019	19:07	Cash
493-65-6248	C	Naypyitaw	Member	Female	Sports and travel	36.98	10	388.29	1/1/2019	19:48	Credit card

Futuro (TO BE)

En el reporte futuro se quiere contar con la siguiente información:

ID Factura	Tipo de Factura	Ciudad	Tipo de cliente	Genero	Línea de producto	Producto	Precio Unitario	Cantidad	Fecha	hora	Medio de Pago	Empleado	Sucursal
750-67-8428	A	Yangon	Member	Female	Almacen	Crema facial reafirmante	5	7	1/5/2019	13:08	Ewallet	257020	San Justo
226-31-3081	C	Naypyitaw	Normal	Female	Bebidas	Leche entera Asturiana	1.25	5	3/8/2019	10:29	Cash	257021	Ramos Mejia
631-41-3108	A	Yangon	Normal	Male	Bebidas	Combinado de aperitivo	1.2	7	3/3/2019	13:23	Credit card	257022	Lomas del Mirador
123-19-1176	A	Yangon	Member	Male	Frescos	Carne de pimiento chorizo	1.3	8	1/27/2019	20:33	Ewallet	257023	San Justo
373-73-7910	A	Yangon	Normal	Male	Frescos	Queso de cabra Hacendado	1.9	7	2/8/2019	10:37	Ewallet	257024	Ramos Mejia

Se entrega un archivo con la información complementaria de cómo se espera almacenar la información.

El proyecto por cotizarse debe contemplar lo siguiente:

- Registración del empleado y la sucursal en donde se realizó la venta
- Carga de productos de forma unificada. Cada sucursal maneja su propio catálogo. Se cuentan con tres catálogos de productos:
  - Electronic accessories.xlsx
  - catalogo.csv
  - Productos\_importados.xlsx

Cada sucursal debe poder realizar la venta de cualquiera de los productos que se encuentran en los catálogos informado por el cliente.

- Estandarización en la forma de ingresar de los medios de pago
- Reportes:

El sistema debe ofrecer los siguientes reportes en xml.

Mensual: ingresando un mes y año determinado mostrar el total facturado por días de la semana, incluyendo sábado y domingo.

Trimestral: mostrar el total facturado por turnos de trabajo por mes.

Por rango de fechas: ingresando un rango de fechas a demanda, debe poder mostrar la cantidad de productos vendidos en ese rango, ordenado de mayor a menor.

Por rango de fechas: ingresando un rango de fechas a demanda, debe poder mostrar la cantidad de productos vendidos en ese rango por sucursal, ordenado de mayor a menor.

Mostrar los 5 productos más vendidos en un mes, por semana

Mostrar los 5 productos menos vendidos en el mes.

Mostrar total acumulado de ventas (o sea tambien mostrar el detalle) para una fecha y sucursal particulares

## Fuera de Alcance

- Sueldos y jornales de los empleados de la sucursal
- Generación de Facturas digitales según condición de IVA.
- Administración de Stock de productos
- Gestion de proveedores

## Aceptación del usuario

Se debe tener en cuenta la performance de las operaciones, se espera que el equipo de desarrollo implemente las medidas necesarias para el correcto funcionamiento en producción, es por ello por lo que se le pide junto con la entrega del producto software una prueba de stress y la documentación (DER) correspondiente al sistema desarrollado.

El Trabajo práctico es de carácter integrador, conforme se avance con el cronograma de estudio de la materia se deben completar los hitos de este.

### **Entrega 1**

Durante el análisis del sistema a desarrollar para la firma “Aurora SA.” usted y su equipo concluyen que será necesario contar con una base de datos relacional.

Deben presentar alternativas al cliente sobre motores de bases de datos.

Su grupo será responsable de investigar:

- Requerimientos técnicos
  - Software de base (presente opciones)
  - Se cuenta con un servidor con procesador de última generación con procesador de 8 (ocho) núcleos y 64 GB de memoria, además de varios TB de almacenamiento en RAID.
- Costo y modo de licenciamiento
  - La primera estimación es que el sistema se accederá desde 50 puestos.
  - Se estima que la base de datos acumulará 4 GB en los primeros dos años.
  - No se cuenta con personal capacitado en la organización que pueda desempeñarse como dba.
  - Se requiere un esquema de alta disponibilidad.
  - Hay información sensible que debe ser almacenada utilizando cifrado.
  - Informe los costos en USD (BNA) o pesos a la cotización del día.

Su grupo recibirá un motor de base de datos para investigar. No es necesario que investigue más de uno.

#### **Genere un informe como el que entregaría en mano al cliente:**

El informe debe contener los siguientes puntos:

- Cumplimiento de los requisitos técnicos.
- Personal capacitado requerido (DBA-Soporte-Mantenimiento).
- Costo del soporte técnico del motor.
- Cantidad de horas Soporte técnico si las ofrece y el costo de estas.
- Costo de Licencia.
- Seguridad que ofrece sobre la información.
- Conclusiones (indique si el equipo utilizaría el motor asignado para la implementación del sistema, justificando la utilización o no del mismo.)

El informe debe completarse según la plantilla Estimacion de costos.docx. en formato PDF con la siguiente nomenclatura: [comisión-Equipo]- Estimacion de costos\_[version].pdf

Comisión: martes o viernes según corresponda.

Equipo: número de equipo con dos dígitos

Versión: indica la versión del documento. Utilizar dos dígitos comenzando por 01

Durante la clase conversaremos sobre lo que han investigado. Preparen las preguntas que puedan surgir y asegúrense de que todo el grupo cuente con copia del informe. Investiguen las siglas, tecnologías y terminología que desconozcan.

Prepárese la justificación con respecto al uso del DBMS que se le ha asignado como alternativa al cliente. En caso de que considere que no es recomendable, deberá fundamentarlo.

## **Entrega 2**

Luego de presentar las alternativas al cliente, el mismo planteó la posibilidad de alojar la base de datos en la nube.

Repita el análisis realizado en la entrega 1 con un servicio alojado en la nube, y agregue estos costos al informe entregado.

Realice el ejercicio de cálculo de costo para:

- Amazon Web Services (<https://calculator.aws/#/>)
- Microsoft Azure (<https://azure.microsoft.com/es-es/pricing/calculator>)
- Google Cloud (<https://cloud.google.com/products/calculator>)

Utilice los mismos lineamientos de capacidad y cálculo que estimó en la primera parte. Puede que algunas alternativas cloud no incluyan el motor de base de datos que se le indicó investigar. En ese caso, presente opciones viables a su criterio.

Similarmente, si su grupo no recomendó utilizar el DBMS que se le asignó en la primera entrega, escojan una de las alternativas viables.

Establezca en el planteamiento de la tecnología a utilizar cual de los sistemas de Cloud computing: IaaS, PaaS y SaaS utilizaría y por qué.

El cliente comenta que RRHH está analizando la posibilidad de contratar un DBA. Investigue y agregue el perfil técnico que crea conveniente

El objetivo de esta parte es generar una estimación de inversión inicial y costo mensual de mantener una base de datos en la nube tal que pueda compararlo con lo obtenido en la primera parte.

Asegúrese de conocer el significado de términos como CAPEX, OPEX y TCO e indique los mismos dentro del informe a entregar. Le sugerimos indicarlo en el informe pero lo realmente importante es que sepa explicarlos.

Existen aspectos de la implementación que no se incluyen en este ejercicio, incluya los factores que consideran importantes evaluar antes de decidir. Tome una decisión respecto a lo que considera que el cliente querrá, plantee argumentos para defender esa decisión como si tuviera que explicárselo a la junta directiva del comercio minorista e inclúyala en el informe.

El informe debe completarse según la plantilla Estimacion de costos.docx. en formato PDF con la siguiente nomenclatura: [comisión-Equipo]- Estimacion de costos\_[version].pdf

Comisión: martes o viernes según corresponda.

Equipo: número de equipo con dos dígitos

Versión: indica la versión del documento.

Al inicio de la clase entreguen la copia impresa a los docentes. Luego en clase conversaremos sobre lo que han investigado. Prepárense para responder las preguntas que puedan surgir y asegúrense de que todo el grupo cuente con copia del informe.

### **Entrega 3**

Luego de decidirse por un motor de base de datos relacional, llegó el momento de generar la base de datos.

Deberá instalar el DMBS y documentar el proceso. No incluya capturas de pantalla. Detalle las configuraciones aplicadas (ubicación de archivos, memoria asignada, seguridad, puertos, etc.) en un documento como el que le entregaría al DBA.

Cree la base de datos, entidades y relaciones. Incluya restricciones y claves. Deberá entregar un archivo .sql con el script completo de creación (debe funcionar si se lo ejecuta "tal cual" es entregado). Incluya comentarios para indicar qué hace cada módulo de código.

Genere store procedures para manejar la inserción, modificado, borrado (si corresponde, también debe decidir si determinadas entidades solo admitirán borrado lógico) de cada tabla.

Los nombres de los store procedures NO deben comenzar con "SP".

Genere esquemas para organizar de forma lógica los componentes del sistema y aplique esto en la creación de objetos. NO use el esquema "dbo".

El archivo .sql con el script debe incluir comentarios donde consten este enunciado, la fecha de entrega, número de grupo, nombre de la materia, nombres y DNI de los alumnos.

Entregar todo en un zip cuyo nombre sea Grupo\_XX.zip mediante la sección de prácticas de MIEL. Solo uno de los miembros del grupo debe hacer la entrega.



#### **Entrega 4**

***Se proveen los archivos en el TP\_integrador\_Archivos.zip***

***Ver archivo “Datasets para importar” en Miel.***

Se requiere que importe toda la información antes mencionada a la base de datos:

- Genere los objetos necesarios (store procedures, funciones, etc.) para importar los archivos antes mencionados. Tenga en cuenta que cada mes se recibirán archivos de novedades con la misma estructura, pero datos nuevos para agregar a cada maestro.
- Considere este comportamiento al generar el código. Debe admitir la importación de novedades periódicamente.
- Cada maestro debe importarse con un SP distinto. No se aceptarán scripts que realicen tareas por fuera de un SP.
- La estructura/esquema de las tablas a generar será decisión suya. Puede que deba realizar procesos de transformación sobre los maestros recibidos para adaptarlos a la estructura requerida.
- Los archivos CSV/JSON no deben modificarse. En caso de que haya datos mal cargados, incompletos, erróneos, etc., deberá contemplarlo y realizar las correcciones en el fuente SQL. (Sería una excepción si el archivo está malformado y no es posible interpretarlo como JSON o CSV).

**Pautas generales para el desarrollo del TP**

- Documente las correcciones que haga indicando número de línea, contenido previo y contenido nuevo. Esto se cotejará para constatar que cumpla correctamente la consigna.
- El código fuente no debe incluir referencias hardcodeadas a nombres o ubicaciones de archivo. Esto debe permitirse ser provisto por parámetro en la invocación. En el código de ejemplo el grupo decidirá dónde se ubicarían los archivos. Esto debe aparecer en comentarios del módulo.
- Adicionalmente se requiere que el sistema sea capaz de generar un archivo XML detallando XXXXX. El mismo debe constar de los datos del XXXX.
- Deberá presentar un archivo .sql con el script de creación de los objetos correspondientes. En el mismo incluya un comentario donde conste este enunciado, la fecha de entrega, número de grupo, nombre de la materia, nombres y DNI de los alumnos. El mismo archivo SQL debe permitir la generación de los objetos consignados en esta entrega (debe admitir una ejecución completa sin fallos).
- Cada archivo SQL que contiene el código de creación de objetos debe comenzar su nombre con dos dígitos indicando el orden en que deben ejecutarse. Por ejemplo "00\_CreacionSPImportacionCatalogo". Estos archivos deben entregarse (como todos los scripts) dentro de un proyecto/solución. Todos deben estar en el repositorio git del grupo
- También debe presentar un archivo .sql que consista en las invocaciones a los SP creados para generar la importación. Este archivo (que puede considerarse de testing) debe contener comentarios para indicar el orden de ejecución.
- Entregar todo en un zip cuyo nombre sea Grupo\_XX.zip mediante la sección de prácticas de MIEL. Solo uno de los miembros del grupo debe hacer la entrega.
- Se recomienda revisar periódicamente el foro en Miel de la materia. En el mismo se informará el agregado de información, pautas o dudas respecto al TP