

**UNIVERSIDAD ABIERTA INTERAMERICANA**

**FACULTAD DE TECNOLOGÍA INFORMÁTICA**

MATERIA:

SEMINARIO

DE

APLICACIÓN

PROFESIONAL

1

er Parcial

Teórico

y

Práctico

**Alumno: Lucas Emanuel Saavedra**

:

**Fecha: 25 / 05 / 2024**

**Tema**

**1**

**Comisión – Localización - Turno:**

**Práctica: Teoría: Nota:**

**Temas para evaluar:** Procesos y productos para la creación de software. Análisis y diseño de una aplicación web. Arquitectura de base. POO, Conocimientos de bases de datos. Tecnologías web. Patrones de diseño. Modelado.

**Objetivos:**

Comprender las pautas del proceso para la creación de software desde la perspectiva de la Ing. SW. Comprender cómo se desarrolla una aplicación con características ingenieriles.

**Modalidad:** Parcial domiciliario

**Requisitos para aprobar:** Para que el parcial esté aprobado el alumno deberá tener correctamente desarrolladas el 60% de la teoría y resuelto el ejercicio práctico.

**Tiempo:**

**Recomendaciones:**

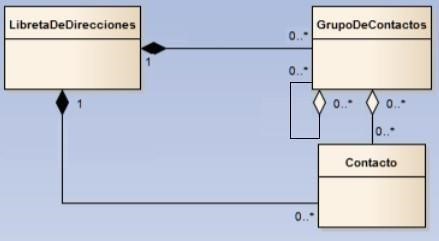
1. Lea todo el parcial antes de comenzar a responder.
2. Desarrolle una redacción clara y precisa contestando lo que la pregunta requiere.
3. Observe la ortografía ya que la misma es parte del parcial.
4. Si considera que no comprende alguna consigna antes de comenzar consulte a su profesor.

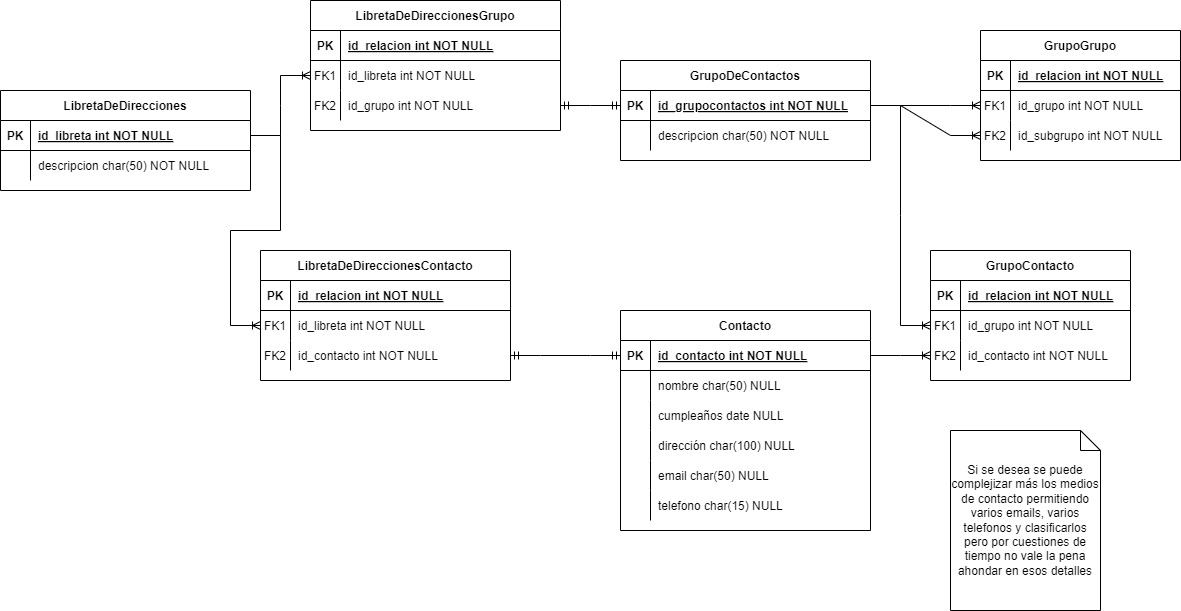
**Notas:** Las preguntas en las que se seleccionen opciones se deberá optar solo por una de las posibilidades. La indicación se efectuará con una X sobre su lateral izquierdo

Las preguntas que solicitan justificación serán consideradas válidas si poseen la misma correctamente.

Las preguntas de múltiples posibilidades y verdadero / falso restan 0.50 puntos en caso de estar mal contestadas. En las preguntas verdadero / falso se debe tachar la opción incorrecta.

(\*) la cifra entre paréntesis en cada pregunta es la cantidad de puntos sobre 100.

1. Nos solicitan desarrollar un DER a partir del siguiente diagrama de clases (Agregue los atributos que considere necesarios al modelo): (50) (Pegar directamente la imagen DER generada con la herramienta que usted desee: por ejemplo draw.io, E.A, etc.)



1. ¿Cómo podría relacionar la plataforma Auth0 (Similitudes/diferencias) con el concepto del patrón (Composite) Usuario-Patente-Familia? (50)

Similitudes:

* La primera similitud que puedo marcar es que tanto la plataforma Auth0 como la otra solución permiten realizar la autenticación y la autorización de los usuarios.
* Otra similitud es que las jerarquías de autorización que ofrecen son similares (cambian un poco los nombres) el equivalente de Patente en Auth0 se lo llama Permiso y las Familias (de patentes) son la contraparte de los Roles en Auth0.

Diferencias:

* La primera diferencia es que Auth nos ofrece muchas funcionalidades ya desarrolladas y listas para usar como por ejemplo la posibilidad de autenticar contra distintos proveedores de identidad, en el otro esquema no está previsto y tendríamos que analizar nosotros como integrarlo
* Otra diferencia es que con la utilización del patrón composite podemos anidar las familias en tantos niveles como queramos. En cambio, en Auth0 ya está definido que los niveles son Roles y Permisos, a un usuario le puedo asignar múltiples roles, todos los que yo quiera, pero no puedo generar un rol que agrupe varios roles.

# PRÁCTICA (100)

1. Desarrollar un diagrama de clases (Sólo del dominio) y el programa donde se observe claramente los campos, propiedades y métodos para un sistema de logística donde nos informan que existen varios depósitos y tiendas donde sus principales datos son: Número de identificación, nombre y dirección (Calle y número, localidad, código postal y provincia). Las acciones que se pueden desarrollar son: Transferencia de stock de diversos productos entre depósitos y transferencia de stock desde depósitos a tiendas. Los depósitos y tiendas cuentan con ciertos productos de los cuales conocemos su nombre, descripción y precio. Se deberá registrar cada movimiento histórico de las operaciones que contendrá: Origen, destino, fecha, usuario y el listado de productos con sus cantidades transferidos. Se deberá contar con al menos un usuario registrado en el sistema para realizar las operaciones. Para poder realizar transferencias de cualquier producto se debe validar el stock sobre el origen. **(10)**



**UNIVERSIDAD ABIERTA INTERAMERICANA**

**FACULTAD DE TECNOLOGÍA INFORMÁTICA**

MATERIA:

SEMINARIO

DE

APLICACIÓN

PROFESIONAL

1

er Parcial

Teórico

y

Práctico

1. Generar un modelo de Repositorio accesible a través de una capa de lógica, que deberá a su vez, ser accedida por una fachada de tipo WebApi (Con los métodos necesarios para las operaciones solicitadas) para persistir la información de transferencias entre los diversos depósitos/tiendas y luego poder consultar las mismas. Puede utilizar cualquier modelo de persistencia, incluso en memoria, para las entidades intervinientes. **(20)**
2. El modelo de solución generado deberá contener mínimamente 5 capas (Sitio web para al cliente, WebApi, Business Layer, DomainModel y Repositorio) asignando las responsabilidades correspondientes. Utilizar los patrones de diseño y arquitectura que usted crea conveniente. **(20)**
3. Deberá generar casos de prueba para testear las operaciones solicitadas. Mínimamente contemplar:
   1. Se tienen que poder filtrar los tipos de operaciones a partir de un destino (Escoger un depósito o tienda) y luego mostrar: Origen, producto, cantidad, usuario y fecha, esta última ordenada de manera descendente.
   2. Se debe de contar con otro informe que nos permita filtrar los destinos que solicitaron más de tres operaciones en el día, mostrando origen, destino, producto, cantidad y usuario que los solicitó ordenado por fecha de manera ascendente.
   3. A partir de un número de tienda, mostrar los primeros 10 movimientos entre las fechas

determinadas por el usuario (Elija los datos que desea mostrar para esta consulta). **(40)**

1. Entre la WebApi y el sitio web se dialogará con formato de mensajes en json. Puede utilizar cualquier tecnología de desarrollo para la web: Razor, Blazor (No es necesaria la api en este caso), HTML5 puro con invocaciones Ajax, etc. También será valorado el diseño CSS y javascript que permita una buena usabilidad. **(10)**

Lucas E Saavedra

Firma del Alumno