



SENATI



PLAN DE TRABAJO DEL ESTUDIANTE

RDA
RECURSO DIDÁCTICO PARA EL APRENDIZAJE

RECURSO DIDÁCTICO PARA EL APRENDIZAJE

DATOS DEL ESTUDIANTE

Apellidos y Nombres:	Yataco Tasayco Fabian Alonso	ID:	001508063
Dirección Zonal/CFP:	ICA- AYACUCHO CFP CHINCHA		
Carrera:	Ing. De Software Con Inteligencia Artificial	Semestre:	V
Curso/ Mód. Formativo:	FULLSTACK DEVELOPER SOFTWARE		
Tema de Trabajo Final:	Implementación de un software web de gestión de inventario y venta de una botica		

1. INFORMACIÓN**▪ Identifica la problemática del caso práctico propuesto.**

Gestión manual del inventario: Provoca errores y dificulta el seguimiento preciso del stock.

Desabastecimientos frecuentes: Falta de alertas y monitoreo de niveles de inventario.

Tiempos de espera prolongados: Procesos manuales que ralentizan la atención al cliente.

Control ineficiente de ventas: Falta de un sistema centralizado para registrar y analizar transacciones.

▪ Identifica propuesta de solución y evidencias.**Implementación de software web centralizado:**

- Sistema de inventario automatizado con actualización en tiempo real
- Alertas automáticas para reposición de productos
- Registro digital de ventas
- Interfaz optimizada para atención al cliente

Beneficios esperados:

- Reducción de errores en gestión de stock
- Eliminación de desabastecimientos mediante alertas preventivas
- Agilización del proceso de atención al cliente
- Mejora en la toma de decisiones basada en datos
- Mayor competitividad en el mercado farmacéutico

▪ **Respuestas a preguntas guía**

Durante el análisis y estudio del caso práctico, debes obtener las respuestas a las interrogantes:

Pregunta 01:	¿Cuáles son los principales desafíos operativos que enfrenta la botica "FarmaPlus" debido a la falta de un sistema de gestión de inventarios automatizado?
FarmaPlus enfrenta errores frecuentes en su inventario manual, causando desabastecimientos inesperados y tiempos de espera prolongados. La dificultad para monitorear fechas de vencimiento aumenta las pérdidas, mientras que la imposibilidad de acceder a información en tiempo real obstaculiza la toma de decisiones estratégicas.	
Pregunta 02:	¿Qué características específicas debe incluir el software web para atender las necesidades de control de stock y atención al cliente en "FarmaPlus"?
El software debe incluir actualización en tiempo real del inventario, alertas para niveles críticos de stock y productos próximos a vencer, interfaz de punto de venta intuitiva y búsqueda rápida de productos. También necesita gestión automatizada de proveedores, registro de clientes con historial de compras y generación de reportes personalizados.	
Pregunta 03:	¿De qué manera la implementación de un software web para la gestión de inventarios puede impactar en la reducción del desabastecimiento y la mejora del servicio al cliente?
El software permitirá anticipar la demanda mediante alertas antes del desabastecimiento, reducirá tiempos de atención y eliminará errores en el conteo. Los clientes recibirán información precisa sobre disponibilidad y fechas de reposición, mientras que el análisis de patrones históricos optimizará los niveles de inventario.	
Pregunta 04:	¿Qué beneficios adicionales, aparte del control de inventarios, podría ofrecer el software para incrementar la competitividad de "FarmaPlus" en su mercado?
Más allá del inventario, el software puede implementar un programa de fidelización, análisis de tendencias de consumo, gestión de prescripciones recurrentes y recordatorios para medicamentos crónicos. La integración con delivery, consultas farmacéuticas online y manejo de promociones personalizadas posicionarán a FarmaPlus como una farmacia moderna y centrada en el cliente.	
Pregunta 05:	¿Qué métricas se pueden utilizar para evaluar la efectividad del software en mejorar los tiempos de respuesta al cliente y reducir los errores en la gestión de inventario?
Más allá del inventario, el software puede implementar un programa de fidelización, análisis de tendencias de consumo, gestión de prescripciones recurrentes y recordatorios para medicamentos crónicos. La integración con delivery, consultas farmacéuticas online y manejo de promociones personalizadas posicionarán a FarmaPlus como una farmacia moderna y centrada en el cliente.	

2. PLANIFICACIÓN DEL TRABAJO

▪ **Cronograma de actividades:**

N°	ACTIVIDADES	CRONOGRAMA					
1	Desarrollo del crud usando node js	01/05/2025					

▪ **Lista de recursos necesarios:**

1. MÁQUINAS Y EQUIPOS	
Descripción	Cantidad
LAPTOP	1

2. HERRAMIENTAS E INSTRUMENTOS	
Descripción	Cantidad
Navegador web	--
Internet	--

3. MATERIALES E INSUMOS	
Descripción	Cantidad
Documentaciones	--
Artículos	--

3. DECIDIR PROPUESTA

- Describe la propuesta determinada para la solución del caso práctico

PROPUESTA DE SOLUCIÓN

Solución Propuesta: Sistema de Gestión Integral FarmaSys

Proponemos implementar "FarmaSys", una plataforma web escalable diseñada específicamente para farmacias con las siguientes características:

1. Gestión Inteligente de Inventario

- Actualización automática en tiempo real tras cada venta
- Sistema predictivo de abastecimiento basado en históricos de consumo
- Alertas configurables para niveles mínimos de stock y fechas de vencimiento

2. Optimización de Atención al Cliente

- Interfaz de punto de venta rápida con búsqueda inteligente de productos
- Visualización inmediata de disponibilidad, ubicación y alternativas terapéuticas
- Historial clínico básico de clientes frecuentes con recordatorios de medicación

3. Automatización de Procesos Administrativos

- Generación de órdenes de compra basadas en niveles de inventario
- Dashboard ejecutivo con indicadores clave de rendimiento
- Integración con sistemas de facturación electrónica

La implementación seguirá un enfoque gradual en tres fases: configuración inicial (2 semanas), capacitación del personal (1 semana) y monitoreo post-implementación (1 mes).

Los resultados esperados incluyen reducción del 90% en errores de inventario, disminución del 70% en casos de desabastecimiento y reducción del 40% en tiempos de espera, mejorando significativamente la experiencia del cliente y la rentabilidad del negocio.

4. EJECUTAR

- Resolver el caso práctico, utilizando como referencia el problema propuesto y las preguntas guía proporcionadas para orientar el desarrollo.
- Fundamentar sus propuestas en los conocimientos adquiridos a lo largo del curso, aplicando lo aprendido en las tareas y operaciones descritas en los contenidos curriculares.

INSTRUCCIONES: Ser lo más explícito posible. Los gráficos ayudan a transmitir mejor las ideas. Tomar en cuenta los aspectos de calidad, medio ambiente y SHI.

[illegible]

Para empezar a realizar el proyecto, debemos tener una base de datos. En este caso, será para una tienda de videojuegos cuyo objetivo es implementar un CRUD que esté configurado con un catálogo para realizar cambios en todo momento, ya sea de imagen, precio, etc.

```
1 CREATE DATABASE tiendaJuegos;
2 USE tiendaJuegos;
3
4 CREATE TABLE categoria(
5     idcategoria INT PRIMARY KEY AUTO_INCREMENT,
6     nomcategoria VARCHAR(30)
7 ) ENGINE = INNODB;
8
9 CREATE TABLE plataformas(
10     idplataforma INT PRIMARY KEY AUTO_INCREMENT,
11     nomplataforma VARCHAR(35)
12 )ENGINE = INNODB;
13
14
15 CREATE TABLE juegos(
16     idjuego INT PRIMARY KEY AUTO_INCREMENT,
17     nomjuego VARCHAR(50) NOT NULL,
18     precio DECIMAL(10,2) NOT NULL,
19     clasificacion ENUM('Todos', 'Todos +10', 'Adolescentes', 'Maduro +17', 'Adulto') NOT NULL,
20     descripcion TEXT NOT NULL,
21     editor VARCHAR(40) NOT NULL,
22     referenciaimg VARCHAR(250),
23     idcategoria INT,
24     idplataforma INT,
25     FOREIGN KEY (idcategoria) REFERENCES categoria(idcategoria),
26     FOREIGN KEY (idplataforma) REFERENCES plataformas(idplataforma)
27 ) ENGINE = INNOD
```

```
INSERT INTO categoria (nomcategoria) VALUES
('Sandbox'),
('Estrategia'),
('Shooter'),
('Batalla multijugador'),
('RPG'),
('Simulación y deporte'),
('Puzzles'),
('Acción-aventura'),
('Supervivencia y terror');

INSERT INTO plataformas (nomplataforma) VALUES
('PlayStation'),
('Xbox'),
('Nintendo Switch'),
('Teléfono'),
('Multi-plataforma');
```

Este código, en específico, nos será de ayuda al momento de listar los juegos en la página, ya que unirá la tabla de categorías con la tabla de juegos, dando un resultado más ordenado para el usuario.

```
SELECT
  j.idjuego,
  j.nomjuego,
  j.precio,
  j.clasificacion,
  j.descripcion,
  j.editor,
  j.referenciaimg,
  c.nomcategoria AS categoria,
  p.nomplataforma AS plataforma
FROM juegos j
JOIN categoria c ON j.idcategoria = c.idcategoria
JOIN plataformas p ON j.idplataforma = p.idplataforma;
```

```
INSERT INTO juegos (
  nomjuego, precio, clasificacion, descripcion, editor, referenciaimg, idcategoria, idplataforma
) VALUES
('The Legend of Zelda: Breath of the Wild', 59.99, 'Todos +10',
'Aventura épica en un mundo abierto donde exploras y resuelves acertijos.', 'Nintendo',
'/img/zelda_botw.jpg', 8, 3),

('FIFA 24', 69.99, 'Todos',
'Simulador de fútbol con licencias oficiales, equipos y jugadores reales.', 'EA Sports',
'/img/fifa24.jpg', 6, 5),

('Call of Duty: Modern Warfare III', 69.99, 'Maduro +17',
'Shooter en primera persona con campaña intensa y multijugador competitivo.', 'Activision',
'/img/cod_mw3.jpg', 3, 5);
```

Luego configuramos el archivo app.js para manejar las plantillas, configurar las rutas y manejar los errores.


```
1  const express = require('express');
2  const bodyParser = require('body-parser');
3  const path = require('path');
4  const multer = require('multer');
5
6  // Acceso a rutas
7  const rutaJuego = require('./routes/juegos');
8
9  // Iniciar la app
10 const app = express();
11 const PORT = process.env.PORT || 3000;
12
13 // Configurar "middleware" => "capa de comunicación"
14 app.use(bodyParser.urlencoded({ extended: true }));
15 app.use(bodyParser.json());
16 app.use(express.static(path.join(__dirname, 'public')));
17
18 // Crear la carpeta de uploads si no existe
19 const fs = require('fs');
20 const uploadsDir = path.join(__dirname, 'public/uploads');
21 if (!fs.existsSync(uploadsDir)) {
22   fs.mkdirSync(uploadsDir, { recursive: true });
23 }
24
25 // Motor de plantillas
26 app.set('view engine', 'ejs');
27 app.set('views', path.join(__dirname, 'views'));
```

```
29 // Configuración de rutas
30 app.use('/', rutaJuego); // Usar la ruta para manejar los juegos
31 // app.use('/api/categorias', rutaCategoria); // Descomentar si tienes rutas para las categorías
32
33 // Manejo de errores generales
34 app.use((err, req, res, next) => {
35   console.error(err.stack);
36   res.status(500).send('¡Algo salió mal!');
37 });
38
39 // Servidor Web
40 app.listen(PORT, () => {
41   console.log(`Servidor iniciado en http://localhost:${PORT}`);
42 });
```

DIBUJO / ESQUEMA / DIAGRAMA DE PROPUESTA

(Adicionar las páginas que sean necesarias)

Para luego pasar al archivo enrutador donde configuraremos las consultas básicas como son crear, guardar y editar agregando otra función utilizando “Multer” para guardar las imágenes y se muestre en nuestro catálogo.

```
1  const express = require('express')
2  const router = express.Router()
3  const db = require('../config/database')
4  const multer = require('multer')
5  const path = require('path')
6  const fs = require('fs')
7
8  // Configuración de almacenamiento para Multer
9  const storage = multer.diskStorage({
10   destination: function (req, file, cb) {
11     // Directorio donde se guardarán las imágenes
12     cb(null, 'public/uploads/')
13   },
14   filename: function (req, file, cb) {
15     // Generar un nombre único para el archivo
16     const uniqueSuffix = Date.now() + '-' + Math.round(Math.random() * 1E9)
17     const extension = path.extname(file.originalname)
18     cb(null, 'game-' + uniqueSuffix + extension)
19   }
20 })
```

```
22 // Filtro para solo permitir imágenes
23 const fileFilter = (req, file, cb) => {
24   // Comprobar que el tipo de archivo es una imagen
25   if (file.mimetype.startsWith('image/')) {
26     cb(null, true)
27   } else {
28     cb(new Error('El archivo debe ser una imagen (jpg, png, gif)'), false)
29   }
30 }
31
32 // Configuración de Multer
33 const upload = multer({
34   storage: storage,
35   fileFilter: fileFilter,
36   limits: {
37     fileSize: 5 * 1024 * 1024 // límite de 5MB
38   }
39 })
```

```
40
41 // Mostrar todos los juegos
42 router.get('/', async (req, res) => {
43   try {
44     const query = `
45       SELECT
46         j.idjuego,
47         j.nomjuego,
48         j.precio,
49         j.clasificacion,
50         j.descripcion,
51         j.editor,
52         j.referenciaimg,
53         c.nomcategoria AS categoria,
54         p.nomplataforma AS plataforma
55       FROM juegos j
56       JOIN categoria c ON j.idcategoria = c.idcategoria
57       JOIN plataformas p ON j.idplataforma = p.idplataforma
58     `
59     const [juegos] = await db.query(query)
60     res.render('index', { juegos, path: '/' })
61   } catch (error) {
62     console.error(error)
63     res.status(500).send('Error al cargar los juegos')
64   }
65 })
```

```
66
67 // Formulario para crear nuevo juego
68 router.get('/create', async (req, res) => {
69   try {
70     const [categorias] = await db.query('SELECT * FROM categoria')
71     const [plataformas] = await db.query('SELECT * FROM plataformas')
72     res.render('create', { categorias, plataformas, path: '/create' })
73   } catch (error) {
74     console.error(error)
75     res.status(500).send('Error al cargar el formulario')
76   }
77 })
```

```

78
79 // Guardar nuevo juego con imagen
80 router.post('/create', upload.single('imagen'), async (req, res) => {
81   try {
82     const { nomjuego, precio, clasificacion, descripcion, editor, idcategoria, idplataforma } = req.body
83
84     // Obtener la ruta de la imagen si se subió una, o null si no
85     const referenciaimg = req.file ? `/uploads/${req.file.filename}` : null
86
87     await db.query(
88       `INSERT INTO juegos (nomjuego, precio, clasificacion, descripcion, editor, referenciaimg, idcategoria, idplataforma)
89       VALUES (?, ?, ?, ?, ?, ?, ?, ?)`,
90       [nomjuego, precio, clasificacion, descripcion, editor, referenciaimg, idcategoria, idplataforma]
91     )
92     res.redirect('/')
93   } catch (error) {
94     console.error(error)
95     res.status(500).send('Error al crear el juego')
96   }
97 })

```

```

99 // Formulario de edición
100 router.get('/edit/:id', async (req, res) => {
101   try {
102     const [categorias] = await db.query('SELECT * FROM categoria')
103     const [plataformas] = await db.query('SELECT * FROM plataformas')
104     const [registro] = await db.query('SELECT * FROM juegos WHERE idjuego = ?', [req.params.id])
105
106     if (registro.length > 0)
107       res.render('edit', { juego: registro[0], categorias, plataformas, path: '/edit' })
108     else
109       res.redirect('/')
110   } catch (error) {
111     console.error(error)
112     res.status(500).send('Error al cargar el formulario de edición')
113   }
114 })

```

```

116 // Guardar edición con imagen
117 router.post('/edit/:id', upload.single('imagen'), async (req, res) => {
118   try {
119     const { nomjuego, precio, clasificacion, descripcion, editor, idcategoria, idplataforma } = req.body
120
121     // Si hay un archivo nuevo, usarlo; si no, mantener la referencia existente
122     let referenciaimg
123
124     if (req.file) {
125       referenciaimg = `/uploads/${req.file.filename}`
126
127       // Opcionalmente: eliminar la imagen anterior si existe
128       const [juegoAnterior] = await db.query('SELECT referenciaimg FROM juegos WHERE idjuego = ?', [req.params.id])
129       if (juegoAnterior.length > 0 && juegoAnterior[0].referenciaimg) {
130         const rutaAnterior = path.join(__dirname, '..', 'public', juegoAnterior[0].referenciaimg)
131         // Verificar si el archivo existe antes de intentar eliminarlo
132         if (fs.existsSync(rutaAnterior)) {
133           fs.unlinkSync(rutaAnterior)
134         }
135       }
136     } else {
137       // Mantener la imagen existente
138       const [juegoActual] = await db.query('SELECT referenciaimg FROM juegos WHERE idjuego = ?', [req.params.id])
139       referenciaimg = juegoActual[0].referenciaimg
140     }
141

```

```
42     await db.query(  
43       'UPDATE juegos SET nomjuego=?, precio=?, clasificacion=?, descripcion=?, editor=?, referenciaimg=?, idcategoria=?  
44       WHERE idjuego=?',  
45       [nomjuego, precio, clasificacion, descripcion, editor, referenciaimg, idcategoria, idplataforma, req.params.id]  
46     )  
47     res.redirect('/')  
48   } catch (error) {  
49     console.error(error)  
50     res.status(500).send('Error al actualizar el juego')  
51   }  
52 })  
53
```

```
// Eliminar juego  
router.get('/delete/:id', async (req, res) => {  
  try {  
    // Obtener información del juego para eliminar también la imagen  
    const [juego] = await db.query('SELECT referenciaimg FROM juegos WHERE idjuego = ?', [req.params.id])  
  
    // Eliminar el juego de la base de datos  
    await db.query('DELETE FROM juegos WHERE idjuego = ?', [req.params.id])  
  
    // Si el juego tenía una imagen, eliminarla del sistema de archivos  
    if (juego.length > 0 && juego[0].referenciaimg) {  
      const rutaImagen = path.join(__dirname, '..', 'public', juego[0].referenciaimg)  
      if (fs.existsSync(rutaImagen)) {  
        fs.unlinkSync(rutaImagen)  
      }  
    }  
  
    res.redirect('/')  
  } catch (error) {  
    console.error(error)  
    res.status(500).send('Error al eliminar el juego')  
  }  
})
```

```

1 //
2 //Ruta para ver el catálogo de juegos
3 // Esta ruta permite filtrar por categoría, orden y búsqueda
4 // y renderiza la vista del catálogo con los resultados obtenidos.
5 router.get('/catalogo', async (req, res) => {
6   try {
7     // Obtener filtros de la URL
8     const { categoria, orden, busqueda } = req.query;
9
10    // Consulta base
11    let query = `
12      SELECT
13        j.idjuego,
14        j.nomjuego,
15        j.precio,
16        j.clasificacion,
17        j.descripcion,
18        j.editor,
19        j.referenciaimg,
20        c.idcategoria,
21        c.nomcategoria AS categoria,
22        p.nomplataforma AS plataforma
23    FROM juegos j
24    JOIN categoria c ON j.idcategoria = c.idcategoria
25    JOIN plataformas p ON j.idplataforma = p.idplataforma
26    WHERE 1=1
27  `;
28
29

```

```

205 // Parámetros para la consulta
206 const params = [];
207
208 // Aplicar filtro de categoría
209 if (categoria && categoria !== 'todas') {
210   query += ' AND j.idcategoria = ?';
211   params.push(categoria);
212 }
213
214 // Aplicar filtro de búsqueda
215 if (busqueda && busqueda.trim() !== '') {
216   const terminoBusqueda = `%${busqueda.trim()}%`;
217   query += ' AND (j.nomjuego LIKE ? OR j.editor LIKE ? OR j.descripcion LIKE ?)';
218   params.push(terminoBusqueda, terminoBusqueda, terminoBusqueda);
219 }
220

```

```
// Aplicar orden
if (orden === 'precio_desc') {
  query += ' ORDER BY j.precio DESC';
} else if (orden === 'precio_asc') {
  query += ' ORDER BY j.precio ASC';
} else if (orden === 'nombre_asc') {
  query += ' ORDER BY j.nomjuego ASC';
} else if (orden === 'nombre_desc') {
  query += ' ORDER BY j.nomjuego DESC';
} else {
  // Orden por defecto
  query += ' ORDER BY j.idjuego DESC';
}

// Ejecutar consulta
const [juegos] = await db.query(query, params);

// Obtener todas las categorías para el filtro
const [categorias] = await db.query('SELECT * FROM categoria');
```

```
// Renderizar vista
res.render('catalogo', {
  juegos,
  categorias,
  filtros: {
    categoria: categoria || 'todas',
    orden: orden || 'recientes',
    busqueda: busqueda || ''
  },
  path: '/catalogo'
});

} catch (error) {
  console.error('Error al cargar el catálogo:', error);
  res.status(500).send('Error al cargar el catálogo de juegos');
}
});
```

```

258
259 // Manejo de errores de Multer
260 router.use((err, req, res, next) => {
261   if (err instanceof multer.MulterError) {
262     // Error de Multer
263     if (err.code === 'LIMIT_FILE_SIZE') {
264       return res.status(400).send('El archivo es demasiado grande (máximo 5MB)')
265     }
266     return res.status(400).send('Error al subir el archivo: ' + err.message)
267   } else if (err) {
268     // Otro tipo de error
269     return res.status(500).send(err.message)
270   }
271   next()
272 })
273
274 module.exports = router

```

Luego de configurar lo que son las rutas y las consultas para que el CRUD sea funcional al 100% pasamos a la vista donde se mostrara la tabla junto con el catálogo de forma dinámica. Para que el usuario tenga la facilidad de realizar consultas y crear nuevos productos para el catálogo.

```

1  <%- include('partials/header') %>
2
3  <div class="table-responsive">
4    <table class="table table-sm">
5      <thead>
6        <tr>
7          <th>ID</th>
8          <th>Nombre</th>
9          <th>Precio</th>
10         <th>Clasificación</th>
11         <th>Editor</th>
12         <th>Categoría</th>
13         <th>Plataforma</th>
14         <th>Operaciones</th>
15       </tr>
16     </thead>
17     <tbody>
18       <% juegos.forEach(juego => { %>
19         <tr>
20           <td><%= juego.idjuego %></td>
21           <td><%= juego.nomjuego %></td>
22           <td><%= juego.precio %></td>
23           <td><%= juego.clasificacion %></td>
24           <td><%= juego.editor %></td>
25           <td><%= juego.categoria %></td>
26           <td><%= juego.plataforma %></td>
27         <td>

```



```

27         <td>
28             <a href="/edit/<%= juego.idjuego %>" class="btn btn-sm btn-info">Editar</a>
29             <a href="/delete/<%= juego.idjuego %>" class="btn btn-sm btn-danger delete">Eliminar</a>
30         </td>
31     </tr>
32     <% }) %>
33 </tbody>
34 </table>
35 </div>
36
37 <script src="https://cdn.jsdelivr.net/npm/sweetalert2@11"></script>
38 <script>
39     document.addEventListener("DOMContentLoaded", () => {
40         const enlacesEliminar = document.querySelectorAll(".delete");
41
42         enlacesEliminar.forEach(enlace => {
43             enlace.addEventListener("click", (event) => {
44                 event.preventDefault();
45

```

```

        enlacesEliminar.forEach(enlace => {
            enlace.addEventListener("click", (event) => {
                event.preventDefault();

                Swal.fire({
                    title: 'Juegos',
                    text: '¿Está seguro de eliminar este juego?',
                    icon: 'question',
                    footer: 'Tienda de Juegos - Proyecto Node.js',
                    confirmButtonText: 'Aceptar',
                    confirmButtonColor: '#2980b9',
                    showCancelButton: true,
                    cancelButtonText: 'Cancelar',
                    cancelButtonColor: '#c0392b'
                }).then((result) => {
                    if(result.isConfirmed) {
                        window.location.href = event.target.getAttribute("href");
                    }
                });
            });
        });
    });
</script>

<%= include('partials/footer') %>

```

🛒 Tienda de Juegos 📋 Lista 📖 Catálogo ➕ Nuevo Juego

ID	Nombre	Precio	Clasificación	Editor	Categoría	Plataforma	Operaciones
1	The Legend of Zelda: Breath of the Wild	59.99	Todos +10	Nintendo	Acción-aventura	Nintendo Switch	Editar Eliminar
2	FIFA 24	69.99	Todos	EA Sports	Simulación y deporte	Multi-plataforma	Editar Eliminar
3	Call of Duty: Modern Warfare III	69.99	Maduro +17	Activision	Shooter	Multi-plataforma	Editar Eliminar
8	CRASH	45.99	Adolescentes	SEGA	RPG	Teléfono	Editar Eliminar

CREAR NUEVO JUEGO

```

1  <%- include('partials/header') %>
2
3  <form action="/create" id="formulario-registro" autocomplete="off" method="post" enctype="multipart/form-data">
4    <div class="card mt-2">
5      <div class="card-header"><strong>Registrar Juego</strong></div>
6
7      <div class="card-body">
8
9        <div class="row g-2">
10         <div class="col-md-6 mb-2">
11           <div class="form-floating">
12             <input type="text" id="nomjuego" name="nomjuego" class="form-control" placeholder="Nombre del juego">
13             <label for="nomjuego">Nombre del juego</label>
14           </div>
15         </div>
16
17         <div class="col-md-6 mb-2">
18           <div class="form-floating">
19             <input type="number" step="0.01" id="precio" name="precio" class="form-control" placeholder="Precio">
20             <label for="precio">Precio</label>
21           </div>
22         </div>
23       </div>
24
25       <div class="row g-2">
26         <div class="col-md-6 mb-2">
27           <div class="form-floating">
28             <select name="clasificacion" id="clasificacion" class="form-select" required>
29               <option value="" selected disabled>Selecione</option>
30               <option value="Todos">Todos</option>
31               <option value="Todos +10">Todos +10</option>
32               <option value="Adolescentes">Adolescentes</option>
33               <option value="Maduro +17">Maduro +17</option>
34               <option value="Adulto">Adulto</option>
35             </select>
36             <label for="clasificacion">Clasificación</label>
37           </div>
38         </div>
39
40         <div class="col-md-6 mb-2">
41           <div class="form-floating">
42             <input type="text" id="editor" name="editor" class="form-control" placeholder="Editor del juego">
43             <label for="editor">Editor</label>
44           </div>
45         </div>
46       </div>
47     </div>
48   </form>

```

```

<div class="row g-2">
  <div class="col-md-6 mb-2">
    <div class="form-floating">
      <select name="idcategoria" id="idcategoria" class="form-select" required>
        <option value="" selected disabled>Selecione</option>
        <% categorias.forEach(cat => { %>
          <option value="<%= cat.idcategoria %>"><%= cat.nomcategoria %></option>
        <% }) %>
      </select>
      <label for="idcategoria">Categoría</label>
    </div>
  </div>

  <div class="col-md-6 mb-2">
    <div class="form-floating">
      <select name="idplataforma" id="idplataforma" class="form-select" required>
        <option value="" selected disabled>Selecione</option>
        <% plataformas.forEach(plat => { %>
          <option value="<%= plat.idplataforma %>"><%= plat.nomplataforma %></option>
        <% }) %>
      </select>
      <label for="idplataforma">Plataforma</label>
    </div>
  </div>

```

```

<div class="mb-2">
  <div class="form-floating">
    <textarea name="descripcion" id="descripcion" class="form-control" placeholder="Descripción del juego">
    <label for="descripcion">Descripción</label>
  </div>
</div>

<div class="mb-2">
  <label for="imagen" class="form-label">Imagen del juego</label>
  <input type="file" id="imagen" name="imagen" class="form-control" accept="image/*">
  <div class="form-text">Formatos permitidos: JPG, PNG, GIF (Máximo 5MB)</div>
</div>
</div>

```

Registrar Juego

Nombre del juego

Precio

Clasificación
Selecione

Editor

Categoría
Selecione

Plataforma
Selecione

Descripción

Imagen del juego

Seleccionar archivo

Sin archivos seleccionados

Formatos permitidos: JPG, PNG, GIF (Máximo 5MB)

Guardar

Cancelar

EDITAR JUEGO

```

1 <%= include('partials/header') %>
2
3 <form action="/edit/<%= juego.idjuego %>" id="formulario-edicion" autocomplete="off" method="post">
4   <div class="card mt-2">
5     <div class="card-header"><strong>Editar Juego</strong></div>
6
7     <div class="card-body">
8
9       <div class="row g-2">
10        <div class="col-md-6 mb-2">
11          <div class="form-floating">
12            <input type="text" id="nomjuego" name="nomjuego" class="form-control" value="<%= juego.nombre %>">
13            <label for="nomjuego">Nombre del juego</label>
14          </div>
15        </div>
16
17        <div class="col-md-6 mb-2">
18          <div class="form-floating">
19            <input type="number" step="0.01" id="precio" name="precio" class="form-control" value="<%= juego.precio %>">
20            <label for="precio">Precio</label>
21          </div>
22        </div>
23      </div>
24    </div>
  </div>
  </form>

```

```

<div class="row g-2">
  <div class="col-md-6 mb-2">
    <div class="form-floating">
      <select name="clasificacion" id="clasificacion" class="form-select" required>
        <option value="" disabled>Selecione</option>
        <option value="Todos" <%= juego.clasificacion === 'Todos' ? 'selected' : '' %>>Todos</option>
        <option value="Todos +10" <%= juego.clasificacion === 'Todos +10' ? 'selected' : '' %>>Todos +10</option>
        <option value="Adolescentes" <%= juego.clasificacion === 'Adolescentes' ? 'selected' : '' %>>Adolescentes</option>
        <option value="Maduro +17" <%= juego.clasificacion === 'Maduro +17' ? 'selected' : '' %>>Maduro +17</option>
        <option value="Adulto" <%= juego.clasificacion === 'Adulto' ? 'selected' : '' %>>Adulto</option>
      </select>
      <label for="clasificacion">Clasificación</label>
    </div>
  </div>

  <div class="col-md-6 mb-2">
    <div class="form-floating">
      <input type="text" id="editor" name="editor" class="form-control" value="<%= juego.editor %>" />
      <label for="editor">Editor</label>
    </div>
  </div>
</div>

```

```

</div>

<div class="row g-2">
  <div class="col-md-6 mb-2">
    <div class="form-floating">
      <select name="idcategoria" id="idcategoria" class="form-select" required>
        <option value="" disabled>Selecione</option>
        <%= categorias.forEach(cat => { %>
          <option value="<%= cat.idcategoria %>" <%= juego.idcategoria == cat.idcategoria ? 'selected' : '' %>><%= cat.nombre %></option>
        %>
      </select>
      <label for="idcategoria">Categoría</label>
    </div>
  </div>

```

```

<div class="mb-2">
  <div class="form-floating">
    <textarea name="descripcion" id="descripcion" class="form-control" placeholder="Descripción del juego" />
    <label for="descripcion">Descripción</label>
  </div>
</div>

<div class="mb-2">
  <label for="imagen" class="form-label">Imagen del juego</label>
  <input type="file" id="imagen" name="imagen" class="form-control" accept="image/*" />
  <div class="form-text">Formatos permitidos: JPG, PNG, GIF (Máximo 5MB)</div>

  <%= if (juego.referenciaimg) { %>
    <div class="mt-2">
      <p>Imagen actual:</p>
      " style="max-width: 200px; max-height: 100px;" />
    </div>
  <%= } %>
</div>

```

Editar Juego


Nombre del juego CRASH	Precio 45.99
Clasificación Adolescentes	Editor SEGA
Categoría RPG	Plataforma Teléfono
Descripción juego de carreras	

Imagen del juego

Seleccionar archivo Sin archivos seleccionados

Formatos permitidos: JPG, PNG, GIF (Máximo 5MB)

Imagen actual:



Actualizar Cancelar

CATÁLOGO

```

1 <%- include('partials/header') %>
2
3 > <style>...
80 </style>
81 <div class="container">
82   <h1 class="my-4">Catálogo de Juegos</h1>
83
84   <!-- Filtros y buscador -->
85   <div class="card mb-4">
86     <div class="card-header text-white">
87       <i class="fas fa-filter me-2"></i> Filtros
88     </div>
89     <div class="card-body">
90       <form action="/catalogo" method="get" id="filtrosForm">
91         <div class="row g-3">
92           <!-- Buscador -->
93           <div class="col-md-5">
94             <div class="input-group">
95               <input
96                 type="text"
97                 class="form-control"
98                 placeholder="Buscar por nombre, editor, descripción..."
99                 name="busqueda"
100                 value="<%= filtros.busqueda %>"
101               >
102               <button class="btn btn-outline-primary" type="submit">
103                 <i class="fas fa-search"></i> Buscar
104               </button>
105             </div>
106           </div>

```

```

107
108   <!-- Filtro de categoría -->
109   <div class="col-md-3">
110     <select class="form-select" name="categoria" onchange="this.form.submit()">
111       <option value="todas" <%= filtros.categoria === 'todas' ? 'selected' : '' %>>Todas las categorías</option>
112       <% categorias.forEach(cat => { %>
113         <option value="<%= cat.idcategoria %>" <%= filtros.categoria == cat.idcategoria ? 'selected' : '' %>>
114           <%= cat.nomcategoria %>
115         </option>
116       <% }) %>
117     </select>
118   </div>
119
120   <!-- Ordenamiento -->
121   <div class="col-md-3">
122     <select class="form-select" name="orden" onchange="this.form.submit()">
123       <option value="recientes" <%= filtros.orden === 'recientes' ? 'selected' : '' %>>Más recientes</option>
124       <option value="precio_desc" <%= filtros.orden === 'precio_desc' ? 'selected' : '' %>>Precio: Mayor a menor</option>
125       <option value="precio_asc" <%= filtros.orden === 'precio_asc' ? 'selected' : '' %>>Precio: Menor a mayor</option>
126       <option value="nombre_asc" <%= filtros.orden === 'nombre_asc' ? 'selected' : '' %>>Nombre: A-Z</option>
127       <option value="nombre_desc" <%= filtros.orden === 'nombre_desc' ? 'selected' : '' %>>Nombre: Z-A</option>
128     </select>
129   </div>
130

```

```

130
131     <!-- Botón limpiar filtros -->
132     <div class="col-md-1">
133         <a href="/catalogo" class="btn btn-outline-secondary w-100" title="Limpiar filtros">
134             <i class="fas fa-sync-alt"></i>
135         </a>
136     </div>
137 </div>
138 </form>
139 </div>
140 </div>
141

```

```

141
142 <!-- Resultados -->
143 <div class="row">
144     <% if (juegos.length === 0) { %>
145         <div class="col-12 text-center py-5">
146             <div class="alert alert-info">
147                 <h4>No se encontraron juegos</h4>
148                 <p>Intenta con diferentes criterios de búsqueda o <a href="/catalogo">ver todos los juegos</a>.</p>
149             </div>
150         </div>
151     <% } else { %>
152         <!-- Contador de resultados -->
153         <div class="col-12 mb-3">
154             <p class="text-muted">Mostrando <%= juegos.length %> resultados</p>
155         </div>
156

```

```

<!-- Tarjetas de juegos -->
<% juegos.forEach(juego => { %>
    <div class="col-md-4 col-lg-3 mb-4">
        <div class="card h-100" data-id="<%= juego.idjuego %>" data-descripcion="<%= juego.descripcion %>">
            <!-- Imagen del juego -->
            <div class="card-img-top-container">
                <% if (juego.referenciaimg) { %>
                    " sty
                <% } else { %>
                    <div class="text-center p-4 text-muted">
                        <i class="fas fa-gamepad fa-3x"></i>
                        <p class="mt-2 small">Sin imagen</p>
                    </div>
                <% } %>
            </div>
        </div>
    <% } %>
} %>

```



```

<div class="card-body">
  <!-- Etiqueta de categoría -->
  <div class="mb-2">
    <span class="badge bg-primary"><%= juego.categoria %></span>
    <span class="badge bg-secondary"><%= juego.plataforma %></span>
  </div>

  <!-- Título y precio -->
  <h5 class="card-title"><%= juego.nomjuego %></h5>
  <h6 class="text-primary fw-bold mb-2">$<%= parseFloat(juego.precio).toFixed(2) %></h6>

  <!-- Editor y clasificación -->
  <p class="card-text text-muted small">
    <span class="d-block">Editor: <%= juego.editor %></span>
    <span class="d-block">Clasificación: <%= juego.clasificacion %></span>
  </p>

  <!-- Descripción corta -->
  <p class="card-text">
    <%= juego.descripcion.length > 100 ? juego.descripcion.substring(0, 100) + '...'
  </p>
</div>

```

```





<!-- Botones de acción -->
<div class="card-footer bg-white">
  <div class="d-flex justify-content-between">
    <button class="btn btn-sm btn-outline-primary view-details" data-id="<%= juego.idjuego %>">
      <i class="fas fa-eye me-1"></i> Detalles
    </button>
    <div>
      <a href="/edit/<%= juego.idjuego %>" class="btn btn-sm btn-outline-secondary">
        <i class="fas fa-edit"></i>
      </a>
      <a href="#" class="btn btn-sm btn-outline-danger delete-game" data-id="<%= juego.id %>">
        <i class="fas fa-trash-alt"></i>
      </a>
    </div>
  </div>
</div>
</div>
</div>
<%= }) %>
<%= } %>
</div>
</div>

```

```

<!-- Modal de detalles mejorado -->
<div class="modal fade" id="detallesModal" tabindex="-1" aria-hidden="true">
  <div class="modal-dialog modal-lg">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="modalTitle">Detalles del juego</h5>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body" id="modalBody">
        <div class="text-center">
          <div class="spinner-border text-primary" role="status">
            <span class="visually-hidden">Cargando...</span>
          </div>
        </div>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cerrar</button>
      </div>
    </div>
  </div>
</div>

```

 Tienda de Juegos
  Lista
  Catálogo
  + Nuevo Juego


Catálogo de Juegos

Filtros


Todas las categorías

Más recientes


Mostrando 4 resultados




RPGTeléfono









ShooterMulti-plataforma



Simulación y deporte



Acción-aventura

<div> <div>  <p>RPG Teléfono</p> <p>CRASH</p> <p>\$45.99</p> <p>Editor: SEGA Clasificación: Adolescentes</p> <p>juego de carreras</p> <p>Detalles ✎ ✖</p> </div> <div>  <p>Shooter Multi-plataforma</p> <p>Call of Duty: Modern Warfare III</p> <p>\$69.99</p> <p>Editor: Activision Clasificación: Maduro +17</p> <p>Shooter en primera persona con campaña intensa y multijugador competitivo.</p> <p>Detalles ✎ ✖</p> </div> <div>  <p>Simulación y deporte Multi-plataforma</p> <p>FIFA 24</p> <p>\$69.99</p> <p>Editor: EA Sports Clasificación: Todos</p> <p>Simulador de fútbol con licencias oficiales, equipos y jugadores reales.</p> <p>Detalles ✎ ✖</p> </div> <div>  <p>Acción-aventura Nintendo Switch</p> <p>The Legend of Zelda: Breath of the Wild</p> <p>\$59.99</p> <p>Editor: Nintendo Clasificación: Todos +10</p> <p>Aventura épica en un mundo abierto donde exploras y resuelves acertijos.</p> <p>Detalles ✎ ✖</p> </div> </div>		
<div> <div>  <p>Call of Duty: Modern Warfare III</p> <p>\$69.99</p> <p>Shooter Multi-plataforma</p> <p>Editor Activision</p> <p>Clasificación Maduro +17</p> <p>Descripción Shooter en primera persona con campaña intensa y multijugador competitivo.</p> <p>Cerrar</p> </div> </div>		
 <p>SENATI</p>	[NOMBRE DEL TEMA DEL TRABAJO FINAL]	
	[APELLIDOS Y NOMBRES]	[ESCALA]

5. CONTROLAR

- **Verificar el cumplimiento de los procesos desarrollados en la propuesta de solución del caso práctico.**

EVIDENCIAS	CUMPLE	NO CUMPLE
• ¿Se identificó claramente la problemática del caso práctico?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se desarrolló las condiciones de los requerimientos solicitados?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se formularon respuestas claras y fundamentadas a todas las preguntas guía?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se elaboró un cronograma claro de actividades a ejecutar?	<input type="checkbox"/>	<input checked="" type="checkbox"/>
• ¿Se identificaron y listaron los recursos (máquinas, equipos, herramientas, materiales) necesarios para ejecutar la propuesta?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se ejecutó la propuesta de acuerdo con la planificación y cronograma establecidos?	<input type="checkbox"/>	<input checked="" type="checkbox"/>
• ¿Se describieron todas las operaciones y pasos seguidos para garantizar la correcta ejecución?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se consideran las normativas técnicas, de seguridad y medio ambiente en la propuesta de solución?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿La propuesta es pertinente con los requerimientos solicitados?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
• ¿Se evaluó la viabilidad de la propuesta para un contexto real?	<input type="checkbox"/>	<input checked="" type="checkbox"/>

6. VALORAR

- Califica el impacto que representa la propuesta de solución ante la situación planteada en el caso práctico.

CRITERIO DE EVALUACIÓN	DESCRIPCIÓN DEL CRITERIO	PUNTUACIÓN MÁXIMA	PUNTAJE CALIFICADO POR EL ESTUDIANTE
Identificación del problema	Claridad en la identificación del problema planteado.	3	2
Relevancia de la propuesta de solución	La propuesta responde adecuadamente al problema planteado y es relevante para el contexto del caso práctico.	8	9
Viabilidad técnica	La solución es técnicamente factible, tomando en cuenta los recursos y conocimientos disponibles.	6	4
Cumplimiento de Normas	La solución cumple con todas las normas técnicas de seguridad, higiene y medio ambiente.	3	5
PUNTAJE TOTAL		20	20

