**INSTITUTO SUPERIOR TECNOLÓGICO PRIVADO CIBERTEC**

**DIRECCIÓN ACADÉMICA**

**CARRERA PROFESIONALES**

**CURSO :** LENGUAJE DE PROGRAMACION II

### NOTA

**PROFESOR :** JORGE LUIS VIZCARRA CÁRDENAS

**SEMESTRE :** 2023-62

**CICLO :** CUARTO

**SECCIÓN :** T4LA

**GRUPO :** 01

**FECHA :** MIÉRCOLES 22 DE NOVIEMBRE DE 2023

**DURACIÓN :** 90 MINUTOS

|  |
| --- |
| **ALUMNO (A) :NILTON MANCHA CÁRDENAS** |

**EVALUACIÓN FINAL DE LABORATORIO**

**Logro:**

Tenga presente que el logro de este examen es que usted implemente el Patrón MVC, Servlet, Spring Boot y conexión con Base de Datos MySQL.

**Consideraciones generales:**

* NO está permitido el uso o consulta de cuadernos, separatas, libros o cualquier material de la asignatura durante la evaluación. el uso de papel suelto o consulta de cuadernos, separatas, libros o cualquier material de la asignatura durante el desarrollo de la evaluación.

**RoomReserve Pro** desea implementar en su nueva plataforma virtual, una funcionalidad que permita a sus clientes reservar habitaciones, así como pagar dicha reservación desde la comodidad de su casa. El modelo de Base de Datos es el siguiente:

Interfaz de usuario gráfica, Diagrama, Tabla

Descripción generada automáticamente

* + *Si desea puede adaptar la BD según su criterio*

Usted como programador, deberá realizar los siguientes requerimientos. En el plazo establecido.

1. Crear la Base de Datos con el nombre CIBERTEC2023.
2. La arquitectura del sistema debe ser utilizando el patrón MVC (Entidades de las tablas según lo solicitado).
3. Implementación del repositorio JPA.
4. Implementación del controlador para invocar a los métodos que corresponda.

**Creación y modelado de Base de Datos.**

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos | Excelente  5 puntos | Regular  3 puntos | Deficiente  1 punto |
| 5 Puntos | La BD está correctamente mapeada. | La BD presenta diferencias con el modelo establecido. | La BD está deficiente. |

**Tareas a realizar**

1. **Crear la Base de Datos:**
   * crear la base de datos con las tablas necesarias (Cliente, habitaciones, reservas).
2. **Crear Entidades JPA:**
   * Definir las entidades Cliente, Habitación, y Reserva con las anotaciones JPA apropiadas.
3. **Crear Repositorios JPA:**
   * Crear interfaces de repositorio JPA para cada entidad (Cliente Repository, HabitacionRepository, ReservaRepository).
4. **Implementar Servicios:**
   * Crear servicios (ClienteService, HabitacionService, ReservaService) para manejar la lógica de negocios.
5. **Implementar Controladores:**
   * Crear controladores (ClienteController, HabitacionController, ReservaController) con métodos para manejar las solicitudes web.
6. **Diseñar Vistas Thymeleaf:**
   * Crear plantillas Thymeleaf para mostrar el catálogo de habitaciones, el carrito de reservas, y la confirmación de reserva.
7. **Explorar Habitaciones:**
   * En la vista de catálogo, mostrar la lista de habitaciones disponibles con detalles.
   * Agregar un botón "Agregar al Carrito" junto a cada habitación.
8. **Seleccionar Habitaciones:**
   * Permitir a los Clientes hacer clic en "Agregar al Carrito" para seleccionar habitaciones.
9. **Carrito de Reservas:**
   * Crear una página para mostrar las habitaciones seleccionadas en el carrito de reservas.
   * Mostrar detalles y opciones para ajustar la reserva.
10. **Confirmar Reservas:**
    * En la página del carrito, agregar un botón "Confirmar Reserva".
    * Permitir a los usuarios revisar y ajustar detalles antes de confirmar.
11. **Proceso de Pago:**
    * Agregar un paso para el proceso de pago.
    * Actualizar el estado de pago en la tabla de reservas si se completa el pago.
12. **Confirmación de Reserva:**
    * Mostrar una página de confirmación después de completar la reserva.
    * Proporcionar detalles de la reserva y un número de confirmación.
13. **Entrega del Proyecto:**
    * Subir el código a un repositorio Git (GitHub)
    * Proporcionar la URL del repositorio como parte de la entrega.

**Rubrica de Evaluación**

|  |  |  |  |
| --- | --- | --- | --- |
| Criterio | Excelente  5 puntos | Regular  2 puntos | Deficiente  0 puntos |
| Formulario | Muestra el formulario, según lo establecido. Carga lashabitaciones desde base de datos. | Muestra formulario según lo establecido, pero no carga la información de las habitaciones desde la base de datos. | No realiza el formulario de ventas. |
| Criterio | **Excelente**  **3 puntos** | **Regular**  **2 punto** | Deficiente  0 puntos |
| Opción Agregar | Agrega la información según lo establecido. | Agrega la información pero presenta deficiencias. | No agrega la información |
| Criterio | **Excelente**  **4 puntos** | **Regular**  **2 punto** | Deficiente 0 puntos |
| Opción Pagar | Registra correctamente en la BD, cabecera y detalle de venta. | Registra cabecera y detalle, pero presenta deficiencias. | No registra la compra. |
| Criterio | **Excelente**  **3 puntos** | **Regular**  **1 punto** | Deficiente  0 puntos |
| Usa el patrón MVC | Implementa correctamente el patrón MVC en Spring | Presenta deficiencias en la implementación del modelo MVC. | No implementa el patrón MVC. |