



# Formato de Tarea

## 1. Portada

- **Título de la tarea:** Aplicación web y bases de datos
- **Asignatura:** Web de desarrollo avanzado
- **Nombre del estudiante:** Jeffrey Manobando, Pamela Chipe, Richard Gualotuña, Jhordy Marcillo
- **Fecha de entrega:** 14 de febrero
- **Nombre del profesor o docente:** Ing. Doris Chicaiza
- **Universidad:** Universidad de fuerzas armadas ESPE

## 2. Objetivo General

El objetivo de este proyecto es desarrollar una aplicación web utilizando el patrón de arquitectura Modelo-Vista-Controlador (MVC) en Java con Spring Boot, que permita la gestión de estudiantes y sus calificaciones. La aplicación debe almacenar los datos en una base de datos MySQL y ofrecer funcionalidades como la inserción, consulta, filtrado y ordenación de los registros. Además, se debe integrar una interfaz web dinámica para interactuar con la información almacenada y proporcionar reportes en formato CSV y gráficos de análisis de calificaciones.

## 3. Desarrollo

Este proyecto se basa en el patrón de arquitectura MVC (Modelo-Vista-Controlador), el cual permite una separación clara de responsabilidades dentro de la aplicación. Spring Boot es el framework elegido para el desarrollo backend debido a su facilidad de configuración y su integración con bases de datos. La aplicación manejará datos de estudiantes y sus calificaciones, permitiendo el filtrado y ordenación de notas, generación de reportes y visualización de estadísticas mediante gráficos dinámicos.

## Herramientas de Desarrollo

- **Java y Spring Boot:** Framework utilizado para la implementación del backend y la lógica de negocio.
- **MySQL:** Base de datos utilizada para almacenar la información de los estudiantes y sus calificaciones.
- **Jackson Library:** Para el manejo de archivos JSON en el backend.
- **Bootstrap y JavaScript:** Utilizados en el frontend para mejorar la interfaz de usuario y la interactividad de la página.
- **Chart.js:** Librería de JavaScript utilizada para generar gráficos dinámicos basados en las calificaciones de los estudiantes.
- **CSV Export Library:** Permite la generación de reportes en formato CSV.



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

```
Project: mvc_service_crud_Chimba_Alexis
  > External Libraries
  > Scratches and Consoles

editar_usuario.html  UsuarioControlador.java
8 public class Estudiante {
36 }
37
38 > public Long getId() { return id; }
41
42 > public void setId(Long id) { this.id = id; }
45
46 > public String getNombre() { return nombre; }
49
50 > public void setNombre(String nombre) { this.nombre = nombre; }
53
54 > public String getApellido() { return apellido; }
57
58 > public void setApellido(String apellido) { this.apellido = apellido; }
61
62 > public String getEmail() { return email; }
65
66 > public void setEmail(String email) { this.email = email; }
69
70 > public Date getFechaNacimiento() { return fechaNacimiento; }
73
74 > public void setFechaNacimiento(Date fechaNacimiento) { this.fechaNacimiento = fechaNacimiento; }
77
78 > public List<Nota> getNotas() { return notas; }
81
82 > public void setNotas(List<Nota> notas) { this.notas = notas; }
85 }
```

lexis > mvc\_service\_crud > mvc\_service\_crud > src > main > java > com > example > mvc\_service\_crud > controlador > UsuarioControlador.java 1:9 CRLF UTF-8 4 spaces

```
BaseCalificacionesApplication.java  formulario.html
2 <html xmlns:th="http://www.thymeleaf.org">
7 </head>
8 <body>
10 <div class="container mt-4">
11 <h2>Registrar Estudiante</h2>
12 <form action="/estudiantes/guardar" method="post">
13 <div class="mb-3">
14 <label class="form-label">Nombre</label>
15 <input type="text" name="nombre" class="form-control" required>
16 </div>
17 <div class="mb-3">
18 <label class="form-label">Apellido</label>
19 <input type="text" name="apellido" class="form-control" required>
20 </div>
21 <div class="mb-3">
22 <label class="form-label">Email</label>
23 <input type="email" name="email" class="form-control" required>
24 </div>
25 <div class="mb-3">
26 <label class="form-label">Fecha de Nacimiento</label>
27 <input type="date" name="fechaNacimiento" class="form-control" required>
28 </div>
29 <div class="mb-3">
30 <label class="form-label">Nota</label>
31 <input type="number" step="0.1" name="nota" class="form-control" required>
32 </div>
33 </form>
34 </div>
35 </body>
36 </html>
```

templates > formulario.html 15:18 CRLF UTF-8 4 spaces



```
2 <html xmlns:th="http://www.thymeleaf.org">
8 <body>
9 <div class="container mt-4">
11 <table class="table table-striped">
12 <thead>
17 <th>Fecha de Nacimiento</th>
18 <th>Nota</th>
19 </tr>
20 </thead>
21 <tbody>
22 <tr th:each="estudiante : ${estudiantes}">
23 <td th:text="${estudiante.nombre}"></td>
24 <td th:text="${estudiante.apellido}"></td>
25 <td th:text="${estudiante.email}"></td>
26 <td th:text="${estudiante.fechaNacimiento}"></td>
27 <td th:text="${estudiante.notas[0].nota}"></td>
28 </tr>
29 </tbody>
30 </table>
31 </div>
32 </body>
33 </html>
```

## 4. Conclusión

El desarrollo de esta aplicación en Spring Boot con arquitectura MVC ha permitido implementar una solución eficiente para la gestión de estudiantes y sus calificaciones. La integración con MySQL y el uso de JSON garantizan una persistencia de datos confiable. Además, la implementación de filtros, ordenación y generación de reportes facilita la administración de la información. La interfaz desarrollada con Bootstrap y Chart.js mejora la experiencia del usuario al proporcionar visualizaciones claras y dinámicas. Este proyecto demuestra la importancia de aplicar buenas prácticas en el desarrollo web y el uso de herramientas adecuadas para la manipulación y visualización de datos.