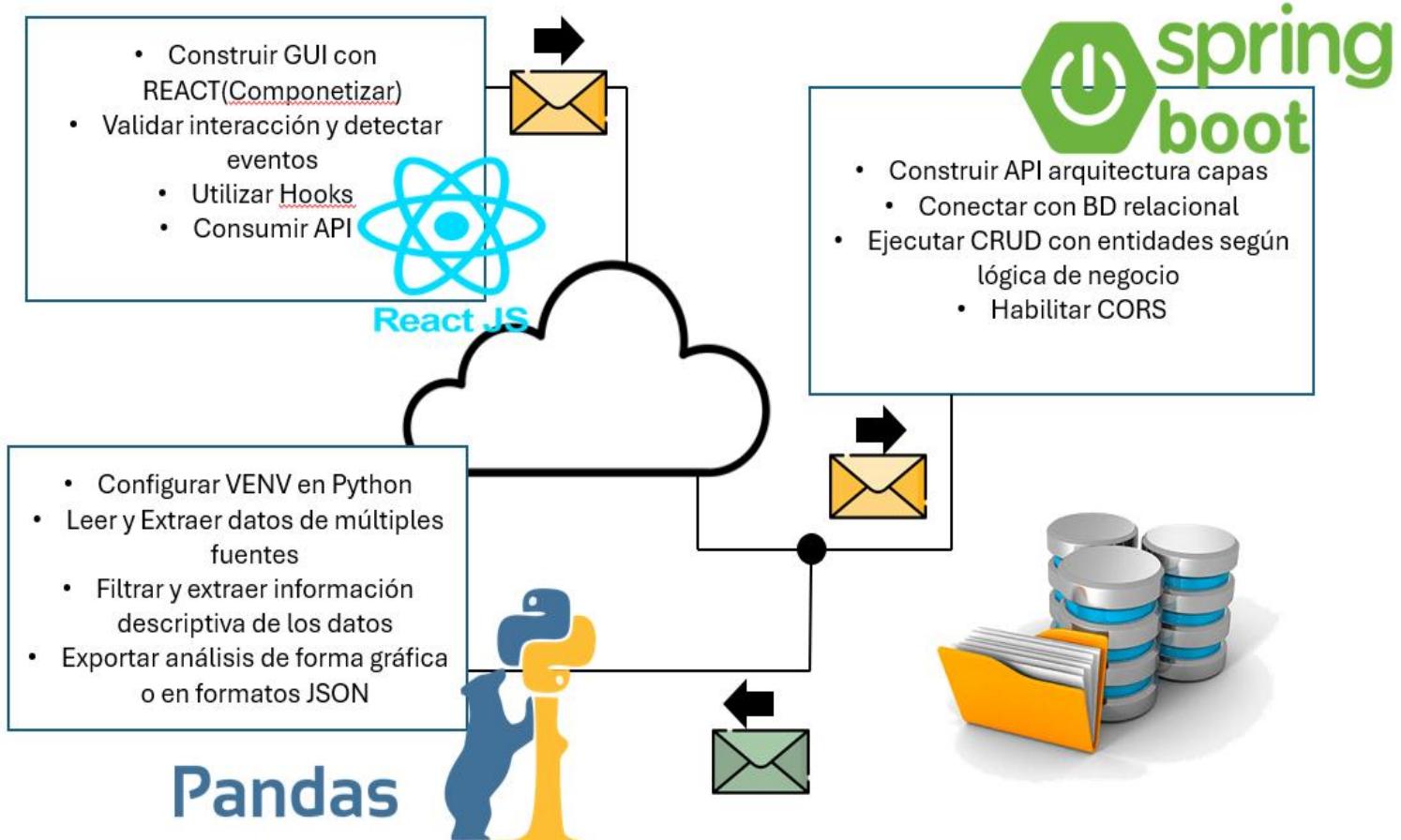


Plataforma Integral de Gestión y Acompañamiento Estudiantil



Objetivo General del Proyecto

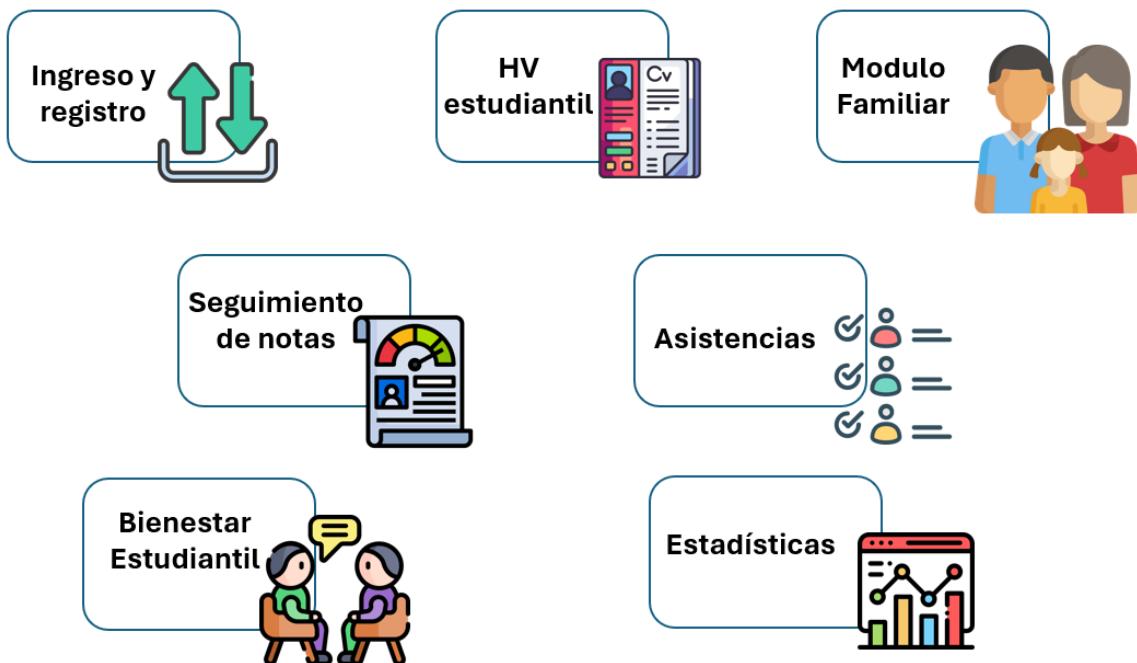
Diseñar y desarrollar una plataforma web integral que permita la gestión académica, social y analítica de los estudiantes de una institución educativa, integrando conocimientos de desarrollo backend con Spring Boot, frontend con React y análisis de datos con Python, mediante un trabajo colaborativo en células de desarrollo que simulen un entorno real de producción.

(604) 480 88 22



Cesde.edu.co

Calle 49 41-9
Colombia - Antioquia - Medellín

Módulos funcionales del sistema**1. Módulo de Ingreso y Registro**

Este módulo es fundamental para el acceso a la plataforma, permitiendo que los diferentes tipos de usuarios (estudiantes, docentes, familiares, personal de bienestar y administradores) se registren e inicien sesión de manera segura. A través de este módulo, los usuarios podrán crear una cuenta, ingresar a la plataforma y acceder a las funcionalidades correspondientes a su rol. Además, se realizarán validaciones de campos para asegurar que los datos ingresados sean correctos y completos, y se implementará control de acceso por rol, de forma que cada usuario solo tenga acceso a la información y funcionalidades que le correspondan según su perfil.

2. Módulo de Hoja de Vida Estudiantil

En este módulo, los estudiantes tendrán un perfil público que les permitirá mostrar su información académica, habilidades, intereses y certificados. Al estilo de plataformas como LinkedIn, los estudiantes podrán cargar evidencias y enlaces que respalden su formación,

(604) 480 88 22
Cesde.edu.co



Calle 49 41-9
Colombia - Antioquia - Medellín

como proyectos, investigaciones, o cualquier otra actividad que complemente su formación académica. Este módulo facilitará que tanto los estudiantes como los coordinadores, familiares y docentes puedan ver el progreso académico y profesional de cada estudiante, además de promover una mayor visibilidad de sus logros.

3. Módulo Familiar

Este módulo está diseñado para familiarizar a los familiares con la evolución del estudiante. Los familiares podrán visualizar el perfil académico del estudiante asociado, con acceso a información sobre su rendimiento académico, asistencia, y posibles alertas (como bajo rendimiento o ausencias frecuentes). Este módulo también les permitirá estar al tanto del progreso del estudiante en tiempo real, y ofrecerles una visión clara de las áreas en las que pueden apoyar a su familiar.

4. Módulo Académico y Seguimiento de Notas

Este módulo es crucial para el seguimiento académico de los estudiantes. A través de este módulo, los estudiantes podrán consultar sus calificaciones por materia y tener acceso a un histórico de su rendimiento académico. Además, se generarán recomendaciones automáticas de apoyo académico cuando un estudiante tenga bajo rendimiento, con el objetivo de que pueda recibir ayuda a tiempo. Este módulo también permitirá que docentes y coordinadores identifiquen áreas de mejora y tomen medidas preventivas para apoyar a los estudiantes que lo necesiten.

5. Módulo de Asistencia

Este módulo permite el registro de asistencia tanto por grupo como por materia, facilitando el control de la presencia de los estudiantes en sus clases. Los estudiantes podrán ser registrados como presentes, ausentes o justificados, y se generarán reportes tanto individuales como grupales para que los docentes y coordinadores puedan hacer seguimiento. Este módulo es clave para identificar patrones de asistencia y tomar decisiones respecto a posibles intervenciones si se detectan faltas recurrentes.

(604) 480 88 22
Cesde.edu.co



Calle 49 41-9
Colombia - Antioquia - Medellín

6. Módulo de Apoyo y Bienestar Estudiantil

En este módulo, los estudiantes podrán reportar sus necesidades (emocionales, académicas, económicas) a través de un formulario fácil de llenar. Estas solicitudes serán gestionadas por el personal adecuado (psicología, bienestar, o dirección), quienes podrán hacer un seguimiento a cada caso y ofrecer el apoyo necesario. Este módulo es esencial para garantizar que los estudiantes tengan acceso a los recursos de apoyo que requieren para su bienestar personal y académico, y que puedan recibir la ayuda necesaria en momentos de dificultad.

7. Módulo Estadístico y Analítico

Este módulo proporciona información clave sobre el rendimiento y el bienestar de los estudiantes a través de dashboards interactivos. Los coordinadores y docentes podrán acceder a análisis estadísticos sobre:

Asistencias: Muestra los niveles de asistencia tanto por grupo como por estudiante.

Distribución de notas: Permite ver las calificaciones por materia o grupo, ayudando a identificar tendencias y áreas de mejora.

Estudiantes en riesgo académico o emocional: A través de análisis de rendimiento y asistencia, el sistema puede identificar estudiantes con riesgo de bajo rendimiento o posibles problemas emocionales, facilitando la intervención temprana.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

Backlog por submódulo:**Backend 2****Módulo de ingreso y registro**

HU01 – Crear entidad Usuario

Objetivo: Diseñar el modelo Usuario para representar datos básicos del sistema.

Criterios de aceptación:

- El modelo debe tener campos como id, nombre, correo, contraseña, rol, estado.
- Debe tener anotaciones JPA (@Entity, @Id, @GeneratedValue, etc.).
- Debe incluir validaciones como @NotNull, @Email y @Size.

HU02 – Crear entidad Estudiante

Objetivo: Diseñar el modelo Estudiante con relación a Usuario.

Criterios de aceptación:

- Debe incluir campos específicos como programa, semestre, fechaNacimiento.
- Relacionar con Usuario usando @OneToOne o @ManyToOne.
- Las relaciones deben tener cascade y fetch bien definidos.

HU03 – Crear entidad Familiar

Objetivo: Implementar el modelo Familiar relacionado a Usuario.

Criterios de aceptación:

- Tener campos como parentesco, telefono, direccion.
- Relación con entidad Usuario.
- Validaciones de campos requeridos.

HU04 – Crear entidad Docente

Objetivo: Implementar el modelo Docente relacionado a Usuario.

Criterios de aceptación:

- Tener campos como especialidad, nivelAcademico, departamento.
- Relación con entidad Usuario.
- Validaciones JPA.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU021 – Crear entidad Empresario

Objetivo: Implementar el modelo Empresario relacionado a Usuario.

Criterios de aceptación:

- Tener campos como nombre, Sector, departamento.
- Relación con entidad Usuario.
- Validaciones JPA.

HU05 – Crear repositorio Usuario

Objetivo: Crear interfaz UsuarioRepository para consultas básicas.

Criterios de aceptación:

- Extender JpaRepository<Usuario, Long>.
- Tener método findByCorreo(String correo).
- Permitir uso en pruebas unitarias de servicios.

HU06 – Crear repositorio Estudiante

Objetivo: Crear interfaz EstudianteRepository.

Criterios de aceptación:

- Extender JpaRepository<Estudiante, Long>.
- Incluir método de búsqueda por nombre o código.
- Permitir uso en servicios para controladores.

HU07 – Crear DTO de Usuario

Objetivo: Diseñar clase UsuarioDTO para encapsular datos.

Criterios de aceptación:

- Tener solo los campos públicos (sin contraseña).
- Debe usarse en el servicio y controlador.
- Validaciones con @NotNull, @Email, etc.

HU08 – Crear servicio de Usuario

Objetivo: Implementar clase UsuarioService con operaciones CRUD.

Criterios de aceptación:

- Tener métodos: crearUsuario, listarUsuarios, obtenerPorId, actualizar, eliminar.
- Usar UsuarioDTO como entrada/salida.
- Manejar excepciones básicas (UsuarioNoEncontradoException).

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU09 – Crear controlador Usuario

Objetivo: Implementar UsuarioController con endpoints REST.

Criterios de aceptación:

- Endpoint POST /usuarios para crear usuario con DTO.
- Endpoint GET /usuarios para listar todos.
- Endpoint GET /usuarios/{id} para obtener por ID.

HU10 – Validar datos al registrar usuario

Objetivo: Garantizar que el usuario se registre con datos válidos.

Criterios de aceptación:

- Validaciones anotadas en DTO (@NotBlank, @Email, etc.).
- Si hay errores, debe devolver código 400 con mensaje.
- La contraseña debe tener mínimo 6 caracteres.

HU11 – Crear servicio Estudiante

Objetivo: Implementar EstudianteService con operaciones CRUD.

Criterios de aceptación:

- Métodos para crear, listar, actualizar, eliminar estudiante.
- Relación correcta con entidad Usuario.
- Uso de EstudianteDTO.

HU12 – Crear DTO Estudiante

Objetivo: Definir DTO para transferir datos del estudiante.

Criterios de aceptación:

- No debe incluir campos irrelevantes.
- Validaciones con @NotNull, @Size, etc.
- Ser usado en EstudianteController.

HU13 – Crear controlador Estudiante

Objetivo: Crear endpoints REST para gestionar estudiantes.

Criterios de aceptación:

- Endpoint POST /estudiantes para crear.
- Endpoint GET /estudiantes y /estudiantes/{id}.
- Endpoint PUT /estudiantes/{id} y DELETE.

HU14 – Crear lógica de actualización de Usuario

Objetivo: Permitir actualizar los datos de un usuario por ID.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

Criterios de aceptación:

- El método debe validar existencia antes de actualizar.
- El campo correo no debe repetirse.
- Devolver DTO actualizado al cliente.

HU15 – Validar existencia antes de eliminar Usuario

Objetivo: Manejar la eliminación segura del usuario.

Criterios de aceptación:

- Eliminar solo si el usuario existe.
- Devolver código 404 si no se encuentra.
- Confirmar con un mensaje de éxito.

HU16 – Gestionar errores en controladores

Objetivo: Manejar errores de entrada/salida de forma controlada.

Criterios de aceptación:

- Si no se encuentra un usuario, retornar 404.
- Si hay errores de validación, retornar 400.
- Usar clases de excepción personalizadas.

HU17 – Crear servicio Familiar y su controlador

Objetivo: Desarrollar FamiliarService y FamiliarController.

Criterios de aceptación:

- CRUD completo basado en FamiliarDTO.
- Relaciones funcionales con entidad Usuario.
- Validaciones al crear y actualizar.

HU18 – Crear controlador Docente

Objetivo: Implementar endpoints para Docente.

Criterios de aceptación:

- CRUD completo para docentes.
- Asociación correcta con Usuario.
- Validaciones de DTO.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU19 – Listar todos los usuarios por rol

Objetivo: Crear endpoint que filtre usuarios por su rol.

Criterios de aceptación:

- Endpoint GET /usuarios/rol/{rol}.
- El rol debe coincidir exactamente con los guardados.
- Devolver lista de DTOs.

HU20 – Probar servicios con datos simulados

Objetivo: Crear pruebas básicas para servicios UsuarioService y EstudianteService.

Criterios de aceptación:

- Simular creación y consulta de usuarios.
- Verificar que los datos se guardan correctamente.
- Usar @DataJpaTest o repositorios embebidos.

Módulo de hoja de vida estudiantil**HU01 – Crear entidad PerfilEstudiante**

Objetivo: Diseñar el modelo de hoja de vida con campos clave del perfil.

Criterios de aceptación:

- El modelo debe incluir: id, resumen, intereses, experiencia, proyectos, estudiante_id.
- Relacionar con la entidad Estudiante mediante @OneToOne.
- Debe estar anotada con @Entity.

HU02 – Crear entidad Habilidad

Objetivo: Representar habilidades técnicas y blandas del estudiante.

Criterios de aceptación:

- El modelo debe tener campos como nombre, nivel, tipo (blanda/técnica).
- Relación con PerfilEstudiante (@ManyToOne).
- Validaciones con @NotNull, @Size.

HU03 – Crear entidad Proyecto

Objetivo: Modelar los proyectos personales o académicos del estudiante.

Criterios de aceptación:

- Campos: titulo, descripcion, url, tecnologias. (604) 480 88 22
- Relación con PerfilEstudiante. Cesde.edu.co
- Validaciones con JPA. Calle 49 41-9

Colombia - Antioquia - Medellín



HU04 – Crear entidad Certificado

Objetivo: Almacenar certificados adjuntos por el estudiante.

Criterios de aceptación:

- Incluir campos como nombre, institucion, fecha, urlArchivo.
- Estar relacionada con PerfilEstudiante.
- Validaciones obligatorias.

HU05 – Crear repositorio PerfilEstudianteRepository

Objetivo: Permitir operaciones de acceso a los perfiles.

Criterios de aceptación:

- Extender JpaRepository<PerfilEstudiante, Long>.
- Método findByEstudiantId(Long id).
- Soportar operaciones de CRUD en servicios.

HU06 – Crear DTO PerfilEstudianteDTO

Objetivo: Transferir los datos del perfil del estudiante.

Criterios de aceptación:

- Incluir campos: resumen, intereses, proyectos, habilidades.
- Excluir relaciones innecesarias o información sensible.
- Validar campos con @NotBlank, @Size.

HU07 – Crear servicio PerfilEstudianteService

Objetivo: Implementar lógica de negocio para CRUD del perfil.

Criterios de aceptación:

- Métodos: crear, actualizar, buscarPorEstudiante, eliminar.
- Usar PerfilEstudianteDTO.
- Lanzar excepción personalizada si no se encuentra el perfil.

HU08 – Crear controlador PerfilEstudianteController

Objetivo: Implementar endpoints REST para gestionar perfiles.

Criterios de aceptación:

- POST /perfiles para crear un perfil.
- GET /perfiles/{id} para obtener perfil por ID.
- PUT /perfiles/{id} y DELETE.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU09 – Crear servicio y controlador de Proyecto

Objetivo: Permitir gestión de proyectos del estudiante.

Criterios de aceptación:

- CRUD completo con DTO.
- Asociar proyectos al perfil correcto.
- Validar campos requeridos.

HU10 – Crear servicio y controlador de Habilidad

Objetivo: Permitir crear, listar y eliminar habilidades del estudiante.

Criterios de aceptación:

- Operaciones POST, GET, DELETE.
- Asociar correctamente con PerfilEstudiante.
- Validaciones con DTO.

HU11 – Crear servicio y controlador de Certificado

Objetivo: Permitir el registro y visualización de certificados.

Criterios de aceptación:

- Subir datos básicos del certificado.
- Asociar con perfil correspondiente.
- Permitir listar certificados por estudiante.

HU12 – Permitir búsqueda de perfiles por palabra clave

Objetivo: Implementar filtro de perfiles por intereses o resumen.

Criterios de aceptación:

- Endpoint GET /perfiles/busqueda?query=texto.
- Búsqueda parcial con LIKE.
- Respuesta con DTOs.

HU13 – Validar que un estudiante no tenga más de un perfil

Objetivo: Prevenir duplicados al crear perfil.

Criterios de aceptación:

- Antes de crear, verificar existencia con estudiante_id.
- Retornar mensaje de error si ya existe.
- Devolver 409 si hay conflicto.

HU14 – Asociar automáticamente el perfil al estudiante autenticado

Objetivo: Relacionar el perfil con el estudiante sin indicar el ID.

Criterios de aceptación:

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- El endpoint debe tomar el ID del estudiante desde el servicio.
- El DTO no debe pedir estudiante_id.
- Funciona en contexto de sesión simulada (sin JWT por ahora).

HU15 – Listar todos los perfiles existentes

Objetivo: Crear endpoint para consultar todos los perfiles creados.

Criterios de aceptación:

- GET /perfiles retorna lista de PerfilEstudianteDTO.
- Debe paginar resultados con Pageable.
- Filtrado por programa o semestre opcional.

HU16 – Subir enlaces o archivos de proyectos

Objetivo: Permitir adjuntar URL o evidencia de proyecto.

Criterios de aceptación:

- DTO debe aceptar urlProyecto.
- Validar formato de URL.
- Endpoint debe guardar y retornar el enlace.

HU17 – Editar un proyecto existente

Objetivo: Permitir modificar la información de un proyecto ya creado.

Criterios de aceptación:

- PUT /proyectos/{id} actualiza los campos.
- Validar que el proyecto pertenezca al estudiante.
- Devolver DTO actualizado.

HU18 – Eliminar habilidad por ID

Objetivo: Eliminar una habilidad específica del perfil.

Criterios de aceptación:

- DELETE /habilidades/{id} elimina si existe.
- Validar propiedad del estudiante.
- Devolver 200 o 204 con confirmación.

HU19 – Obtener perfil completo por estudiante

Objetivo: Mostrar perfil con todos sus elementos (proyectos, habilidades, certificados).

Criterios de aceptación:

- Endpoint GET /perfil-estudiante/{id} con datos anidados.
- Debe retornar lista de objetos relacionados.
- Excluir campos sensibles como contraseñas o IDs internos.

(604) 480 88 22

Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín



HU20 – Exportar perfil a PDF (como extra para integración futura)

Objetivo: Dejar preparado servicio para exportar hoja de vida.

Criterios de aceptación:

- Servicio genera resumen en texto plano (versión simulada).
- Retorna JSON como borrador de hoja de vida.
- Posterior integración con librería de PDF.

Modulo Familiar

HU02 – Crear entidad VinculoFamiliarEstudiante

Objetivo: Representar la relación entre familiar y estudiante.

Criterios de aceptación:

- Campos: id, familiar_id, estudiante_id, autorizado (boolean).
- Relación con Familiar y Estudiante.
- Validación para permitir múltiples vínculos (por ejemplo, madre y padre).

HU033 – Crear entidad AccesoFamiliar

Objetivo: Registrar los accesos del familiar a la plataforma para análisis estadístico y trazabilidad.

Criterios de aceptación:

- Campos: id, familiar_id, fecha_acceso, accion_realizada.
- Relación con la entidad Familiar.
- Validaciones: no permitir fecha_acceso nula ni accion_realizada vacía.

Permitir múltiples registros por familiar.

HU044 – Crear entidad NotificacionFamiliar

Objetivo: Almacenar las notificaciones enviadas a los familiares sobre el desempeño y alertas del estudiante.

Criterios de aceptación:

- Campos: id, familiar_id, titulo, mensaje, fecha_envio, estado (leída/no leída).
- Relación con la entidad Familiar.
- Validaciones: campos obligatorios, máximo de caracteres en mensaje.
- Permitir recuperar historial de notificaciones por familiar.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU055 – Crear entidad ComentarioFamiliar

Objetivo: Permitir que los familiares registren comentarios sobre el proceso académico del estudiante.

Criterios de aceptación:

- Campos: id, familiar_id, estudiante_id, comentario, fecha.
- Relación con Familiar y Estudiante.
- Validaciones: comentario obligatorio, longitud máxima de 500 caracteres.
- Registrar fecha automática y permitir edición posterior.

HU03 – Crear repositorio FamiliarRepository

Objetivo: Permitir operaciones CRUD sobre familiares.

Criterios de aceptación:

- Extender JpaRepository<Familiar, Long>.
- Método findByCorreo(String correo).
- Método para listar familiares por estudiante.

HU04 – Crear DTO FamiliarDTO

Objetivo: Transferir datos del familiar hacia el frontend.

Criterios de aceptación:

- Incluir nombre, correo, parentesco, teléfono.
- No debe incluir el ID del usuario directamente.
- Validaciones con @NotBlank, @Email, etc.

HU05 – Crear servicio FamiliarService

Objetivo: Implementar lógica de negocio para familiares.

Criterios de aceptación:

- Métodos para registrar, actualizar, buscar, eliminar familiares.
- Uso de FamiliarDTO.
- Validación de existencia antes de operaciones.

HU06 – Crear controlador FamiliarController

Objetivo: Exponer endpoints para CRUD de familiares.

Criterios de aceptación:

- POST /familiares, GET /familiares, PUT /familiares/{id}, DELETE /familiares/{id}.
- Devolver respuestas con estado HTTP correcto.

(604) 480 88 22

Cesde.edu.co



Calle 49 41-9
Colombia - Antioquia - Medellín

- Manejo de errores si el familiar no existe.

HU07 – Crear DTO VinculoFamiliarEstudianteDTO

Objetivo: Transferir datos de relación entre familiar y estudiante.

Criterios de aceptación:

- Incluir campos: idFamiliar, idEstudiante, autorizado.
- Usado para registrar vínculos desde el frontend.
- Validaciones con @NotNull.

HU08 – Crear servicio y controlador para VinculoFamiliarEstudiante

Objetivo: Permitir registrar la relación entre familiar y estudiante.

Criterios de aceptación:

- Endpoint POST /vinculos para crear.
- GET /vinculos/estudiante/{id} para ver vínculos por estudiante.
- Validar existencia de familiar y estudiante antes de vincular.

HU09 – Consultar hoja de vida del estudiante desde familiar

Objetivo: Permitir al familiar consultar el perfil del estudiante vinculado.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/perfil-estudiante.
- Validar que haya un vínculo autorizado.
- Retornar datos desde PerfilEstudianteDTO.

HU10 – Consultar historial de notas del estudiante

Objetivo: Permitir al familiar ver las notas de su estudiante vinculado.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/notas.
- Retornar lista de materias y calificaciones.
- Validar autorización del vínculo.

HU11 – Consultar asistencia del estudiante desde familiar

Objetivo: Mostrar al familiar los reportes de asistencia.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/asistencia.
- Filtrar por fecha, grupo o materia.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Usar DTO simplificado de asistencia.

HU12 – Consultar alertas académicas del estudiante

Objetivo: Notificar al familiar si el estudiante tiene bajo rendimiento.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/alertas.
- Mostrar materias con notas críticas.
- Incluir recomendaciones o comentarios.

HU13 – Consultar solicitudes a bienestar realizadas por el estudiante

Objetivo: Dar visibilidad al familiar de solicitudes de apoyo realizadas.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/solicitudes-bienestar.
- Solo mostrar si el estudiante autoriza.
- Visualización con tipo de solicitud y estado.

HU14 – Registrar familiar desde el frontend

Objetivo: Permitir que un familiar se registre de forma independiente.

Criterios de aceptación:

- Endpoint POST /familiares/registro.
- Validaciones completas de entrada.
- Guardar automáticamente en la tabla de usuarios con rol "FAMILIAR".

HU15 – Ver todos los estudiantes asociados a un familiar

Objetivo: Mostrar todos los estudiantes vinculados a un familiar.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/estudiantes.
- Devolver lista con nombre, programa y semestre.
- Incluir campo autorizado.

HU16 – Permitir eliminar vínculo entre familiar y estudiante

Objetivo: Desvincular a un familiar de un estudiante.

Criterios de aceptación:

- Endpoint DELETE /vinculos/{id}.
- Verificación previa de existencia.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Confirmación con mensaje de éxito.

HU17 – Editar datos del familiar

Objetivo: Actualizar la información personal del familiar.

Criterios de aceptación:

- Endpoint PUT /familiares/{id}.
- Validar formato de email y teléfono.
- Devolver DTO actualizado.

HU18 – Verificar si familiar ya está registrado

Objetivo: Permitir validación por correo para evitar duplicados.

Criterios de aceptación:

- Endpoint GET /familiares/correo/{email}.
- Retornar si ya existe o no.
- Incluir mensaje claro al usuario.

HU19 – Permitir ver resumen académico general del estudiante

Objetivo: Mostrar vista rápida del desempeño del estudiante.

Criterios de aceptación:

- Endpoint GET /familiares/{id}/resumen-academico.
- Incluir nota promedio, asistencia y alertas.
- Usar DTO resumido.

HU20 – Listar todos los vínculos registrados

Objetivo: Obtener todos los vínculos familiares del sistema.

Criterios de aceptación:

- Endpoint GET /vinculos.
- Devolver lista con nombres del familiar y estudiante.
- Incluir campo autorizado.

Modulo Académico

HU01 – Crear entidad Nota

Objetivo: Modelar las calificaciones de los estudiantes por materia.

Criterios de aceptación:

- Campos: id, valor, tipoEvaluacion, fecha, estudiante_id, materia_id.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Validar que el valor esté entre 0.0 y 5.0.
- Relacionar correctamente con Estudiante y Materia.

HU02 – Crear entidad Materia

Objetivo: Representar las materias que puede cursar un estudiante.

Criterios de aceptación:

- Campos: id, nombre, codigo, docente_id.
- Relación con Docente.
- Validaciones de campos obligatorios.

HU03 – Crear entidad Grupo

Objetivo: Representar los grupos académicos para cada materia.

Criterios de aceptación:

- Campos: id, nombre, materia_id, semestre.
- Relación con Materia.
- Validaciones de estructura.

HU04 – Crear entidad Matricula

Objetivo: Registrar la matrícula de estudiantes en materias.

Criterios de aceptación:

- Campos: id, estudiante_id, materia_id, grupo_id, periodo.
- Evitar duplicados para un mismo periodo.
- Relaciones correctas entre entidades.

HU05 – Crear repositorio NotaRepository

Objetivo: Permitir acceso a las calificaciones registradas.

Criterios de aceptación:

- Extender JpaRepository<Nota, Long>.
- Métodos: findByEstudianteId, findByMaterialIdAndGrupoid.
- Permitir ordenamiento por fecha.

HU06 – Crear DTO NotaDTO

Objetivo: Transferir datos de calificaciones al frontend.

Criterios de aceptación:

- Campos: materia, valor, fecha, tipoEvaluacion.
- Validaciones con @Min, @Max, @NotNull.
- No incluir información innecesaria.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU07 – Crear servicio NotaService

Objetivo: Implementar lógica CRUD para notas.

Criterios de aceptación:

- Métodos para crear, listar, actualizar, eliminar.
- Validar existencia de estudiante y materia antes de registrar.
- Usar NotaDTO.

HU08 – Crear controlador NotaController

Objetivo: Exponer endpoints REST para gestión de notas.

Criterios de aceptación:

- POST /notas para registrar nota.
- GET /notas/estudiante/{id} para listar por estudiante.
- PUT /notas/{id}, DELETE /notas/{id} para editar y eliminar.

HU09 – Calcular promedio general del estudiante

Objetivo: Obtener el promedio de todas las notas activas de un estudiante.

Criterios de aceptación:

- Servicio que calcule y devuelva valor promedio.
- Endpoint GET /notas/promedio/estudiante/{id}.
- Incluir en la respuesta número de materias.

HU10 – Obtener notas por materia y grupo

Objetivo: Consultar notas filtradas por materia y grupo.

Criterios de aceptación:

- Endpoint GET /notas/materia/{materialId}/grupo/{grupoid}.
- Solo accesible para docentes o admins.
- Retorna lista de estudiantes con notas.

HU11 – Detectar rendimiento bajo

Objetivo: Identificar estudiantes con notas por debajo de 3.0.

Criterios de aceptación:

- Servicio que filtre notas < 3.0.
- Endpoint GET /notas/bajo-rendimiento.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Retornar recomendaciones básicas en DTO.

HU12 – Registrar múltiples notas por estudiante y materia

Objetivo: Permitir diferentes tipos de evaluación por materia.

Criterios de aceptación:

- Registrar varias notas con campo tipoEvaluacion (quiz, parcial, proyecto).
- Validar límite de repeticiones por tipo si se desea.
- Consolidar notas por tipo.

HU13 – Agregar comentarios a las notas

Objetivo: Permitir incluir observaciones del docente.

Criterios de aceptación:

- Campo comentario opcional en Nota.
- DTO debe reflejarlo.
- Visible solo para usuarios autorizados.

HU14 – Consultar resumen académico del estudiante

Objetivo: Mostrar materias cursadas, promedio y notas.

Criterios de aceptación:

- Endpoint GET /estudiantes/{id}/resumen-academico.
- Incluir promedio general, listado de materias y últimas notas.
- Usar DTO especializado.

HU15 – Visualizar notas desde módulo familiar

Objetivo: Permitir que un familiar consulte el desempeño académico.

Criterios de aceptación:

- Reutilizar servicio de consulta por estudiante vinculado.
- Validar autorización familiar.
- Endpoint exclusivo para perfil "FAMILIAR".

HU16 – Validar rango de nota antes de registrar

Objetivo: Prevenir errores al ingresar calificaciones.

Criterios de aceptación:

- La nota debe estar entre 0.0 y 5.0.
- Devolver 400 si el valor no es válido.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Mostrar mensaje claro.

HU17 – Consultar ranking de estudiantes por grupo

Objetivo: Listar estudiantes ordenados por promedio en un grupo.

Criterios de aceptación:

- Endpoint GET /notas/ranking/grupo/{id}.
- Ordenar de mayor a menor.
- Mostrar nombre y promedio.

HU18 – Consultar evolución académica por periodos

Objetivo: Permitir analizar cambios en el desempeño por semestre o periodo.

Criterios de aceptación:

- Endpoint GET /notas/evolucion/estudiante/{id}.
- Agrupar por periodo.
- Retornar promedios históricos.

HU19 – Eliminar nota de forma segura

Objetivo: Controlar eliminación de notas erróneas.

Criterios de aceptación:

- Validar que la nota exista antes de eliminar.
- Confirmar eliminación con estado 204.
- Registrar log (opcional) de eliminación.

HU20 – Consultar notas por docente y materia

Objetivo: Permitir a un docente ver todas las notas que ha asignado.

Criterios de aceptación:

- Endpoint GET /notas/docente/{id}.
- Filtrar por materia o grupo opcionalmente.
- Incluir nombres de los estudiantes.

Módulo de asistencia

HU01 – Crear entidad Asistencia

Objetivo: Diseñar la entidad JPA que represente la asistencia de un estudiante.

(604) 480 88 22

Criterios de aceptación:



Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín

- La entidad debe tener fecha, estado (presente/ausente), y observaciones.
- Debe relacionarse con la entidad Estudiante y Grupo.
- Debe contar con anotaciones JPA correctas y validaciones básicas.

HU02 – Crear repositorio de asistencia

Objetivo: Implementar el repositorio JPA para gestionar datos de asistencia.

Criterios de aceptación:

- Extiende de JpaRepository.
- Tiene métodos para buscar asistencia por estudiante, por grupo y por fecha.
- Se prueba con datos de ejemplo.

HU03 – Crear DTO de asistencia

Objetivo: Estructurar un DTO para controlar los datos expuestos al cliente.

Criterios de aceptación:

- El DTO expone solo fecha, estado y observaciones.
- Usa validaciones con anotaciones como @NotNull, @Size, etc.
- Se diferencia claramente del modelo de entidad.

HU04 – Crear servicio de registro de asistencia

Objetivo: Implementar el servicio para registrar una nueva asistencia.

Criterios de aceptación:

- Recibe el DTO y guarda la entidad en base de datos.
- Lanza error si el estudiante no existe.
- Se usa ModelMapper o mapper manual.

HU05 – Crear endpoint POST para registrar asistencia

Objetivo: Desarrollar el endpoint para registrar asistencias.

Criterios de aceptación:

- Mapea el endpoint con @PostMapping.
- Devuelve código 201 si se guarda correctamente.
- Maneja errores con @ExceptionHandler.

HU06 – Listar asistencias por estudiante

Objetivo: Crear servicio y endpoint para consultar asistencias por estudiante.

Criterios de aceptación:

- El endpoint recibe el ID del estudiante.
- Devuelve un listado de asistencias.

(604) 480 88 22

Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín



- La información está en formato DTO.

HU07 – Listar asistencias por grupo y fecha

Objetivo: Implementar endpoint para consultar asistencia por grupo en un día.

Criterios de aceptación:

- Recibe el ID del grupo y fecha como parámetro.
- Devuelve listado de estudiantes con estado de asistencia.
- Se prueban casos vacíos.

HU08 – Actualizar estado de asistencia

Objetivo: Permitir modificar el estado (presente/ausente) de una asistencia.

Criterios de aceptación:

- Se accede por el ID de asistencia.
- El endpoint permite solo actualizar el estado.
- El cambio se persiste correctamente.

HU09 – Validar duplicidad de asistencia

Objetivo: Evitar que un estudiante registre asistencia duplicada para una misma fecha.

Criterios de aceptación:

- La lógica de negocio revisa si ya hay asistencia ese día.
- Devuelve error si ya existe.
- Se prueba con datos duplicados.

HU10 – Crear endpoint para asistencia diaria grupal

Objetivo: Permitir registrar múltiples asistencias en lote para un grupo.

Criterios de aceptación:

- Se recibe una lista de DTOs con asistencias.
- Se vincula a un grupo.
- Se validan todos los registros antes de guardar.

HU11 – Crear DTO resumen de asistencia por mes

Objetivo: Generar un DTO que devuelva conteo de asistencias por mes.

Criterios de aceptación:

- Contiene campos: mes, total presentes, total ausentes.
- Se agrupan los datos por estudiante.

(604) 480 88 22

Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín



- Se puede consumir desde frontend.

HU12 – Obtener porcentaje de asistencia del estudiante

Objetivo: Calcular y retornar el porcentaje de asistencia de un estudiante.

Criterios de aceptación:

- El cálculo es (asistencias presentes / total sesiones) * 100.
- Devuelve el porcentaje con dos decimales.
- Se expone en un endpoint GET.

HU13 – Endpoint para ver resumen por grupo

Objetivo: Listar todos los estudiantes con sus porcentajes de asistencia.

Criterios de aceptación:

- Se consulta por ID de grupo.
- Devuelve lista con nombre del estudiante y su porcentaje.
- Ordenado por menor porcentaje.

HU14 – Reportar asistencia con observación especial

Objetivo: Permitir agregar comentarios a una asistencia específica.

Criterios de aceptación:

- Campo de observación es opcional.
- Se puede editar posteriormente.
- Se visualiza en los listados de asistencia.

HU15 – Integrar validación de fechas futuras

Objetivo: Evitar que se registren asistencias con fechas futuras.

Criterios de aceptación:

- Se valida la fecha ingresada.
- Devuelve error si la fecha está en el futuro.
- Se testea esta lógica en el servicio.

HU16 – Crear endpoint para eliminar asistencia

Objetivo: Eliminar un registro de asistencia por su ID.

Criterios de aceptación:

- Solo accesible vía DELETE.
- Confirma la eliminación con status 200.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Lanza error si el ID no existe.

HU17 – Ver asistencias de un día específico

Objetivo: Consultar todas las asistencias registradas para una fecha dada.

Criterios de aceptación:

- Se recibe la fecha como parámetro.
- Muestra nombre del estudiante, grupo y estado.
- Ordenado alfabéticamente.

HU18 – Filtrar asistencias por rango de fechas

Objetivo: Implementar endpoint que permita consultar asistencias por periodo.

Criterios de aceptación:

- Recibe fechalinicio y fechaFin.
- Devuelve solo registros en ese rango.
- Se valida que fechalinicio < fechaFin.

HU19 – Registrar inasistencia justificada

Objetivo: Permitir marcar una ausencia como justificada.

Criterios de aceptación:

- Campo booleano “justificada” en la entidad.
- Puede editarse solo por ciertos roles (en futuras mejoras).
- Se ve reflejado en los reportes.

HU20 – Generar asistencia automática para días hábiles

Objetivo: Simular el registro automático de asistencias de estudiantes activos.

Criterios de aceptación:

- Se simula con un endpoint temporal.
- Se crean registros “ausente” si no hay asistencia.
- Se valida que no sobreescriba datos existentes.

Modulo apoyo estudiantil**Registrar necesidad de apoyo**

Objetivo: Permitir a un estudiante registrar una necesidad de apoyo personal o académico.

(604) 480 88 22

Criterios de aceptación:



Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín

- Se debe crear un modelo NecesidadApoyo con campos: tipo, descripción, fecha.
- Implementar el repositorio usando JpaRepository.
- El endpoint /api/necesidades debe permitir guardar una necesidad vía DTO.

2. Listar necesidades por estudiante

Objetivo: Consultar todas las necesidades registradas por un estudiante específico.

Criterios de aceptación:

- Crear método en el repositorio para filtrar por idEstudiante.
- Servicio debe devolver DTO con información de cada necesidad.
- Endpoint /api/necesidades/estudiante/{id} devuelve lista en JSON.

3. Actualizar necesidad registrada

Objetivo: Permitir al estudiante actualizar la información de una necesidad.

- Criterios de aceptación:
- Validar que la necesidad exista antes de modificar.
- Usar DTO con anotaciones de validación.
- Endpoint PUT /api/necesidades/{id} debe aplicar cambios.

4. Eliminar necesidad registrada

Objetivo: Permitir eliminar una necesidad que ya no sea relevante.

Criterios de aceptación:

- Verificar existencia previa antes de eliminar.
- Repositorio implementa método deleteById.
- Endpoint DELETE /api/necesidades/{id} responde con estado 200.

5. Registrar tipo de apoyo disponible

Objetivo: Permitir al personal registrar tipos de apoyo ofrecidos por bienestar.

Criterios de aceptación:

- Crear entidad TipoApoyo con nombre y descripción.
- Usar DTO para validación del nombre (no vacío).
- Endpoint POST /api/tipos-apoyo debe guardar nuevo tipo.

6. Listar tipos de apoyo

Objetivo: Consultar todos los tipos de apoyo disponibles.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Servicio debe devolver lista de DTOs.
- Repositorio usa findAll().
- Endpoint GET /api/tipos-apoyo debe estar disponible.

7. Asignar profesional a una necesidad

Objetivo: Permitir asignar un psicólogo, docente u orientador a una necesidad.

Criterios de aceptación:

- Crear relación entre Profesional y NecesidadApoyo.
- Validar existencia del profesional y necesidad.
- Endpoint POST /api/necesidades/{id}/asignar-profesional.

8. Consultar necesidades asignadas a un profesional

Objetivo: Permitir a un profesional ver las necesidades a su cargo.

Criterios de aceptación:

- Método en repositorio filtra por idProfesional.
- Servicio mapea resultados a DTO.
- Endpoint GET /api/profesionales/{id}/necesidades.

9. Registrar seguimiento a una necesidad

Objetivo: Permitir al profesional registrar acciones realizadas.

Criterios de aceptación:

- Crear entidad Seguimiento con fecha, descripción y estado.
- Asociar seguimiento a necesidad.
- Endpoint POST /api/seguimientos registra nuevo seguimiento.

10. Ver historial de seguimiento de una necesidad

Objetivo: Consultar el historial de atención de una necesidad.

Criterios de aceptación:

- Crear relación entre necesidad y múltiples seguimientos.
- Servicio devuelve DTOs ordenados por fecha.
- Endpoint GET /api/necesidades/{id}/seguimientos.

11. Registrar profesional de apoyo

Objetivo: Permitir registrar un nuevo profesional de apoyo.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

Criterios de aceptación:

- Modelo Profesional con nombre, área y contacto.
- Validaciones en DTO.
- Endpoint POST /api/profesionales.

12. Listar profesionales registrados

Objetivo: Consultar todos los profesionales disponibles en bienestar.

Criterios de aceptación:

- Método en servicio llama a findAll().
- DTO con nombre, cargo y correo.
- Endpoint GET /api/profesionales.

13. Actualizar datos de profesional

Objetivo: Editar información de contacto o especialidad.

Criterios de aceptación:

- DTO debe validar formato del correo electrónico.
- Servicio debe verificar existencia.
- Endpoint PUT /api/profesionales/{id}.

14. Eliminar registro de profesional

Objetivo: Quitar un profesional del sistema cuando ya no pertenece.

Criterios de aceptación:

- Servicio debe validar que no tenga necesidades asignadas.
- Endpoint DELETE /api/profesionales/{id}.

15. Registrar categoría de necesidad

Objetivo: Clasificar las necesidades en categorías (emocional, económica, etc.).

Criterios de aceptación:

- Modelo CategoriaNecesidad con nombre único.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Validaciones: nombre no repetido.
- Endpoint POST /api/categorias-necesidad.

16. Listar categorías de necesidad

Objetivo: Permitir que los usuarios conozcan las categorías existentes.

Criterios de aceptación:

- Servicio devuelve lista con nombres.
- Repositorio usa findAll.
- Endpoint GET /api/categorias-necesidad.

17. Filtrar necesidades por categoría

Objetivo: Consultar necesidades asociadas a una categoría específica.

Criterios de aceptación:

- Método en repositorio filtra por idCategoria.
- Servicio devuelve DTO con tipo y estado.
- Endpoint GET /api/necesidades/categoría/{id}.

18. Establecer prioridad a una necesidad

Objetivo: Definir niveles de atención (alta, media, baja).

Criterios de aceptación:

- Agregar campo prioridad al modelo.
- Validar solo tres valores posibles.
- DTO debe incluir campo y validación.

19. Cambiar estado de una necesidad

Objetivo: Permitir cambiar el estado (pendiente, en atención, cerrada).

Criterios de aceptación:

- Servicio debe validar transición válida.
- Endpoint PUT /api/necesidades/{id}/estado.
- Respuesta con mensaje de confirmación.

20. Generar reporte de necesidades por tipo

Objetivo: Permitir ver cuántas necesidades hay por tipo de apoyo.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

Criterios de aceptación:

- Crear DTO de resumen: tipo → cantidad.
- Repositorio debe usar @Query con GROUP BY.
- Endpoint GET /api/reportes/necesidades-tipo

Modulo Estadísticas

HU01 - Consultar asistencia por grupo

Objetivo: Generar estadísticas de asistencia por grupo.

Criterios de aceptación:

- Crear entidad Asistencia con relación a Grupo.
- Implementar repositorio con consulta agrupada por grupo.
- Exponer endpoint que devuelva resumen de asistencia en DTO.

HU02 - Obtener promedio de notas por estudiante

Objetivo: Calcular el promedio de notas de cada estudiante.

Criterios de aceptación:

- Crear DTO con nombre del estudiante y su promedio.
- Implementar método de agregación en servicio.
- Exponer endpoint GET /estadisticas/notas/promedio.

HU03 - Generar ranking académico por grupo

Objetivo: Listar estudiantes ordenados por promedio académico dentro de un grupo.

Criterios de aceptación:

- Crear DTO RankingEstudianteDTO con nombre, grupo y promedio.
- Implementar método en servicio para ordenar los estudiantes.
- Exponer endpoint GET /estadisticas/ranking.

HU04 - Consultar evolución académica por estudiante

Objetivo: Mostrar variación de calificaciones por periodo o materia.

Criterios de aceptación:

(604) 480 88 22

Cesde.edu.co

Calle 49 41-9

Colombia - Antioquia - Medellín



- Crear DTO con lista de notas por fecha o materia.
- Implementar servicio que agrupe por estudiante.
- Exponer endpoint GET /estadisticas/evolucion/{idEstudiante}.

HU05 - Visualizar indicadores de asistencia mensual

Objetivo: Ver reporte de asistencia mensual por estudiante.

Criterios de aceptación:

- Crear modelo de consulta agrupada por mes.
- Generar DTO con fecha y porcentaje de asistencia.
- Exponer endpoint con filtros de fecha.

HU06 - Ver participación en reportes por grupo

Objetivo: Medir cuántos estudiantes por grupo han hecho reportes en bienestar.

Criterios de aceptación:

- Consultar desde módulo de apoyo cuántos reportes se han hecho.
- Generar DTO ParticipacionDTO.
- Exponer endpoint GET /estadisticas/reportes-bienestar.

HU07 - Identificar estudiantes con bajo rendimiento

Objetivo: Listar estudiantes cuyo promedio esté por debajo de cierto umbral.

Criterios de aceptación:

- Definir parámetro configurable de corte de rendimiento.
- Crear consulta en servicio con filtro por promedio.
- Exponer endpoint GET /estadisticas/bajo-rendimiento.

HU08 - Generar gráficos de asistencia

Objetivo: Proveer datos para graficar asistencia en frontend.

Criterios de aceptación:

- Crear DTO para estructura de datos tipo gráfica (eje X/Y).
- Implementar endpoint que agrupe datos por grupo.
- Validar formato JSON compatible con librerías de frontend.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU09 - Resumen estadístico por materia

Objetivo: Mostrar estadísticas (promedio, desviación, mínimo, máximo) por materia.

Criterios de aceptación:

- Crear clase DTO ResumenMateriaDTO.
- Implementar cálculo estadístico en el servicio.
- Exponer endpoint GET /estadisticas/resumen-materia.

HU10 - Ver porcentaje de estudiantes con dificultades

Objetivo: Calcular cuántos estudiantes tienen alertas activas por área.

Criterios de aceptación:

- Consultar alertas del módulo de apoyo.
- Generar porcentaje sobre total de estudiantes.
- Exponer datos en endpoint para dashboards.

HU11 - Analizar tendencias de notas por grupo

Objetivo: Comparar promedio de grupos en distintas materias.

Criterios de aceptación:

- Agrupar promedios por grupo y materia.
- Crear DTO para comparar tendencias.
- Exponer endpoint con datos para línea de tiempo.

HU12 - Ver evolución de participación familiar

Objetivo: Mostrar cuántos familiares consultan la plataforma.

Criterios de aceptación:

- Recolectar logs o eventos del módulo familiar.
- Crear DTO de participación con fechas.
- Exponer endpoint GET /estadisticas/familiares/uso.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

HU13 - Calcular tasa de inasistencia por mes

Objetivo: Obtener porcentaje de inasistencia por grupo y mes.

Criterios de aceptación:

- Crear consulta SQL/JPA que agrupe por mes.
- Crear DTO InasistenciaMensualDTO.
- Exponer datos con filtro por grupo.

HU14 - Analizar solicitudes por tipo en bienestar

Objetivo: Ver cuántas solicitudes se hacen por tipo de apoyo.

Criterios de aceptación:

- Obtener tipos de apoyo desde módulo de bienestar.
- Agrupar por cantidad.
- Exponer en endpoint para visualización.

HU15 - Ver tasa de estudiantes con mejoras

Objetivo: Identificar estudiantes que mejoraron su rendimiento en el tiempo.

Criterios de aceptación:

- Comparar promedio entre dos períodos.
- Crear DTO con delta de mejora.
- Exponer endpoint GET /estadisticas/mejoras.

HU16 - Generar resumen por cohorte

Objetivo: Mostrar indicadores generales de cada cohorte de ingreso.

Criterios de aceptación:

- Agrupar estudiantes por año o periodo de ingreso.
- Calcular estadísticas agregadas.
- Crear endpoint GET /estadisticas/cohorte.

HU17 - Consultar distribución por género

(604) 480 88 22



Objetivo: Visualizar el porcentaje de estudiantes por género.

Cesde.edu.co



Calle 49 41-9

Colombia - Antioquia - Medellín



Criterios de aceptación:

- Extraer datos de entidad Usuario o Estudiante.
- Calcular porcentaje por género.
- Exponer datos para pie chart.

HU18 - Exportar estadísticas a CSV

Objetivo: Permitir exportar reportes a formato descargable.

Criterios de aceptación:

- Generar archivo CSV desde DTOs.
- Incluir encabezados claros y datos filtrados.
- Exponer endpoint GET /estadisticas/exportar?formato=csv.

HU19 - Visualizar alertas tempranas

Objetivo: Integrar reportes de rendimiento y asistencia para generar alertas.

Criterios de aceptación:

- Detectar patrones de riesgo (bajo rendimiento + alta inasistencia).
- Generar DTO con lista de alertas activas.
- Exponer endpoint GET /estadisticas/alertas.

HU20 - Integrar panel general de indicadores

Objetivo: Consolidar en un solo endpoint los principales KPIs del sistema.

Criterios de aceptación:

- Crear DTO DashboardIndicadoresDTO con asistencia, notas, reportes, etc.
- Consultar múltiples servicios en un solo método.
- Exponer endpoint GET /dashboard/indicadores.

FRONTEND (React)

Módulo de Ingreso y Registro

(604) 480 88 22



HUF1. Yo como usuario quiero visualizar el formulario de registro con campos validados.

Cesde.edu.co



Calle 49 41-9
Colombia - Antioquia - Medellín



Objetivo: Visualizar y validar campos del formulario de registro usando componentes React.

Criterios de aceptación:

- Debe tener componentes reutilizables para campos de texto, contraseña y selección.
- Validaciones en tiempo real usando estados de React.
- El formulario se debe estilizar usando un framework CSS (Tailwind, Bootstrap).

HUF2. Yo como usuario quiero recibir confirmación visual al completar el registro.

Objetivo: Confirmar visualmente el éxito del registro usando notificaciones o redirecciones.

Criterios de aceptación:

- Debe mostrarse una alerta de éxito o snack bar.
- Debe redirigirse automáticamente al login en 3 segundos.
- El botón debe desactivarse mientras se procesa la petición.

Módulo Hoja de Vida Estudiantil

HUH1. Yo como estudiante quiero visualizar y editar mi hoja de vida en una vista ordenada.

Objetivo: Crear componentes visuales para mostrar y editar información personal y académica.

Criterios de aceptación:

- Se deben usar componentes organizados por secciones (educación, habilidades, etc.).
- Permitir edición mediante formularios dinámicos.
- Usar hooks para manejar estado del formulario.

HUH2. Yo como estudiante quiero cargar mi foto de perfil desde el frontend.

Objetivo: Subir imágenes al servidor desde la hoja de vida.

Criterios de aceptación:

(604) 480 88 22

 Cesde.edu.co

 Calle 49 41-9
Colombia - Antioquia - Medellín

- Se debe permitir vista previa antes de subir.
- Validar tipo y tamaño del archivo.
- Integrar con API para guardar en el backend.

Módulo Familiar

HUFAM1. Yo como acudiente quiero consultar el perfil académico del estudiante.

Objetivo: Mostrar datos del estudiante al familiar en una vista controlada.

Criterios de aceptación:

- Mostrar los datos personales y el resumen académico del estudiante.
- Componente protegido según el rol "familiar".
- Estilo claro y organizado, con uso de tablas o tarjetas.

HUFAM2. Yo como acudiente quiero recibir alertas visuales si el estudiante tiene bajo rendimiento.

Objetivo: Mostrar alertas con íconos y colores según desempeño.

Criterios de aceptación:

- Mostrar mensajes si hay materias reprobadas.
- Mostrar recomendaciones en tarjetas separadas.
- Usar íconos o color para representar alertas.

Módulo Académico

HUA1. Yo como estudiante quiero ver mis notas en una tabla clara y exportable.

Objetivo: Presentar notas con posibilidad de filtrado y exportación.

Criterios de aceptación:

- Usar tabla ordenable con filtros por materia o semestre.
- Agregar botón de descarga como PDF o Excel.
- Estilizar con librería visual.

HUA2. Yo como estudiante quiero ver recomendaciones cuando obtengo bajo desempeño.

Objetivo: Mostrar ayudas sugeridas al usuario según su nota.

Criterios de aceptación:

(604) 480 88 22

Cesde.edu.co

Calle 49 41-9
Colombia - Antioquia - Medellín



- Si la nota es inferior a 3.0, debe aparecer una tarjeta de sugerencia.
- Recomendaciones deben consumirse desde la API.
- Diseño responsivo y accesible.

Módulo Asistencia

HUAS1. Yo como docente quiero ver la asistencia del grupo en una tabla interactiva.

Objetivo: Visualizar asistencia por fecha y estudiante con filtros.

Criterios de aceptación:

- Mostrar tabla de asistencia con filtro por fecha.
- Permitir expandir fila para ver detalle.
- Colores según estado (presente, ausente, justificado).

HUAS2. Yo como estudiante quiero ver mi porcentaje de asistencia.

Objetivo: Calcular y mostrar asistencia total en componente visual.

Criterios de aceptación:

- Mostrar un gráfico de progreso (circular o barra).
- Mostrar porcentaje total y por materia.
- Estilos visuales atractivos.

Módulo de Apoyo y Bienestar

HUAW1. Yo como estudiante quiero registrar una solicitud de apoyo.

Objetivo: Crear formulario para enviar solicitud al área de bienestar.

Criterios de aceptación:

- Selección del tipo de apoyo (psicológico, académico, económico).
- Campo de texto con descripción.
- Confirmación visual del envío.

HUAW2. Yo como coordinador quiero revisar solicitudes desde el frontend.

Objetivo: Mostrar solicitudes en lista filtrable y ordenable.

(604) 480 88 22



Criterios de aceptación:

Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Listado paginado y ordenado por fecha.
- Permitir marcar solicitudes como atendidas.
- Estilo limpio y diferenciación por tipo de solicitud.

Módulo Estadístico

HUE1. Yo como docente quiero ver un dashboard de estadísticas académicas.

Objetivo: Mostrar gráficos y KPIs relevantes en vista principal.

Criterios de aceptación:

- Mostrar gráfico de barras con promedios por materia.
- Mostrar tarjetas con métricas claves (asistencia, rendimiento).
- Visualización responsive y atractiva.

HUE2. Yo como usuario quiero filtrar los datos estadísticos por grupo y periodo.

Objetivo: Agregar filtros dinámicos que actualicen gráficos.

Criterios de aceptación:

- Selectores desplegables de grupo y fecha.
- Gráficos se deben actualizar automáticamente.
- Mostrar loading mientras carga.

PYTHON - PANDAS

Módulo de Ingreso y Registro

HUP1. Yo como analista quiero analizar cuántos usuarios nuevos se registran por semana.

Objetivo: Usar pandas para agrupar y visualizar usuarios por fecha de registro.

Criterios de aceptación:

- Leer datos desde CSV o base de datos.
- Agrupar por semana y contar registros.
- Mostrar resultado como DataFrame limpio.

HUP2. Yo como analista quiero identificar el tipo de usuarios más registrados.

Objetivo: Agrupar usuarios por tipo (estudiante, familiar, docente).

(604) 480 88 22



Criterios de aceptación:

Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Hacer value_counts sobre la columna tipo.
- Mostrar porcentaje de cada tipo.
- Visualización con seaborn opcional.

Módulo Hoja de Vida

HUPH1. Yo como analista quiero saber cuántos estudiantes registran hoja de vida completa.

Objetivo: Validar registros completos y contarlos.

Criterios de aceptación:

- Detectar nulos o campos vacíos.
- Crear una columna 'completado' booleana.
- Agrupar por estado y calcular proporciones.

HUPH2. Yo como analista quiero encontrar las habilidades más frecuentes en los perfiles.

Objetivo: Extraer y contar habilidades de una columna de texto.

Criterios de aceptación:

- Procesar texto con .str.split() y explode().
- Contar ocurrencias de cada habilidad.
- Exportar top 10 a CSV.

Módulo Familiar

HUPF1. Yo como analista quiero analizar cuántos familiares consultan el perfil del estudiante.

Objetivo: Analizar logs o registros de acceso.

Criterios de aceptación:

- Agrupar por ID de estudiante.
- Contar número de accesos.
- Detectar estudiantes sin interacción.

HUPF2. Yo como analista quiero conocer en qué horarios acceden más los familiares.

Objetivo: Analizar accesos por hora del día.

(604) 480 88 22



Criterios de aceptación:

Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Extraer hora de columna timestamp.
- Agrupar por franja horaria.
- Mostrar histograma o conteo.

Módulo Académico

HUPA1. Yo como analista quiero calcular el promedio general por grupo.

Objetivo: Agrupar calificaciones por grupo y calcular media.

- Criterios de aceptación:
- Agrupar por grupo_id.
- Calcular .mean() de nota.

Ordenar de mayor a menor.

HUPA2. Yo como analista quiero identificar materias con mayor índice de reprobación.

Objetivo: Contar estudiantes con nota < 3.0 por materia.

Criterios de aceptación:

- Filtrar DataFrame por notas < 3.0.
- Agrupar por materia.
- Mostrar porcentaje de reprobación.

Módulo Asistencia

HUPAS1. Yo como analista quiero calcular la asistencia promedio por estudiante.

Objetivo: Evaluar asistencia total de cada estudiante.

Criterios de aceptación:

- Agrupar por estudiante_id.
- Calcular porcentaje de asistencia.
- Exportar resultados.

HUPAS2. Yo como analista quiero detectar estudiantes con ausencias recurrentes.

Objetivo: Filtrar quienes tengan más de 3 ausencias.

Criterios de aceptación:

(604) 480 88 22

Cesde.edu.co



Calle 49 41-9

Colombia - Antioquia - Medellín



Contar registros con estado 'ausente'.

Filtrar estudiantes con más de 3.

Mostrar DataFrame con resultados.

Módulo Apoyo y Bienestar

HUPAW1. Yo como analista quiero saber qué tipos de apoyo se solicitan más.

Objetivo: Clasificar y contar solicitudes por tipo.

Criterios de aceptación:

- Agrupar por tipo de apoyo.
- Mostrar top 3 apoyos más comunes.
- Representación en gráfico de pastel.

HUPAW2. Yo como analista quiero conocer la frecuencia de solicitudes por mes.

Objetivo: Analizar temporalidad de las solicitudes.

Criterios de aceptación:

- Extraer mes del timestamp.
- Agrupar por mes y contar.

Módulo Estadístico

HUPE1. Yo como analista quiero generar un resumen estadístico por grupo.

Objetivo: Usar describe() para generar estadística básica por grupo.

Criterios de aceptación:

- Agrupar por grupo_id.
- Aplicar .describe() sobre nota, asistencia.
- Guardar resumen en archivo.

HUPE2. Yo como analista quiero identificar correlaciones entre nota y asistencia.

Objetivo: Calcular correlaciones para análisis.

(604) 480 88 22



Criterios de aceptación:

Cesde.edu.co



Calle 49 41-9



Colombia - Antioquia - Medellín

- Unir datasets de notas y asistencia.
- Usar .corr() entre columnas numéricas.
- Representar con heatmap.

(604) 480 88 22



Cesde.edu.co



Calle 49 41-9
Colombia - Antioquia - Medellín

