**EVALUACIÓN CORRESPONDIENTE AL PRIMER PARCIAL**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| FACULTAD: | **Tecnología Informática** | | | | |
| CARRERA: | **Analista programador** | | | | |
| ALUMNO/A: | **Della Maggiora, Cynthia** | | | | |
| SEDE: | **N/A** | | LOCALIZACIÓN: | **Remoto** | |
| ASIGNATURA: | **Lenguajes de última generación** | | | | |
| CURSO: | **2º B-D** | | TURNO: | **Noche** | |
| PROFESOR: | **Arigós Gonzalo** | | FECHA: | **19/05/2025** | |
| TIEMPO DE RESOLUCIÓN: | | **180’** | EXAMEN PARCIAL NRO: | | **1** |
| MODALIDAD DE RESOLUCIÓN: **A Distancia / Individual** | | | Presencial / Virtual / Escrito / Oral / Individual / Grupal | | |
|  | | |  | | |
| RESULTADOS DE APRENDIZAJE: | | | | | |

**Propósito:**

Evaluar la capacidad del estudiante para diseñar y desarrollar un programa orientado a objetos. Su habilidad para administrar el tiempo y los recursos seleccionados para el logro del objetivo propuesto, su capacidad para integrar el marco teórico propuesto con los resultados alcanzados.

CRITERIOS DE EVALUACIÓN:

* Creatividad y originalidad de la propuesta.
* Claridad en la organización de la escritura del código y los comentarios incluidos en el mismo.
* Precisión en el resultado obtenido.
* Utilización de los conceptos abordados en clase: Definición de clase abstractas, selladas y concretas. Propiedades, métodos, eventos, constructores y destructores. Herencia, polimorfismo, agregación y asociación. Sobreescritura y sobrecarga.
* Racionalidad y coherencia en la validación de los datos ingresados y obtenidos, así como en el control de las excepciones

El examen se considerará aprobado con una nota de 4 (cuatro) que se obtendrá con el 60% de los ítems del PROYECTO correctamente desarrollados.

**Guía de Resolución:**

**Título de proyecto software:**

Distribuidora de bebidas: Registro de stock para una distribuidora de bebidas

**1. Introducción**

**2. Marco de Referencia**

**3. Objetivos**

Los objetivos del programa son:

1. Poder crear registros de ingresos de stock y persistirlos en una base de datos.
2. Validar los datos ingresados por el usuario.
3. Permitir que el usuario visualizar todos los registros de las observaciones realizadas.
4. Permitir eliminar el ingreso de stock
5. Permitir modificar un determinado ingreso

**4. Relevamiento de Datos**

El empleado de la distribuidora recibe un camión todos los días con nuevo stock destinado para la venta y debe realizar la carga masiva de las nuevas bebidas que entran, para ello contamos con:

Ingreso:

* ID de ingreso - PK
* Id de Marca – FK tabla Marca
* Descripción (Lata Monster sabor Mango 473ml)
* BultosIngresados
* CantidadPorBulto (4 | 6 | 12 | 24 unidades)
* DNI Empleado responsable
* Observaciones (“Se recibieron dos latas con daños menores”)
* Fecha de registro del ingreso

Marca:

* ID de marca
* Descripcion (“CocaCola” “Pepsi” “Monster Energy” “Quilmes”)

El sistema debe ser capaz de almacenar y visualizar los registros cargados.

El sistema tendrá la capacidad de validar los siguientes datos:

Marca:

* No puede estar vacío.

Descripción:

* No puede estar vacío
* Debe ser mayor a 5 caracteres

BultosIngresados:

* Debe ser mayor o igual que 1

Cantidad por bulto:

* No puede estar vacío
* Solo se admiten bultos de: 4 | 6 | 12 | 24 unidades

DNI Empleado responsable:

* No puede estar vacío
* Debe ser de 8 digitos

Observaciones:

* Campo opcional
* En caso de ser cargado, la longitud debe ser mayor a 5 caracteres

También se pide mostrar los registros cargados en una grilla (DatagridView). La grilla debe recibir una lista de objetos del tipo **Ingreso**.

Implementar un mecanismo de borrado de registros. Solicitar un Id de **Ingreso** al usuario y ejecutar el borrado del registro en la base de datos.

Permitir actualizar la cantidad de bultos ingresados de un determinado registro por su ID.

Utilizar excepciones. Prestar atención en el diseño de la base de datos y las relaciones.

**5. Plan de trabajo**

Prever una interfaz de usuario (Windows Form) que permita instanciar la clase "**Ingreso**".

¿Qué datos serán provistos por el usuario?

* Selección del Marca de bebida mediante ComboBox – Se deben obtener de la tabla correspondiente al momento de carga de la aplicación.
* Descripción, Bultos Ingresados, Cantidad por bulto, DNI empleado responsable y observaciones mediante TextBox
* Botón para guardar un ingreso

¿Qué datos serán administrados por el sistema?

* Id de Ingreso (Autonumérico)
* Fecha de registro de la observación (Fecha de hoy)

¿Qué acciones podrá hacer el usuario?

* Registrar un Ingreso de stock.
* Consultar y visualizar todos los ingresos cargados
* Actualizar la cantidad de bultos ingresados de un determinado ingreso (por ID de ingreso)
* Borrar un registro mediante su Id.

Criterios de aprobación:

* Utilizar arquitectura en capas (UI, BLL, DAL, Entity).
* Aplicar buenas prácticas de la programación orientada a objetos.
* Aplicar buenas prácticas en las capas de negocio y acceso a datos.
* Las cadenas de conexión deben estar embebidas en archivos de configuración.
* Utilizar excepciones en caso de que no se cumpla una regla de negocio y capturar todos los errores en bloques try/catch.
* Liberar correctamente los recursos mediante el uso de bloques using.
* Utilizar SqlParameters para queries con datos de negocio.
* Desarrollar la base de datos en SQL Server.
* El programa se debe entregar sin errores de compilación, comprimido en formato zip o rar, y debe incluir los scripts de instalación de la base de datos, incluyendo los datos de prueba utilizados durante el proceso de desarrollo de la evaluación.