

## Taller Agiles

### Roles:

**Scrum Máster:** Nicolas Cadena.

**Product Owner:** Yasmani Tapia

**Team Scrum:** Harold Calderón y Juan Perdomo.

### Productos:

#### Sprint 1:

##### Product Sprint 1 -> R1, R2.

**R1:** Permitir registrar Productos (nombre, cantidad, valor unitario) en dos diferentes motores de bases de datos. El motor de base de datos 1 debe almacenar los productos con valor unitario superior a 100.000 pesos en caso contrario se debe almacenar en el motor

2. Utilizar el patrón Sigleton y Factory Method.

**R2:** Permitir visualizar todos los productos.

#### Sprint 2:

##### Product Sprint 2 -> R3, R4.

**R3:** En el mismo programa se desea permitir crear productos utilizando registros existentes en el inventario (Utilizar patrón prototype).

**R4:** Se quiere implementar una funcionalidad que permita visualizar la cantidad total de productos y el valor total de inventario. Cada uno debe ser un elemento de interfaz diferente, la cantidad de productos mostrarlo en una Etiqueta (JLabel) y el total de inventario en una caja de texto (JtextField). Los valores de estos productos se deben ver refrescados instantáneamente después de ingresar un nuevo producto. (Utilizar el patrón Observar).

**Eventos:**

Fecha inicio: 20/01/2024, 11:00 am.

Sprint 1:

**Harold Calderón****¿Qué se ha hecho?**

Las clases Main, cnxMySQL y cnxPostgreSql para realizar la conexión a bases de datos PostgreSQL y Mysql.

**¿Qué dificultades hay?**

La implementación de los patrones Singleton y Factory Method en la conexión.

**¿Qué queda por hacer?**

Insertar los datos en las bases de datos correspondientes según su valor unitario.

**Juan Perdomo****¿Qué se ha hecho?**

La instalación de dependencias necesarias y configuración de archivos del motor de bases de datos.

**¿Qué dificultades hay?**

La conexión de los datos realizado en cada patrón.

**¿Qué queda por hacer?**

Insertar los datos correspondientes según su valor en cada campo.

**Yasmani Tapia****¿Qué se ha hecho?**

Creación y diseño de la base de datos de PostgreSQL y Mysql.

**¿Qué dificultades hay?**

Problemas en el planteamiento de patrones de diseño.

**¿Qué queda por hacer?**

Insertar los datos correspondientes según su valor en cada campo.

## **Harold Calderón**

### **¿Qué se ha hecho?**

Los respectivos atributos y el método la clase CProducto para registrar los productos según su valor en las bases de datos.

### **¿Qué dificultades hay?**

Implementación del patrón Factory Method

### **¿Qué queda por hacer?**

El método para mostrar los productos en las bases de datos.

## **Juan Perdomo**

### **¿Qué se ha hecho?**

Se creó un formulario para el ingreso de datos.

### **¿Qué dificultades hay?**

La implementación del patrón de diseño de Singleton.

### **¿Qué queda por hacer?**

El método para mostrar los productos en las bases de datos.

## **Yasmani Tapia**

### **¿Qué se ha hecho?**

Se creo un método dentro la clase para visualizar los productos de la base de datos pertenecientes en la tabla

### **¿Qué dificultades hay?**

Realizar una función que clasifique los datos por cada columna

### **¿Qué queda por hacer?**

La visualización de cada producto que se ingresó en el formulario.

- **Review:** 23/01/2024, 7:00 pm a 08:00 pm.

### **¿Qué porcentaje del producto sprint se terminó?**

Se realizo un porcentaje de un 70%

Servidor: 127.0.0.1 » Base de datos: bdprueba » Tabla: persona

Examinar

Estructura

SQL

Buscar

Insertar
















Mostrando filas 0 - 4 (total de 5, La consulta tardó 0,0004 segundos.)

SELECT \* FROM `persona`

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Ver código SQL](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ | Filtrar filas:

Opciones extra

				id	cantidad	nombres	valuno
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	7	4	ARROZ	2300
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	8	4	NARANJAS	2345
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	9	3	PORTATIL	320000
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	10	4	TELEVISOR	4000000
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	11	5	MANGOS	500000

localhost/phpmyadmin/index.php?route=/table/structure&db=bdprueba&table=persona

phpMyAdmin

Reciente Favoritas

- Nueva
- bdprueba
  - Nueva
  - persona
- dbsem2
- demo
- information\_schema
- mysql
- performance\_schema
- phpmyadmin
- producto
- pruebas
- registro
- test
- uploadfile

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Operaciones Disparadores

Estructura de tabla Vista de relaciones

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
1	id	int(11)			No	Ninguna		AUTO_INCREMENT	<a href="#">Cambiar</a> <a href="#">Eliminar</a> <a href="#">Más</a>
2	cantidad	varchar(50)	utf8mb4_general_ci		Si	NULL			<a href="#">Cambiar</a> <a href="#">Eliminar</a> <a href="#">Más</a>
3	nombres	varchar(50)	utf8mb4_general_ci		Si	NULL			<a href="#">Cambiar</a> <a href="#">Eliminar</a> <a href="#">Más</a>
4	valuno	char(8)	utf8mb4_general_ci		Si	NULL			<a href="#">Cambiar</a> <a href="#">Eliminar</a> <a href="#">Más</a>
5	base	text	utf8mb4_general_ci		Si	NULL			<a href="#">Cambiar</a> <a href="#">Eliminar</a> <a href="#">Más</a>

☐ Seleccionar todo Para los elementos que están marcados: [Examinar](#) [Cambiar](#) [Eliminar](#) [Primaria](#) [Único](#) [Índice](#)

[Espacial](#) [Texto completo](#)

[Imprimir](#) [Planteamiento de la estructura de tabla](#) [Mover columnas](#) [Normalizar](#)

[Agregar](#) 1 columna(s) después de base [Continuar](#)

Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
<a href="#">Renombrar</a> <a href="#">Eliminar</a>	PRIMARY	BTREE	Si	No	id	5	A	No	

Consola

## Productos

Nombre

Cantidad

Valor Unit

Base de datos

[GUARDAR](#) [LISTAR](#) [GUARDAR CAMBIOS](#) [NUEVO](#) [BORRAR REGISTRO](#) [SALIR](#)

Id	Nombre	Cantidad	Valor Unitario	BD
----	--------	----------	----------------	----

```
public String getCantidad() {
```

Productos

Nombre

MECATOS

Cantidad

5

Valor Unit

3500

Base de datos

MYSQL

GUARDAR

LISTAR

NUEVO

GUARDAR CAMBIOS

BORRAR REGISTRO

SALIR

Id	Nombre	Cantidad	Valor Unitario	BD
----	--------	----------	----------------	----

**Productos**

Nombre

Cantidad

Valor Unit

Base de datos

GUARDAR

LISTAR

NUEVO

GUARDAR CAMBIOS

BORRAR REGISTRO

SALIR

Id	Nombre	Cantidad	Valor Unitario	BD
7	ARROZ	4	2300	POSGRESQL
8	NARANJAS	4	2345	MYSQL
9	PORTATIL	3	320000	POSGRETSQL
10	TELEVISOR	4	4000000	POSGRETSQL
11	MANGOS	5	500000	POSGRETSQL

- Insertar

```
public class ProductoDAO extends Producto {

    @Override
    public void Insertar() throws SQLException {
        PreparedStatement pstmt=null;
        Connection cnx=null;
        try{
            String sql="INSERT INTO persona(cantidad, nombres, valuno, base)VALUES(?,?,?,?) ";
            cnx=Conexion.Obtener();
            pstmt=cnx.prepareStatement(sql);
            pstmt.setString(1, this.getCantidad());
            pstmt.setString(2, this.getNombres());
            pstmt.setString(3, this.getValuno());
            pstmt.setString(4, this.getBase());
            pstmt.execute();
        }finally{
            if(pstmt!=null)
                pstmt.close();
        }
    }
}
```

- Listar

```
@Override
public List Listar() throws SQLException {
    PreparedStatement pstmt=null;
    ResultSet rs=null;
    Connection cnx=null;
    List<ProductoDAO> lista= new ArrayList<>();
    try{
        String sql="SELECT * FROM persona";
        cnx=Conexion.Obtener();
        pstmt=cnx.prepareStatement(sql);
        rs=pstmt.executeQuery();
        while(rs.next()){
            ProductoDAO oPrueba= new ProductoDAO();
            oPrueba.setId(rs.getInt(1));
            oPrueba.setNombres(rs.getString(2));
            oPrueba.setCantidad(rs.getString(3));
            oPrueba.setValuno(rs.getString(4));
            oPrueba.setBase(rs.getString(5));
            lista.add(oPrueba);
        }
        return lista;
    }finally{
        if(pstmt!=null)
            pstmt.close();

        if(rs!=null)
            rs.close();
    }
}
```



- Interfaz

**Productos**

Nombre

Cantidad

Valor Unit

Base de datos

GUARDAR

LISTAR

GUARDAR CAMBIOS

NUEVO

BORRAR REGISTRO

SALIR

Id	Nombre	Cantidad	Valor Unitario	BD
----	--------	----------	----------------	----