**HOJA INTERACTIVA DE APRENDIZAJE Nº 09**

**Apellidos y Nombres:** Espinoza Ccente Miguel Angel

**Docente:** Nestor Audante Ramos **Fecha de entrega:** 28/12/2021

**MVC**

1. **Indicadores:**
   1. Identifica uso de capas en una aplicación backend.
   2. Identifica el uso de inyección de dependencias.
   3. Crea aplicaciones usando las capas Controlador, servicio y DAO.
   4. Prueba los servicios Rest creados.
2. **Actividades para desarrollar en forma individual:** 
   1. Complete la tabla sobre la sintaxis Java:

|  |  |
| --- | --- |
| **Código fuente** | **Errores** |
| @RestController public class Controlador {  @Autowired  private Servicio servicio;  @ResponseMapping(value = "/obtener-productos",  method = RequestMethod.*POST*,  produces = "application/json;charset=utf-8"  )  public @RequestBody RespuestaProducto obtenerProductos(){  RespuestaProducto respuestaProducto = new RespuestaProducto();  respuestaProducto.setLista(servicio.obtenerProductos());  return respuestaProducto;  } } | @RequestMapping  @ResponseBody |
| public class DaoImpl implements Dao{    private JdbcTemplate jdbcTemplate;  private Connection conexion;  } | @Repository  @Autowired |
| public class ServicioImpl implements Servicio{  @Autowired  private Dao dao;  } | @Service  @Transactional |

* 1. Complete la tabla escribiendo los conceptos aprendidos en clase

|  |
| --- |
| **Conceptos** |
| **Patrones de diseño estructural**  **Explican como ensamblar objetos y clases en estructuras mas grandes, permiten componer objetos en estructuras de árbol.** |
| **Decorator**  **Usa la agregación como sustituto de herencia pura** |
| **Facade**  **Proporciona una interfaz de una biblioteca de forma simplificada** |
| **Adapter**  **Permite la colaboración entre objetos con interfaces incompatibles** |
| **Bridge**  **Permite separar las clases extensas mediante la abstracción e implementación permitiendo desarrollar de forma independiente.** |
| **MVC**  **Es modelo-vista-controlador, utilizado para implementar interfaces de usuario, datos y lógica de control.** |
| **Controlador**  **Actúa como intermediario entre el modelo y la vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.** |
| **DAO**  **Data Access object, objeto de acceso a datos, es una interfaz común entre la aplicación y los dispositivos de almacenamiento de datos.** |
| **Inyección de dependencias**  **Servicio de impresión dependa de otros servicios y las responsabilidades queden mas claras, es decir el código sea mas ordenado o limpio.** |

* 1. Complete la tabla.

|  |
| --- |
| **Código fuente: Cree una consulta SQL que contemple actualice datos de un registro de tabla en una Clase DaoImpl.** |
| public void actualizarUbicacion(String nuevaUbicacion, Departamento departamento) {  String sql = "update departamento set ubicacion = ? where codigo\_departamento=?";  crearConexion();  try {  PreparedStatement sentencia = conexion.prepareStatement(sql);  sentencia.setString(1,nuevaUbicacion);  sentencia.setString(2,departamento.getCodigo());  Integer filasAfectadas = sentencia.executeUpdate();  System.*out*.println("Filas afectadas : " + filasAfectadas);  sentencia.close();  conexion.commit();  cerrarConexion();  } catch (SQLException throwables) {  throwables.printStackTrace();  }  } |

* 1. Suba el documento resuelto y la solución del caso de estudio en un directorio llamado lab-9.

1. **Referencia de material** 
   1. Alvarez, Cecilio (2016). ¿Qué es Spring Boot?. Recuperado de <https://www.arquitecturajava.com/que-es-spring-boot/>
   2. Spring (2018). Building a RESTful Web Service. Recuperado de <https://spring.io/guides/gs/rest-service/>

1. **Conclusiones y recomendaciones**

La manera en que se separan los diferentes servicios hace que sea más fácil entender la programación orientada a objetos, existen diferentes paginas que nos ayudan a entender los diversos conceptos uno de los que encontré fue :

<https://www.arquitecturajava.com/el-patron-de-inyeccion-de-dependencia/> tal vez se podría poner en la bibliografía para repasar los conceptos.