

Proyecto #2
Entrega Única

Grupo 5

Juan Felipe Sáenz - 202311148

Miguel Angel Velandia - 202312487

Alejandro Cruz Acevedo - 201912149

Universidad de Los Andes

**Departamento de Ingeniería de Sistemas y
Computación**

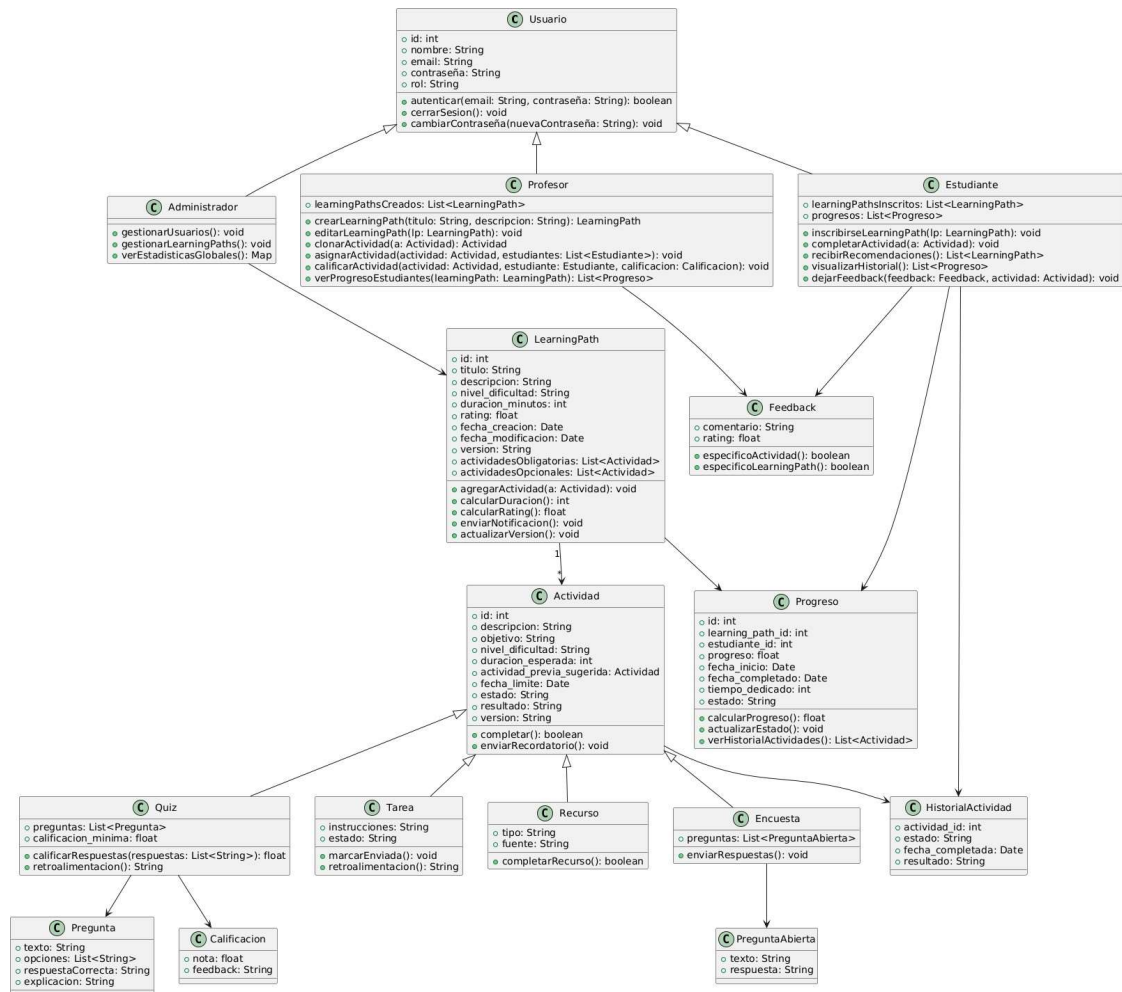
**Diseño y Programación Orientada a Objetos – ISIS
1226**

2024-2

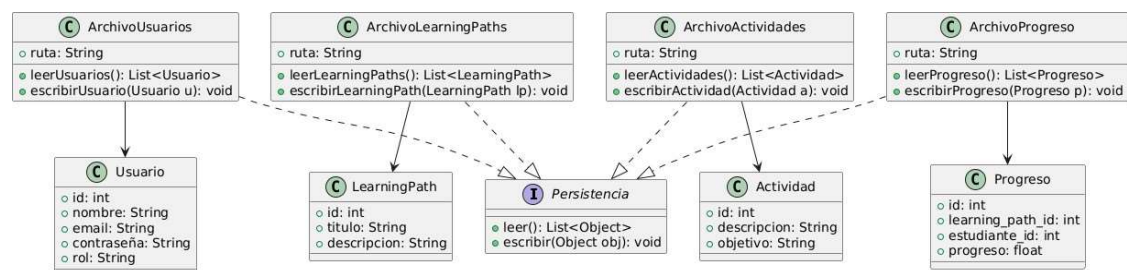
I. Contexto del problema

Este proyecto se centra en el desarrollo de un Learning Path Recommendation System que permita a los profesores crear y gestionar caminos de aprendizaje personalizados y a los estudiantes seguirlos, recibir recomendaciones y monitorear su progreso. El sistema incluirá autenticación de usuarios, persistencia de datos en archivos externos, y soporte para distintos tipos de actividades: clases, tareas, quices, exámenes y encuestas.

II. Diagramas



Modelo de Conceptos (Persistencia)



a. Entidades

1. **Usuario**
 2. **Administrador**
 3. **Profesor**
 4. **Estudiante**
 5. **Learning Path**
 6. **Actividad**
 - Quiz
 - Tarea
 - Recurso
 - Encuesta
 7. **Progreso**
 8. **HistorialActividad**
 9. **Pregunta**
 10. **PreguntaAbierta**
 11. **Calificacion**
 12. **Feedback**
-

b. Características (atributos y métodos) de las entidades

1. **Usuario**
 - **Atributos:**
 - `id`: int - Identificador único del usuario.
 - `nombre`: String - Nombre completo del usuario.
 - `email`: String - Correo electrónico.
 - `contraseña`: String - Contraseña encriptada.
 - `rol`: String - Rol del usuario (puede ser "profesor", "estudiante" o "administrador").
 - **Métodos:**
 - `autenticar(email: String, contraseña: String): boolean` - Valida si las credenciales del usuario son correctas.
 - `cerrarSesion(): void` - Finaliza la sesión del usuario.
 - `cambiarContraseña(nuevaContraseña: String): void` - Permite al usuario cambiar su contraseña.
2. **Administrador (hereda de Usuario)**
 - **Métodos:**
 - `gestionarUsuarios(): void` - Gestiona la creación, modificación y eliminación de usuarios.
 - `gestionarLearningPaths(): void` - Gestiona la creación y eliminación de Learning Paths globales.
 - `verEstadisticasGlobales(): Map` - Ver estadísticas generales del sistema, como el número de usuarios, cantidad de Learning Paths, etc.

3. Profesor (hereda de Usuario)

0 Atributos:

- `learningPathsCreados: List<LearningPath>` - Lista de Learning Paths creados por el profesor.

0 Métodos:

- `crearLearningPath(titulo: String, descripcion: String): LearningPath` - Crea un nuevo Learning Path.
- `editarLearningPath(lp: LearningPath): void` - Permite editar un Learning Path existente.
- `clonarActividad(a: Actividad): Actividad` - Clona una actividad de otro profesor y permite modificarla.
- `asignarActividad(actividad: Actividad, estudiantes: List<Estudiante>): void` - Asigna una actividad a estudiantes.
- `calificarActividad(actividad: Actividad, estudiante: Estudiante, calificacion: Calificacion): void` - Califica una actividad completada por un estudiante.
- `verProgresoEstudiantes(learningPath: LearningPath): List<Progreso>` - Visualiza el progreso de todos los estudiantes inscritos en un Learning Path.

4. Estudiante (hereda de Usuario)

0 Atributos:

- `learningPathsInscritos: List<LearningPath>` - Lista de Learning Paths en los que está inscrito.
- `progresos: List<Progreso>` - Progreso del estudiante en cada Learning Path.

0 Métodos:

- `inscribirseLearningPath(lp: LearningPath): void` - Inscribe al estudiante en un Learning Path.
- `completarActividad(a: Actividad): void` - Marca una actividad como completada.
- `recibirRecomendaciones(): List<LearningPath>` - Recibe recomendaciones de caminos de aprendizaje basados en su progreso e intereses.
- `visualizarHistorial(): List<Progreso>` - Visualiza el historial de actividades completadas en todos los Learning Paths.
- `dejarFeedback(feedback: Feedback, actividad: Actividad): void` - Deja un comentario o calificación sobre una actividad.

5. Learning Path

0 Atributos:

- `id: int` - Identificador único del Learning Path.
- `titulo: String` - Título del camino de aprendizaje.
- `descripcion: String` - Breve descripción del contenido y objetivos.

- `nivel_dificultad`: String - Nivel de dificultad (básico, intermedio, avanzado).
- `duracion_minutos`: int - Duración total en minutos (calculada a partir de la duración de las actividades).
- `rating`: float - Promedio de calificaciones recibidas por los estudiantes.
- `fecha_creacion`: Date - Fecha de creación del Learning Path.
- `fecha_modificacion`: Date - Fecha de la última modificación.
- `version`: String - Número de versión.
- `actividadesObligatorias`: List<Actividad> - Lista de actividades obligatorias.
- `actividadesOpcionales`: List<Actividad> - Lista de actividades opcionales.

0 **Métodos:**

- `agregarActividad(a: Actividad): void` - Agrega una actividad al Learning Path.
- `calcularDuracion(): int` - Calcula la duración total del Learning Path.
- `calcularRating(): float` - Calcula el promedio de las calificaciones dadas por los estudiantes.
- `enviarNotificacion(): void` - Envía una notificación a los estudiantes inscritos sobre actividades nuevas o pendientes.
- `actualizarVersion(): void` - Actualiza la versión del Learning Path cuando se realizan cambios importantes.

6. Actividad

0 **Atributos:**

- `id`: int - Identificador único de la actividad.
- `descripcion`: String - Descripción detallada de la actividad.
- `objetivo`: String - Objetivo de aprendizaje de la actividad.
- `nivel_dificultad`: String - Dificultad de la actividad.
- `duracion_esperada`: int - Tiempo estimado para completar la actividad.
- `actividad_previa_sugerida`: Actividad - Actividad recomendada como pre-requisito.
- `fecha_limite`: Date - Fecha límite sugerida para completar la actividad.
- `estado`: String - Estado de la actividad (completada, pendiente, en progreso).
- `resultado`: String - Resultado de la actividad (éxito, fracaso, etc.).
- `version`: String - Número de versión de la actividad.

0 **Métodos:**

- `completar(): boolean` - Marca la actividad como completada.
- `enviarRecordatorio(): void` - Envía un recordatorio al estudiante sobre la actividad pendiente.

7. Quiz (hereda de Actividad)

0 Atributos:

- `preguntas: List<Pregunta>` - Lista de preguntas del quiz.
- `calificacion_minima: float` - Calificación mínima para aprobar el quiz.

0 Métodos:

- `calificarRespuestas(respuestas: List<String>): float` - Calcula la calificación del estudiante según sus respuestas.
- `retroalimentacion(): String` - Devuelve una retroalimentación sobre las respuestas correctas e incorrectas.

8. Tarea (hereda de Actividad)

0 Atributos:

- `instrucciones: String` - Instrucciones para completar la tarea.
- `estado: String` - Estado de la tarea (enviada, pendiente, etc.).

0 Métodos:

- `marcarEnviada(): void` - Marca la tarea como enviada.
- `retroalimentacion(): String` - Proporciona feedback sobre la tarea entregada.

9. Recurso (hereda de Actividad)

0 Atributos:

- `tipo: String` - Tipo de recurso (video, libro, sitio web, etc.).
- `fuelle: String` - Fuente o URL del recurso.

0 Métodos:

- `completarRecurso(): boolean` - Marca el recurso como completado por el estudiante.

10. Encuesta (hereda de Actividad)

0 Atributos:

- `preguntas: List<PreguntaAbierta>` - Lista de preguntas abiertas.

0 Métodos:

- `enviarRespuestas(): void` - Envía las respuestas de la encuesta.

11. Progreso

0 Atributos:

- `id: int` - Identificador único del progreso.
- `learning_path_id: int` - Referencia al Learning Path en el que está inscrito el estudiante.
- `estudiante_id: int` - Referencia al estudiante.
- `progreso: float` - Porcentaje completado del Learning Path.
- `fecha_inicio: Date` - Fecha en la que el estudiante comenzó el Learning Path.
- `fecha_completado: Date` - Fecha en la que el estudiante completó el Learning Path (si corresponde).
- `tiempo_dedicado: int` - Tiempo total dedicado al Learning Path.
- `estado: String` - Estado del progreso (en progreso, completado, etc.).

0 Métodos:

- `calcularProgreso(): float` - Calcula el porcentaje de actividades completadas.
- `actualizarEstado(): void` - Actualiza el estado del progreso (completado, en progreso, etc.).
- `verHistorialActividades(): List<Actividad>` - Muestra el historial de actividades completadas.

12. HistorialActividad

◦ Atributos:

- `actividad_id: int` - Identificador de la actividad.
- `estado: String` - Estado de la actividad (completada, en progreso, etc.).
- `fecha_completada: Date` - Fecha en la que se completó la actividad.
- `resultado: String` - Resultado de la actividad.

13. Pregunta

◦ Atributos:

- `texto: String` - Texto de la pregunta.
- `opciones: List<String>` - Opciones de respuesta.
- `respuestaCorrecta: String` - Respuesta correcta.
- `explicacion: String` - Explicación de la respuesta correcta.

14. PreguntaAbierta

◦ Atributos:

- `texto: String` - Texto de la pregunta.
- `respuesta: String` - Respuesta proporcionada por el estudiante.

15. Calificacion

◦ Atributos:

- `nota: float` - Calificación obtenida.
- `feedback: String` - Comentarios sobre la calificación.

16. Feedback

◦ Atributos:

- `comentario: String` - Comentario dejado por el estudiante.
- `rating: float` - Calificación en forma de estrellas o número.

◦ Métodos:

- `especificoActividad(): boolean` - Verifica si el feedback está relacionado con una actividad específica.
- `especificoLearningPath(): boolean` - Verifica si el feedback está relacionado con todo el Learning Path.

c. Relaciones entre las entidades, incluyendo asociaciones y herencia

1. Relaciones de herencia:

- `Administrador`, `Profesor`, y `Estudiante` heredan de `Usuario`.
- `Quiz`, `Tarea`, `Recurso`, y `Encuesta` heredan de `Actividad`.

2. Asociaciones:

- **Usuario - Learning Path:**
 - Un profesor puede crear y gestionar múltiples **LearningPath**.
 - Un estudiante puede inscribirse en varios **LearningPath**.
- **Learning Path - Actividades:**
 - Un **LearningPath** está compuesto por una secuencia de **Actividad**. Estas actividades se dividen en obligatorias y opcionales.
- **Estudiante - Progreso:**
 - Un estudiante tiene un **Progreso** asociado a cada **LearningPath** en el que está inscrito.
- **Actividad - HistorialActividad:**
 - Cada **Actividad** completada por un estudiante genera un registro en **HistorialActividad**, donde se almacena el estado, fecha y resultado.
- **Quiz - Pregunta:**
 - Un **Quiz** tiene una lista de preguntas de opción múltiple (**Pregunta**).
- **Encuesta - PreguntaAbierta:**
 - Una **Encuesta** tiene una lista de preguntas abiertas (**PreguntaAbierta**).
- **Estudiante - Feedback:**
 - Los estudiantes pueden dejar comentarios y calificaciones (**Feedback**) sobre las actividades o los Learning Paths.
- **Profesor - Feedback:**
 - Los profesores pueden ver y gestionar el **Feedback** recibido.

III. Descripción con las restricciones del proyecto

1. Persistencia de datos:

- Toda la información debe almacenarse en archivos (planos o binarios) dentro de una carpeta externa al código fuente. No se permite utilizar bases de datos, y los archivos deben estructurarse de forma organizada para facilitar la lectura y escritura.
- Los archivos pueden estar separados por entidad, como usuarios, actividades, caminos de aprendizaje, y progreso de los estudiantes. Se debe garantizar que solo la aplicación pueda leer y modificar estos archivos.

2. Autenticación obligatoria:

- Todos los usuarios deben tener un login y una contraseña. Solo los usuarios autenticados pueden acceder a las funcionalidades del sistema.
- Los usuarios se dividen en tres roles: administrador, profesor y estudiante. Cada rol tiene acceso a diferentes funcionalidades:
 - **Administrador:** Gestión de usuarios y caminos de aprendizaje.
 - **Profesor:** Creación y gestión de caminos de aprendizaje y actividades.
 - **Estudiante:** Inscripción y participación en los caminos de aprendizaje, con posibilidad de dejar feedback.

3. Gestión de versiones:

- Tanto los Learning Paths como las actividades deben manejar versiones, de forma que cuando se edite un camino de aprendizaje o actividad, se actualice su número de versión.
- Las actividades pueden ser clonadas por otros profesores, pero siempre deben mantener la versión original.

4. Actividades opcionales y obligatorias:

- Los caminos de aprendizaje deben tener actividades obligatorias y opcionales. El progreso del estudiante se calculará con base en la finalización de actividades obligatorias, aunque las actividades opcionales pueden proporcionar puntos adicionales.

5. Sistema de notificaciones:

- El sistema debe enviar notificaciones y recordatorios sobre actividades pendientes y fechas límite, de manera que los estudiantes estén informados sobre las próximas tareas que deben completar.

6. Escalabilidad futura:

- El sistema debe permitir en el futuro la integración con plataformas de gestión de aprendizaje (LMS) para automatizar procesos como la entrega de tareas y quizzes.
- Las actividades pueden ampliarse en el futuro para incluir nuevos tipos.

7. Ejecución de programas de prueba:

- Se debe implementar un conjunto de pruebas unitarias para garantizar que las funcionalidades de cada clase funcionan correctamente.

IV. Historias de usuario

1. Historia de Usuario: Registro de Usuario

- **Como** un nuevo usuario (estudiante o profesor),
- **Quiero** registrarme en el sistema,
- **Para** poder acceder a mis Learning Paths y actividades personalizadas.
- **Criterios de Aceptación:**
 - El sistema debe permitir al usuario ingresar su información (nombre, email, contraseña).
 - El sistema debe validar que el email no esté ya registrado.
 - El usuario debe recibir una confirmación de registro exitoso.

2. Historia de Usuario: Inicio de Sesión

- **Como** un usuario registrado,
- **Quiero** iniciar sesión en el sistema,
- **Para** acceder a mis Learning Paths y actividades.
- **Criterios de Aceptación:**
 - El sistema debe permitir al usuario ingresar su email y contraseña.
 - El sistema debe validar las credenciales y permitir el acceso si son correctas.
 - El sistema debe mostrar un mensaje de error si las credenciales son incorrectas.

3. Historia de Usuario: Crear Learning Path

- **Como** un profesor,
- **Quiero** crear un Learning Path,
- **Para** que mis estudiantes puedan seguirlo y aprender de manera estructurada.
- **Criterios de Aceptación:**
 - El sistema debe permitir al profesor ingresar un título, descripción, nivel de dificultad, duración y rating del Learning Path.
 - El sistema debe guardar el Learning Path y mostrarlo en la lista de Learning Paths disponibles.

4. Historia de Usuario: Asignar Actividad a Learning Path

- **Como** un profesor,
- **Quiero** asignar actividades a un Learning Path,
- **Para** que los estudiantes puedan completarlas como parte de su aprendizaje.
- **Criterios de Aceptación:**
 - El sistema debe permitir al profesor seleccionar actividades existentes y agregarlas a un Learning Path.
 - El sistema debe mostrar un mensaje de confirmación al agregar actividades exitosamente.

5. Historia de Usuario: Inscribirse en Learning Path

- **Como** un estudiante,
- **Quiero** inscribirme en un Learning Path,
- **Para** seguir las actividades y mejorar mis habilidades.
- **Criterios de Aceptación:**
 - El sistema debe mostrar una lista de Learning Paths disponibles.
 - El estudiante debe poder seleccionar un Learning Path y confirmarlo para inscribirse.
 - El sistema debe mostrar un mensaje de éxito al inscribirse.

6. Historia de Usuario: Completar Actividad

- **Como** un estudiante,
- **Quiero** completar una actividad,
- **Para** avanzar en mi Learning Path y registrar mi progreso.
- **Criterios de Aceptación:**
 - El sistema debe permitir al estudiante marcar una actividad como completada.
 - El sistema debe actualizar el estado de la actividad y registrar el resultado.
 - El sistema debe mostrar un mensaje de confirmación al completar la actividad.

7. Historia de Usuario: Consultar Progreso

- **Como** un estudiante,
- **Quiero** consultar mi progreso en un Learning Path,
- **Para** saber cuántas actividades he completado y cuánto me falta.
- **Criterios de Aceptación:**
 - El sistema debe mostrar el porcentaje de progreso del estudiante en el Learning Path.
 - El sistema debe listar las actividades completadas y pendientes.

8. Historia de Usuario: Calificar Actividad

- **Como** un profesor,
- **Quiero** calificar las actividades completadas por mis estudiantes,
- **Para** proporcionar retroalimentación y evaluar su desempeño.
- **Criterios de Aceptación:**
 - El sistema debe permitir al profesor seleccionar una actividad completada por un estudiante.
 - El sistema debe permitir al profesor ingresar una calificación y un comentario.
 - El sistema debe mostrar un mensaje de confirmación al calificar la actividad.