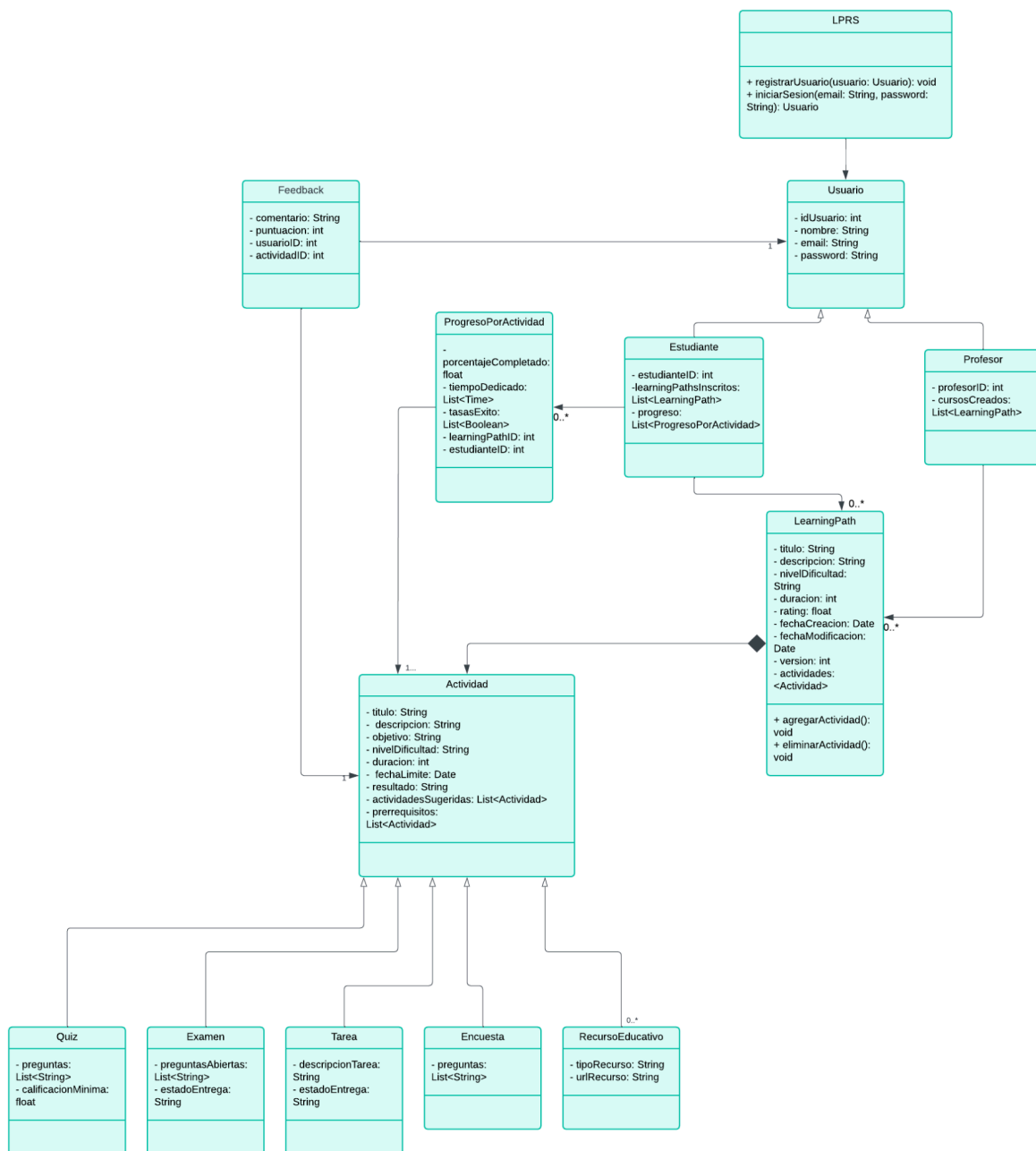


**Diseño y Programación Orientada a Objetos**  
**William Andrés Gómez Muñoz / 202210892**  
**Miguel Rey / 202112736**

**Proyecto 1**  
**Entrega 1**

**1. UML**



## 2. Análisis

### a. Entidades del Caso:

1. LPRS: Esta es la entidad principal que representa el sistema en su totalidad. Es responsable de gestionar las funcionalidades relacionadas con el registro y autenticación de usuarios y el inicio de la sesión de los mismos.
2. Usuario: Es la entidad que representa a los individuos que utilizan el sistema, y puede dividirse en dos subtipos: Profesor y Estudiante. El Profesor tiene la capacidad de crear y gestionar Learning Paths, mientras que el Estudiante sigue los Learning Paths y completa las actividades recomendadas.
3. Learning Path: La entidad learning path almacena una lista de actividades según su tipo. Puede ser creada y editada por profesores, y los estudiantes pueden inscribirse para acceder a información como el título, la descripción, y el nivel de dificultad.
4. Actividad: cada actividad pertenece a un learning path y posee datos legibles por los estudiantes y modificables por los profesores, esta entidad es padre de los distintos tipos de actividades que pueden haber y muestran según la actividad las características de esta misma.
5. Feedback: esta entidad contiene comentarios y puntuación de cada actividad por parte de un usuario hacia esta misma por medio de un id.
6. Progreso: esta entidad va relacionada con cada estudiante y las actividades que va realizando, la entidad lleva un registro del tiempo y el porcentaje completado de cada actividad.

### b. Atributos de las Entidades:

Cada una de estas entidades identificadas tiene atributos específicos que definen su comportamiento y estado dentro del sistema:

1. **LRS:** En este no posee atributos explícitos, pero tiene métodos importantes que permiten la interacción entre las entidades, como registrarUsuario e iniciarSesion.
2. **Usuario:**
  - ❖ idUsuario: identificador único de cada usuario.
  - ❖ nombre: Nombre del usuario.
  - ❖ email: Dirección de correo electrónico del usuario.

- ❖ password: contraseña para autenticación.

Subclases:

- Profesor: incluye atributos adicionales como profesorID y cursosCreados[] (lista de Learning Paths creados).
- Estudiante: Incluye estudianteID, learningPathsInscritos[] (lista de Learning Paths en los que está inscrito) y progreso[] (lista de los progresos registrados en cada Learning Path).

### 3. Learning Path:

- ❖ titulo: Título descriptivo del Learning Path.
- ❖ descripcion: Descripción general del Learning Path.
- ❖ nivelDificultad: Nivel de dificultad del learning path
- ❖ duracion: Duración estimada del Learning Path en minutos.
- ❖ rating: Valoración promedio del Learning Path basada en los comentarios de los usuarios.
- ❖ fechaCreacion: Fecha en que se creó el Learning Path.
- ❖ fechaModificacion: Fecha de la última modificación del Learning Path.
- ❖ version: Versión actual del Learning Path.
- ❖ actividades[]: Lista de actividades que componen el Learning Path.

### 4. Actividad:

- ❖ descripcion: descripción de la actividad.
- ❖ objetivo: Objetivo específico que se busca alcanzar con la actividad.
- ❖ nivelDificultad: Dificultad de la actividad.
- ❖ duracion: tiempo estimado para completar la actividad.
- ❖ fechaLimite: Fecha límite sugerida para la actividad.
- ❖ estado: Estado actual de la actividad.

Subclases:

- RecursoEducativo: Atributos adicionales como tipoRecurso y urlRecurso.
- Tarea: Incluye descripcionTarea y estadoEntrega.
- Quiz: incluye preguntas[] (lista de preguntas) y calificacionMinima
- Examen: similar a Quiz, pero con preguntasAbiertas[] en lugar de preguntas cerradas.
- Encuesta: Atributo clave es preguntas[], que contiene una lista de preguntas abiertas.

### 5. Feedback:

- ❖ comentario: Texto con la opinión del usuario sobre la actividad.

- ❖ puntuacion: Valor numérico que representa la calificación de la actividad.
- ❖ usuarioID: Identificador del usuario que dejó el feedback.
- ❖ actividadID: Identificador de la actividad a la que se refiere el feedback.

#### 6. Progreso:

- ❖ porcentajeCompletado: Porcentaje de actividades obligatorias completadas dentro de un Learning Path.
- ❖ tiempoDedicado[]: Tiempo que el estudiante ha dedicado a cada actividad.
- ❖ tasasExito: Registro de éxito o fracaso en las actividades.
- ❖ learningPathID: Identificador del Learning Path al que pertenece este progreso.
- ❖ estudianteID: Identificador del estudiante que está realizando el Learning Path.

#### 7. Login:

- ❖ **email**: Dirección de correo electrónico utilizada para autenticarse.
- ❖ **password**: Contraseña asociada a la cuenta de usuario.

#### c. Restricciones:

El proyecto tiene varias restricciones que deben tomarse en cuenta durante su desarrollo. Una de las restricciones principales es que todos los datos del sistema, desde los usuarios hasta los Learning Paths y actividades, deben guardarse de forma persistente en archivos. Estos archivos pueden ser en formato plano o binario, pero no deben almacenarse en la misma carpeta donde se encuentra el código fuente. Solo la aplicación tendrá acceso a leer y escribir en estos archivos, por lo que no se permitirá modificaciones externas, garantizando la integridad de los datos.

Otra restricción importante es el sistema de autenticación. Todos los usuarios deben registrarse y autenticarse usando un correo electrónico y una contraseña. El sistema tiene que garantizar que solo usuarios registrados puedan acceder, y debe manejar la información de manera segura para proteger la privacidad. Además, los diferentes tipos de usuarios, como profesores y estudiantes, tienen diferentes permisos y accesos dentro del sistema, lo que significa que el sistema debe ser capaz de identificar claramente qué tipo de usuario está accediendo para asignar los privilegios correctos.

Por último, el proyecto debe ser desarrollado completamente en Java, lo cual significa que no se permitirá el uso de otros lenguajes para la implementación de la lógica del sistema. A pesar de que en esta etapa no es necesario construir una interfaz gráfica para los usuarios, la lógica de todas las funcionalidades debe estar implementada y probada. Se espera que se proporcionen programas de prueba que permitan demostrar el funcionamiento de las

principales características del sistema a través de la consola, asegurando que se cumplen con todos los requerimientos funcionales esperados.

#### d. Demostración de Cada Uno de los Programas de Prueba

### **1. Prueba de registro y autenticación de usuarios**

**Descripción:** Para validar que el sistema puede registrar nuevos usuarios y autenticarlos correctamente, se probará el registro de dos tipos de usuarios: un profesor y un estudiante. Luego, se intentará iniciar sesión con estos usuarios para verificar que el proceso de autenticación funciona correctamente.

#### **Datos de prueba:**

- Profesor: nombre = "William Gómez", email = "wgomez@uniandes.edu", password = "prof123"
- Estudiante: nombre = "Miguel Rey", email = "mrey@uniandes.edu", password = "est123"

#### **Pasos:**

1. Registrar al profesor y al estudiante utilizando los datos proporcionados.
2. Intentar iniciar sesión con ambos usuarios utilizando su email y contraseña.
3. Verificar que, después de la autenticación, el sistema los reconozca como un profesor y un estudiante.

#### **Resultado esperado:**

- El sistema debe registrar ambos usuarios y permitirles iniciar sesión con éxito.
- Después de iniciar sesión, el profesor debería poder acceder a las funcionalidades de creación de learning paths, mientras que el estudiante debería poder inscribirse en learning paths existentes.

### **2. Prueba de creación de un Learning Path**

**Descripción:** Esta prueba validará la funcionalidad de creación de learning paths por parte de un profesor. Se comprobará que el profesor puede crear un learning path y que este se guarda correctamente, con todas las actividades asociadas.

#### **Datos de prueba:**

- Profesor: William Gómez (ya registrada en la prueba anterior)
- Learning path: titulo = "Aprender java basico", descripcion = "Un curso introductorio a la programación en Java", nivelDificultad = "Básico", duracion = 120 minutos

#### **Pasos:**

1. Iniciar sesión como el profesor William Gómez.
2. Crear un Learning Path con los datos especificados.
3. Consultar los Learning Paths creados por William Gómez para verificar que el nuevo Learning Path aparece correctamente.

**Resultado esperado:**

- El sistema debería permitir la creación del learning path y asociarlo al profesor que lo creó.
- Al consultar los Learning Paths del profesor, el learning path "Aprender java basico" debe aparecer con todos sus detalles.

### **3. Prueba de agregar actividades a un Learning Path**

**Descripción:** Esta prueba verificara que las actividades pueden ser añadidas a un learning pat y que cada actividad se guarda con sus atributo. Se utilizarán diferentes tipos de actividades (Recurso Educativo, Tarea, Quiz) para garantizar la flexibilidad del sistema.

**Datos de prueba:**

- Learning Path: Aprender Java Básico (creado en la prueba anterior)
- Actividades:
  - Recurso Educativo: descripcion = "Video introductorio sobre sintaxis Java", tipoRecurso = "Video", urlRecurso = "https://example.com/java-intro"
  - Tarea: descripcion = "Ejercicio de programación: Crear un Hola Mundo en Java", estadoEntrega = "Pendiente"
  - Quiz: preguntas = ["¿Qué es una variable en Java?", "¿Cómo se declara una clase en Java?"], calificacionMinima = 70%

**Pasos:**

1. Iniciar sesión como William Gómez.
2. Agregar las actividades al learning path "Aprender java basico" utilizando los datos proporcionados.
3. Consultar el learning path y verificar que las actividades agregadas aparecen correctamente asociadas.

**Resultado esperado:**

- El sistema debe agregar cada actividad al learning path y permitir su consulta posterior.
- Al visualizar el learning path "Aprender Java basico", deben aparecer las tres actividades, cada una con su tipo y detalles.

### **4. Prueba de inscripción de un estudiante en un Learning Path**

**Descripción:** Esta prueba validara la capacidad del sistema para inscribir a un estudiante en un learning path. Se asegurará del estudiante puede visualizar las actividades y comenzar a realizarlas.

**Datos de prueba:**

- Estudiante: Miguel Rey (registrado en la prueba anterior)
- Learning path: Aprender Java basico

**Pasos:**

1. Iniciar sesión como Miguel Rey.
2. Inscribirse en el learning path "Aprender java básico".
3. Consultar el progreso del estudiante en el learning path para verificar que está inscrito correctamente.
4. Visualizar las actividades del learning path para asegurarse de que están accesibles para el estudiante.

**Resultado esperado:**

- El sistema debe permitir al estudiante inscribirse en el learning path.
- El progreso inicial del estudiante debe mostrarse como 0%, y debe poder acceder a las actividades del learning path.

## **5. Prueba de rastreo del progreso de un estudiante**

**Descripción:** Esta prueba validará que el sistema puede rastrear el progreso del estudiante en un Learning path. Se comprobara que el porcentaje completado se actualiza correctamente al completar actividades obligatorias.

**Datos de prueba:**

- Estudiante: Miguel Rey
- Learning path: Aprender java básico
- Actividad completada: Video introductorio sobre sintaxis Java

**Pasos:**

1. Iniciar sesión como Miguel Rey.
2. Comenzar y completar la primera actividad del learning path ("Video introductorio sobre sintaxis Java").
3. Consultar el progreso del estudiante en el learning path para verificar que el porcentaje completado se ha actualizado.

**Resultado esperado:**

- El progreso del estudiante debe incrementarse tras la finalización de la actividad obligatoria. Por ejemplo, si la actividad completada representa el 20% del Learning Path, el progreso debe actualizarse al 20%.

## **6. Prueba de retroalimentación y rating de actividades**

**Descripción:** Esta prueba validará que tanto estudiantes como profesores pueden dejar comentarios y calificaciones en actividades, y que estos reflejan correctamente en el sistema.

### **Datos de prueba:**

- Estudiante: Miguel Rey
- Actividad: Video introductorio sobre sintaxis Java
- Comentario: "Muy útil para entender los conceptos básicos de Java"
- Puntuación: 5 estrellas

### **Pasos:**

1. Iniciar sesión como Miguel Rey.
2. Completar la actividad "Video introductorio sobre sintaxis Java".
3. Dejar un comentario y una puntuación sobre la actividad.
4. Consultar la actividad para verificar que el comentario y la puntuación han sido guardados correctamente.

### **Resultado esperado:**

- El sistema debe registrar el comentario y la puntuación dejados por el estudiante.
- Al consultar la actividad, deben aparecer la reseña y el rating promedio actualizados.