EDUGESTOR

Version 1.0

Fecha: 18/01/2025

1. VISIÓN DEL PROYECTO

1.1. DESCRIPCIÓN GENERAL

EduGestor es una solución integral diseñada para automatizar y optimizar los procesos académicos y administrativos de instituciones educativas.

El sistema facilitará la gestión de estudiantes, padres, docentes, cursos, períodos académicos, asignaturas y secciones, mejorando la eficiencia operativa y la experiencia de todos los usuarios.

EduGestor está desarrollado utilizando una arquitectura MVC (Modelo-Vista-Controlador), con Java como lenguaje principal, Servlets para el controlador, JSP para las vistas, y MySQL como base de datos.

El sistema permitirá a los usuarios realizar tareas como matrículas (solo administradores), registro de calificaciones (solo docentes), consulta de notas (estudiantes y padres), y subida de contenido educativo (solo docentes), todo desde una plataforma centralizada.

1.2. OBJETIVOS PRINCIPALES

- Automatizar procesos de matrícula y gestión académica: Simplificar el proceso de matrícula de estudiantes, gestionado exclusivamente por administradores.
- Facilitar el seguimiento del rendimiento académico: Proporcionar herramientas para el registro de calificaciones (docentes) y consulta de notas (estudiantes y padres).
- Mejorar la comunicación entre stakeholders: Integrar un sistema de notificaciones para informar a los usuarios sobre cambios relevantes.
- Optimizar la gestión de recursos educativos: Gestionar eficientemente cursos, asignaturas, secciones y horarios.
- Proporcionar análisis y reportes en tiempo real: Generar reportes académicos y estadísticas para la toma de decisiones.

1.3. STAKEHOLDERS

- Estudiantes: Acceden a su información académica y consultan sus notas.
- Padres de familia/tutores: Acceden a las notas y progreso académico de sus hijos.
- Docentes: Registran calificaciones y suben contenido educativo.
- Personal administrativo: Realiza tareas de matrícula y gestión de recursos.
- Directivos académicos: Supervisan el rendimiento académico y toman decisiones basadas en datos.
- Departamento de TI: Mantiene y asegura el correcto funcionamiento del sistema.

2. ALCANCE Y OBJETIVOS

2.1. MÓDULOS DEL SISTEMA

2.1.1. Gestión de Estudiantes

- Registro y actualización de datos: Captura de información personal, académica y de contacto (solo administradores).
- Consulta de notas: Los estudiantes pueden ver sus calificaciones.
- Portal del estudiante: Acceso a horarios y notas.
- Historial académico: Visualización del progreso académico del estudiante.

2.1.2. Gestión de Padres

- Registro y vinculación: Cada estudiante debe estar vinculado a un padre o tutor.
- Consulta de notas: Los padres pueden ver las calificaciones de sus hijos.
- Portal del padre: Acceso a información académica de sus hijos.

2.1.3. Gestión de Docentes

- Registro y perfil profesional: Captura de información personal y profesional (solo administradores).
- Ingreso de calificaciones: Los docentes pueden registrar y actualizar las notas de los estudiantes.
- Subida de contenido educativo: Los docentes pueden subir material de apoyo para los cursos.
- Portal del docente: Acceso a cursos asignados y estudiantes.

2.1.4. Gestión Académica

- Planificación académica: Creación y gestión de períodos académicos (solo administradores).
- Gestión de cursos: Creación, actualización y eliminación de cursos (solo administradores).
- Control de evaluaciones: Registro de calificaciones (solo docentes).
- Gestión de horarios: Creación y asignación de horarios de clases (solo administradores).

2.1.5. Gestión Administrativa

- Gestión de matrículas: Proceso de matrícula en cursos y secciones (solo administradores).
- Control de documentación: Almacenamiento y gestión de documentos académicos.
- Gestión de recursos: Asignación de aulas y recursos educativos.
- Reportes y estadísticas: Generación de reportes académicos y administrativos.
- Configuración del sistema: Personalización de parámetros del sistema.

3. ARQUITECTURA DEL SISTEMA

3.1. STACK TECNOLÓGICO

Frontend

- JSP (JavaServer Pages): Para la construcción de interfaces de usuario dinámicas.
- HTML/CSS/JavaScript: Para el diseño y la interactividad de las páginas.
- Bootstrap: Para componentes de diseño responsivos y consistentes.

Backend

- Java: Como lenguaje principal del sistema.
- Servlets: Para la lógica del controlador en la arquitectura MVC.
- JDBC: Para la conexión y manipulación de la base de datos MySQL.

Base de Datos

- MySQL: Para el almacenamiento de datos relacionales.
- MySQL Workbench: Para el diseño, modelado y administración de la base de datos.

• Backups automatizados: Para garantizar la disponibilidad de los datos.

Infraestructura

- Apache Tomcat: Como servidor de aplicaciones para desplegar el sistema.
- Git: Para el control de versiones del código.
- Maven: Para la gestión de dependencias y construcción del proyecto.

3.2. REQUISITOS NO FUNCIONALES

Disponibilidad

• Uptime: 99.9%.

• Tiempo máximo de inactividad: 8.76 horas/año.

Rendimiento

• Tiempo de respuesta: < 2 segundos.

• Throughput: 1000 requests/segundo.

• Latencia máxima: 200ms.

Seguridad

- Autenticación basada en sesiones: Para la gestión de usuarios.
- Encriptación de datos sensibles: Para proteger información confidencial.
- Auditoría de acciones: Para rastrear cambios en el sistema.
- Backups cifrados: Para garantizar la seguridad de los datos.

Escalabilidad

- Horizontal: Para componentes stateless (por ejemplo, servlets).
- Vertical: Para la base de datos.
- Optimización de consultas: Para mejorar el rendimiento de la base de datos.

4. HISTORIAS DE USUARIO

4.1. EPIC 1: Gestión de Usuarios

HU-001: Registro de Estudiantes

Descripción: Como administrador, quiero poder registrar nuevos estudiantes en el sistema para que puedan acceder a sus notas.

Criterios de Aceptación:

- 1. El formulario debe incluir datos personales, fecha de nacimiento, dirección y contacto.
- 2. Validar formato de email y teléfono.
- 3. Verificar que el DNI no esté duplicado.
- 4. Crear automáticamente un usuario en la base de datos.
- 5. Enviar email de confirmación.

Estimación: 5 puntos.

Prioridad: Alta.

HU-002: Registro de Padres

Descripción: Como administrador, quiero poder registrar padres o tutores y vincularlos a sus hijos.

Criterios de Aceptación:

- 1. El formulario debe incluir datos personales y de contacto.
- 2. Validar formato de email y teléfono.
- 3. Vincular al padre con el estudiante correspondiente.
- 4. Crear automáticamente un usuario en la base de datos.
- 5. Enviar email de confirmación.

Estimación: 5 puntos.

Prioridad: Alta.

HU-003: Registro de Docentes

Descripción: Como administrador, quiero poder registrar nuevos docentes en el sistema para que puedan gestionar cursos y calificaciones.

Criterios de Aceptación:

- 1. El formulario debe incluir datos personales, información profesional y áreas de especialización.
- 2. Permitir subir CV en formato PDF.
- 3. Validar títulos académicos.
- 4. Crear automáticamente un usuario en la base de datos.
- 5. Configurar permisos iniciales.

Estimación: 5 puntos.

Prioridad: Alta.

4.2. EPIC 2: Gestión de Matrículas

HU-004: Proceso de Matrícula

Descripción: Como administrador, quiero poder matricular estudiantes en cursos y secciones para organizar el período académico.

Criterios de Aceptación:

- 1. Mostrar cursos disponibles, horarios y cupos.
- 2. Validar conflictos de horario.
- 3. Verificar prerrequisitos.
- 4. Generar comprobante de matrícula.

Estimación: 8 puntos.

Prioridad: Alta.

4.3. EPIC 3: Gestión Académica

HU-006: Registro de Notas

Descripción: Como docente, quiero poder registrar y gestionar calificaciones para evaluar el rendimiento de los estudiantes.

Criterios de Aceptación:

- 1. Permitir ingreso de notas por evaluación.
- 2. Cálculo automático de promedios.

- 3. Mantener histórico de cambios.
- 4. Bloquear modificaciones fuera de plazo.

Estimación: 8 puntos.

Prioridad: Alta.

HU-007: Subida de Contenido Educativo

Descripción: Como docente, quiero poder subir contenido educativo para apoyar el aprendizaje de los estudiantes.

Criterios de Aceptación:

- 1. Permitir subida de archivos (PDF, videos, presentaciones).
- 2. Organizar contenido por curso y sección.
- 3. Notificar a los estudiantes sobre nuevo contenido.

Estimación: 5 puntos.

Prioridad: Media.

4.4. EPIC 4: Reportes y Análisis

HU-009: Reportes Académicos

Descripción: Como administrador, quiero generar reportes de rendimiento para analizar el desempeño académico.

Criterios de Aceptación:

- 1. Generar reportes de rendimiento por curso.
- 2. Filtros personalizables.
- 3. Exportación a múltiples formatos.
- 4. Gráficos interactivos.

Estimación: 8 puntos.

Prioridad: Media.

4.5. EPIC 5: Comunicación

HU-011: Sistema de Notificaciones

Descripción: Como usuario del sistema, quiero recibir notificaciones relevantes para estar informado de eventos importantes.

Criterios de Aceptación:

- 1. Notificar sobre cambios de horario, nuevas calificaciones y contenido educativo.
- 2. Configuración de preferencias.
- 3. Múltiples canales (email, push).

Estimación: 5 puntos.

Prioridad: Baja.

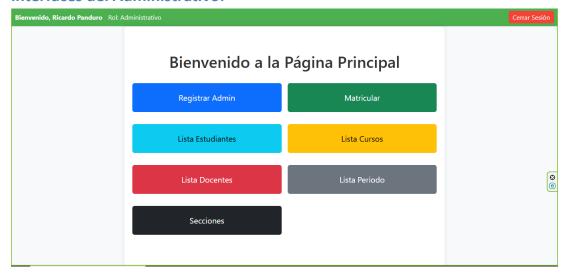
5. ANEXOS

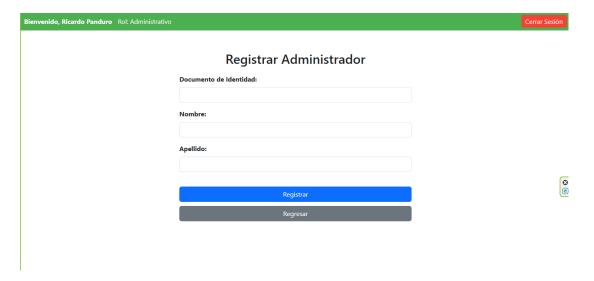
5.1. DIAGRAMAS

- Arquitectura del sistema: Diagrama de componentes y flujo de datos.
- Modelo de datos: Diagrama ER de la base de datos.

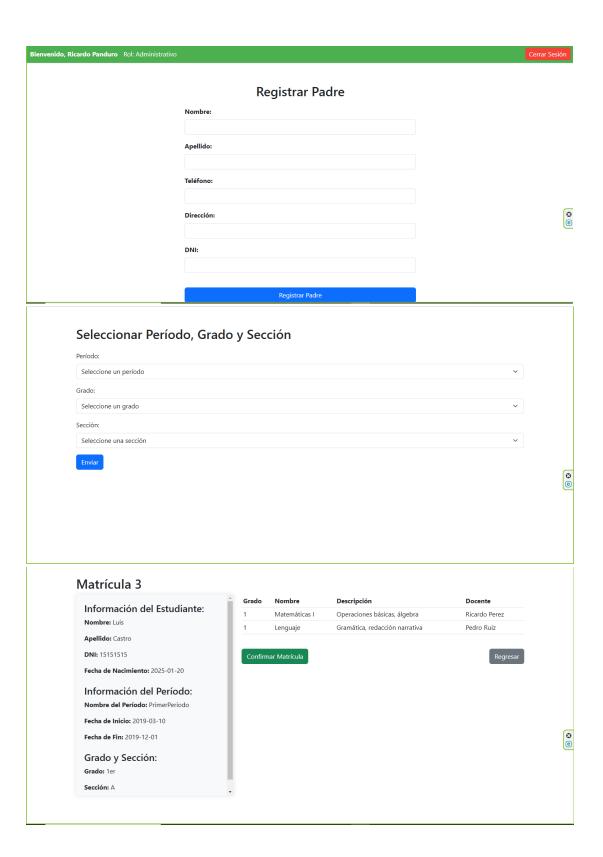
5.2. MOCKUPS

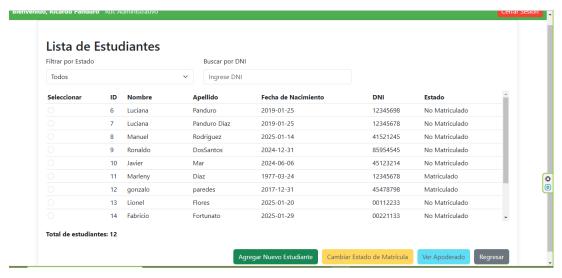
Interfaces del Administrativo:



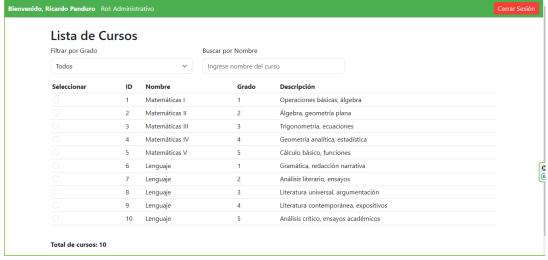


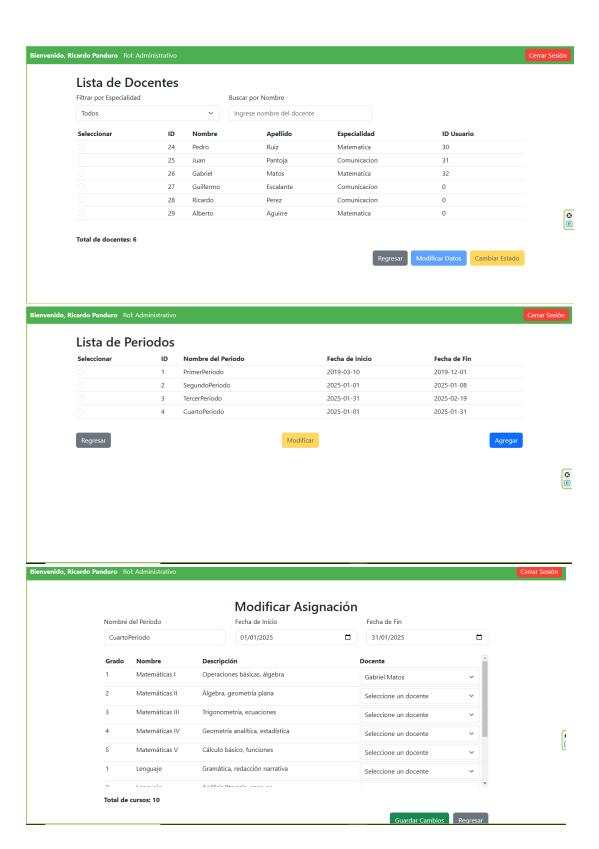












Estudiantes Matriculados

