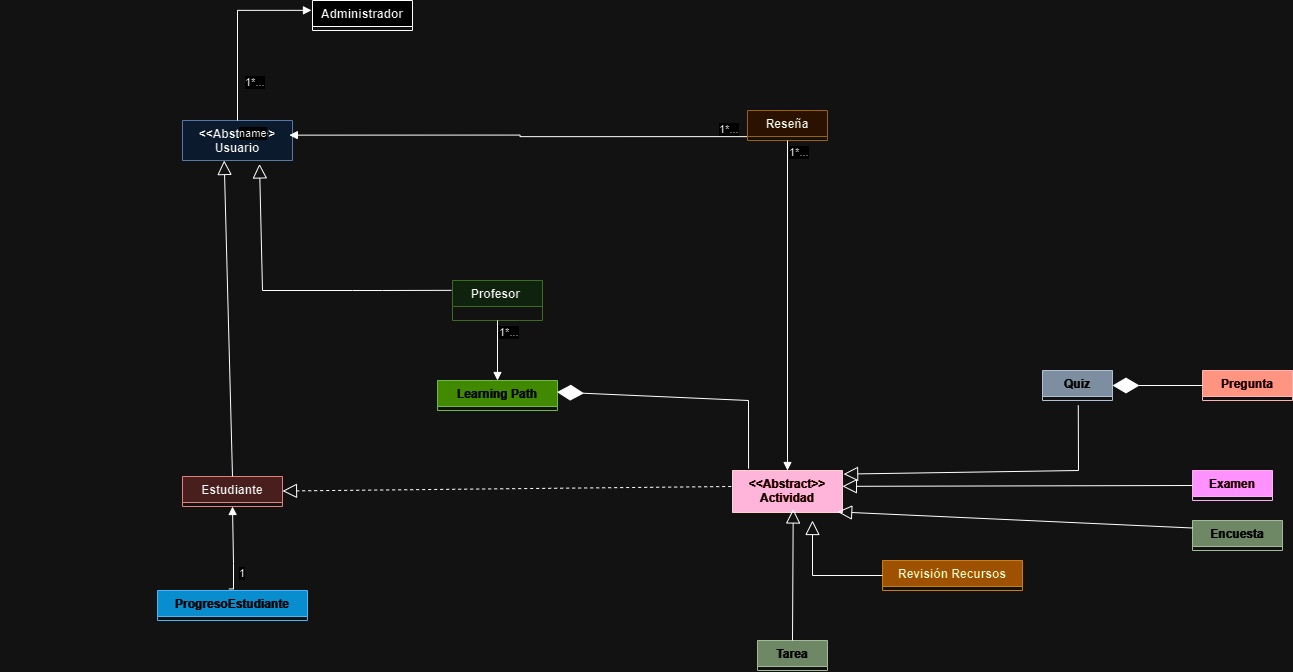
Esteban Garrido - 202021158

Alejandro Enrique Vides Vivero - 202221712

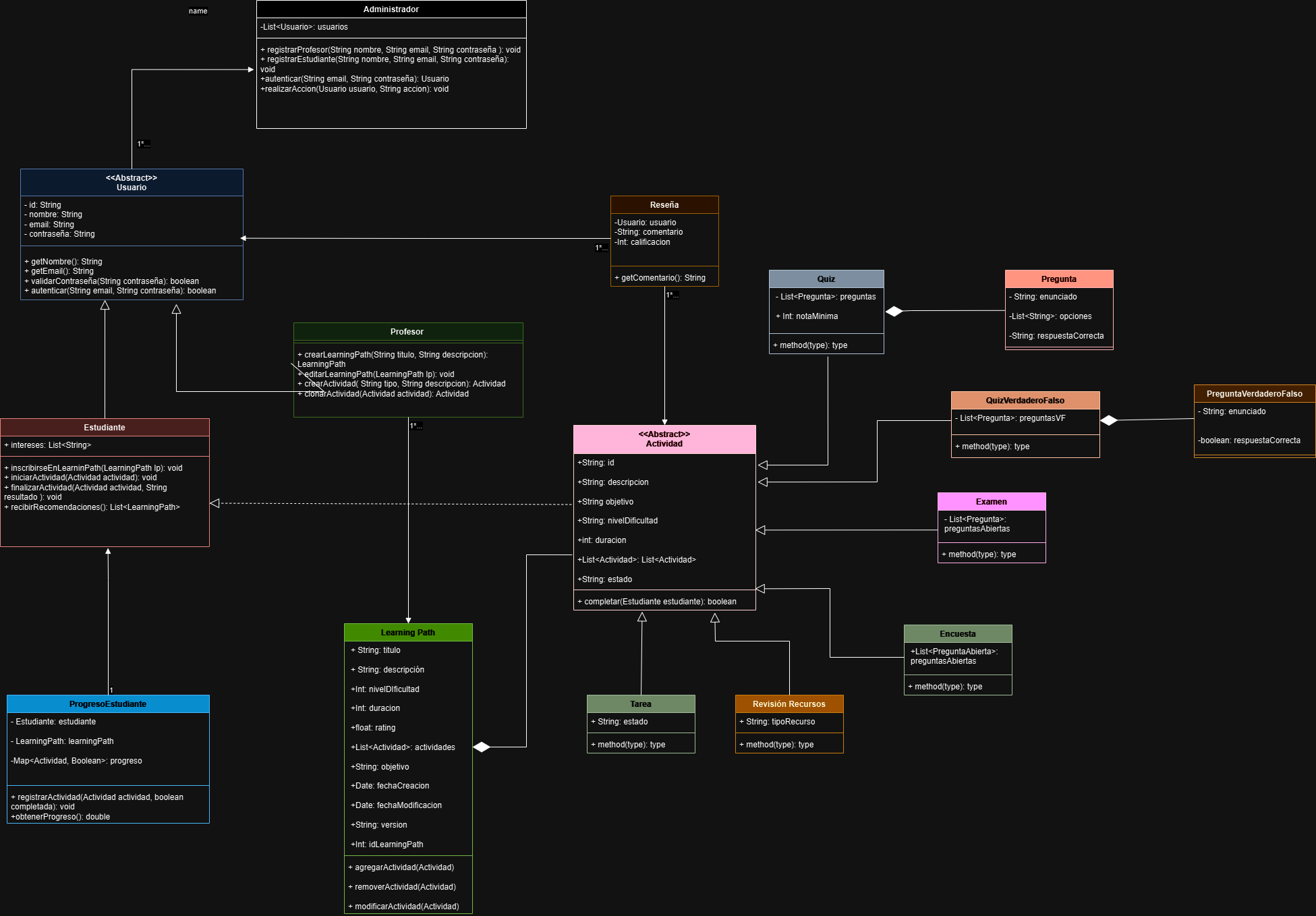
Marilyn Viviana Sánchez Páez - 202114050

**Documento de Diseño - Learning Path Recommendation System**

**Diagrama de diseño:**



**Diagrama de diseño (Detallado)**



**Responsabilidades:**

1. Clase Administrador:

**Responsabilidad**: Gestión de usuarios en el sistema. El administrador puede registrar y autenticar a profesores y estudiantes. Además, tiene la capacidad de realizar acciones sobre usuarios, como gestionar permisos o roles.

1. Clase Usuario (Abstracta):

**Responsabilidad**: Clase base que representa cualquier tipo de usuario en el sistema (como profesores o estudiantes). Define atributos y métodos comunes como el manejo de credenciales (email, contraseña) y la autenticación.

1. Clase Profesor:

**Responsabilidad**: Creación y gestión de rutas de aprendizaje (LearningPath) y actividades relacionadas. Un profesor puede crear, editar y clonar actividades (quiz, tareas, exámenes), lo que le permite estructurar las lecciones y evaluaciones dentro de las rutas de aprendizaje.

1. Clase Estudiante:

**Responsabilidad**: Participación en rutas de aprendizaje. El estudiante se inscribe en rutas de aprendizaje, inicia y finaliza actividades y recibe recomendaciones basadas en sus intereses. Además, tiene un progreso que se puede rastrear a lo largo de las actividades en las que participa.

1. Clase LearningPath:

**Responsabilidad**: Representa un conjunto de actividades y objetivos de aprendizaje que un estudiante puede seguir. Incluye la administración de actividades (como agregar o quitar) y el seguimiento del progreso del estudiante dentro de la ruta.

1. Clase Actividad (Abstracta):

**Responsabilidad:** Clase base para diferentes tipos de actividades (quiz, tareas, exámenes). Cada actividad tiene atributos comunes como la descripción, el objetivo, y el nivel de dificultad, y puede completarse por los estudiantes.

1. Clase Quiz:

**Responsabilidad:** Actividad que incluye una lista de preguntas con un puntaje mínimo. El estudiante debe completar este cuestionario como parte de su ruta de aprendizaje.

1. Clase Examen:

**Responsabilidad**: Un tipo de actividad que incluye preguntas abiertas. A diferencia del quiz, el examen podría requerir respuestas más complejas o abiertas.

1. Clase Encuesta:

**Responsabilidad**: Otra forma de actividad con preguntas abiertas, pero su objetivo es obtener retroalimentación.

1. Clase Tarea:

**Responsabilidad**: Actividad asignada a los estudiantes que incluye tareas específicas con un estado (completado, pendiente, etc.).

1. Clase Revisión de Recursos:

**Responsabilidad:** Actividad donde el estudiante revisa recursos o materiales educativos, como lecturas o videos, para complementar su aprendizaje.

1. Clase Progreso de Estudiante:

**Responsabilidad**: Rastrea el progreso de un estudiante a lo largo de una ruta de aprendizaje. Esta clase incluye información sobre qué actividades se han completado y en qué estado se encuentra el aprendizaje del estudiante.

1. Clase Reseña:

**Responsabilidad**: Permite a los estudiantes o profesores proporcionar comentarios y calificaciones sobre una actividad. Esta clase ayuda a evaluar la calidad y el impacto de una actividad.

1. Clase Pregunta:

**Responsabilidad:** Definir y almacenar el enunciado de una pregunta dentro de los quizzes.

**Justificaciones**

El diseño presentado sigue un enfoque orientado a objetos, asegurando modularidad, flexibilidad, reutilización de código y mantenibilidad. A continuación, detallo las decisiones clave tomadas y sus respectivas justificaciones.

1. División en Clases y Jerarquías

Motivo: La división de funcionalidades entre diferentes clases (como Usuario, Profesor, Estudiante, LearningPath y Actividad) sigue los principios de Single Responsibility y Open/Closed Principle.

Justificación: Cada clase se encarga de una única responsabilidad:

Usuario abstrae las propiedades comunes de profesores y estudiantes.

Profesor y Estudiante tienen comportamientos específicos, como la creación de Learning Paths o la inscripción en ellos, respectivamente.

Las actividades son tratadas como una jerarquía de tipos (RecursoEducativo, Tarea, Quiz, Examen, y Encuesta) para permitir la extensión futura con nuevos tipos sin modificar las clases existentes.

2. Jerarquía mediante Herencia

Motivo: Se usó herencia para evitar duplicación de código y agrupar comportamientos similares.

Justificación:

Usuario como clase base de Profesor y Estudiante, facilita la reutilización del comportamiento común (como autenticación).

Actividad es una superclase de los distintos tipos de actividades, permitiendo definir un comportamiento general para todas las actividades mientras cada subclase agrega detalles específicos.

3. Uso de Composición y Asociación

Motivo: La composición y asociación entre objetos permiten construir relaciones complejas entre entidades sin acoplamiento excesivo.

Justificación:

Composición: Un LearningPath contiene múltiples Actividad (1..\*). Esto refleja que las actividades forman parte integral del camino de aprendizaje.

Asociación: La relación entre Profesor y LearningPath es una asociación, ya que un profesor puede gestionar varios Learning Paths pero no son parte exclusiva de él. Del mismo modo, un estudiante puede inscribirse en múltiples caminos.

Agregación: ProgresoEstudiante agrega las actividades y su estado para un camino de aprendizaje sin poseerlas, lo que refleja un vínculo lógico pero no una dependencia directa.

4. Control de Acceso Mediante Polimorfismo

Motivo: El polimorfismo garantiza que cada tipo de usuario solo pueda realizar las acciones permitidas para su rol.

Justificación:

La clase SistemaUsuarios gestiona los permisos. Por ejemplo, los profesores pueden crear Learning Paths, mientras que los estudiantes no. Esto se logra a través de la separación clara de métodos entre Profesor y Estudiante.

5. Modularidad y Extensibilidad

Motivo: Diseñar un sistema extensible facilita añadir nuevas funcionalidades en el futuro.

Justificación:

La clase Actividad como superclase permite añadir nuevos tipos de actividades (por ejemplo, Proyecto o Práctica) sin afectar el resto del sistema.

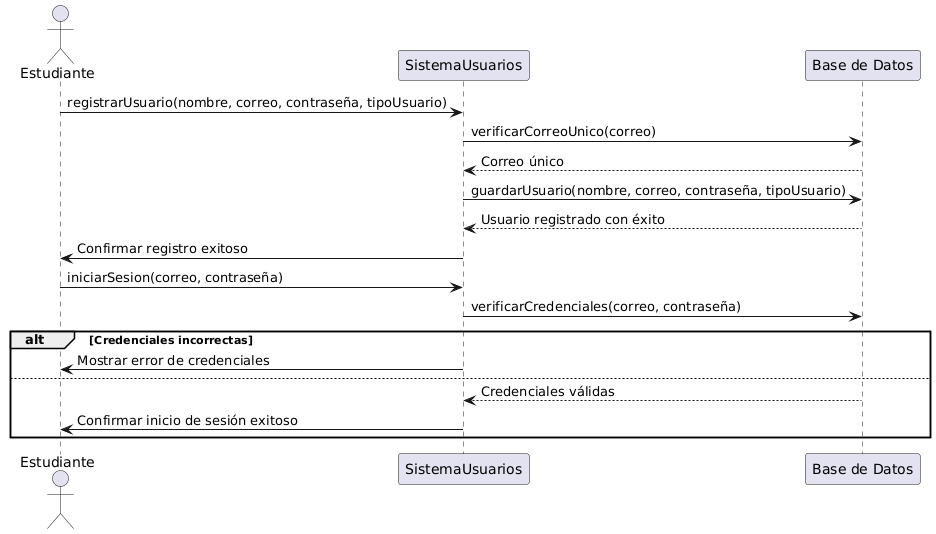
Los profesores y estudiantes pueden interactuar con reseñas y calificaciones, lo que permite mejorar continuamente el contenido

**Persistencia**

El módulo de persistencia maneja el almacenamiento y recuperación de datos para que los usuarios, actividades y progresos en losLearning Paths se mantengan entre sesiones. Para el proyecto usaremos el formato JSON. Esto es porque ofrece varias ventajas, como la legibilidad de los datos, que facilita el proceso de debugging, y la portabilidad, que permite compartir datos con otras aplicaciones que también usen JSON. Además, JSON facilita la extensión de datos, permitiendo añadir propiedades a los objetos sin necesidad de modificar la estructura de almacenamiento subyacente. Por lo tanto, el uso de JSON en la persistencia aporta claridad y flexibilidad, siendo una excelente alternativa para facilitar la interacción y mantenimiento del sistema.

**Diagramas de secuencia:**

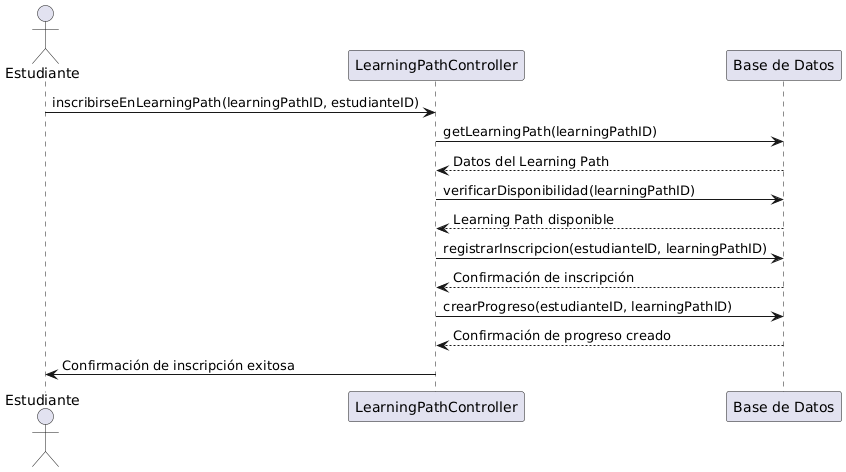
Requerimiento 1: Registro e inicio de sesión



### Descripción del Diagrama:

1. Registro del Estudiante:
   1. El estudiante solicita registrarse proporcionando su nombre, correo electrónico y contraseña.
   2. El SistemaUsuarios verifica que el correo sea único en la base de datos y, si lo es, registra al nuevo usuario.
2. Inicio de Sesión:
   1. El estudiante intenta iniciar sesión proporcionando sus credenciales (correo y contraseña).
   2. El sistema verifica las credenciales en la base de datos. Si son correctas, el inicio de sesión es exitoso; si no lo son, se muestra un error.

Requerimiento 2: inscripción en un learning path



### Descripción del Diagrama:

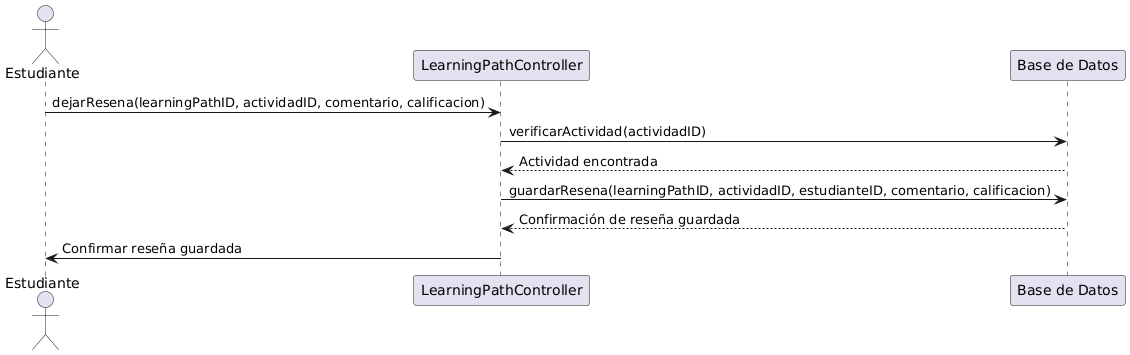
1. Solicitud de Inscripción:
   1. El estudiante solicita inscribirse en un Learning Path específico mediante el LearningPathController.
2. Verificación del Learning Path:
   1. El LearningPathController consulta la base de datos para obtener la información del Learning Path y verificar su disponibilidad.
3. Registro de Inscripción:
   1. Si el Learning Path está disponible, se registra la inscripción del estudiante en la base de datos.
4. Generación del Progreso:
   1. Después de la inscripción, se genera un registro de progreso inicial para el estudiante en el Learning Path.
5. Confirmación al Estudiante:
   1. Finalmente, el sistema confirma al estudiante que la inscripción fue exitosa.

Requerimiento 3: Completitud de actividades.

### Descripción del Diagrama:

1. Solicitud de Completitud:
   1. El estudiante solicita completar una actividad en el LearningPathController.
2. Verificación de Prerrequisitos:
   1. El LearningPathController delega la verificación de prerrequisitos al ProgresoController.
   2. Si los prerrequisitos no están cumplidos, se muestra una advertencia.
   3. Si los prerrequisitos están cumplidos, el sistema procede a marcar la actividad como completada.
3. Actualización del Progreso:
   1. Una vez que la actividad ha sido completada, se actualiza el progreso del estudiante en el Learning Path, y se guarda en la base de datos.
4. Confirmación:
   1. El sistema confirma al estudiante que la actividad ha sido completada exitosamente.

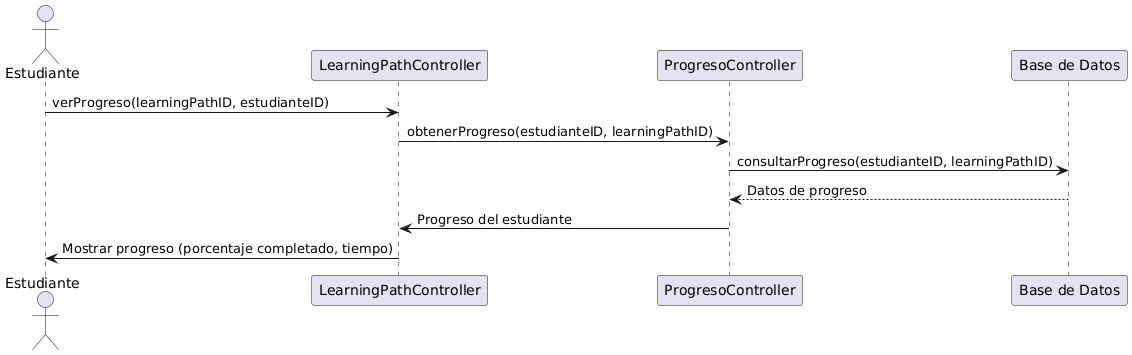
Requerimiento 4: Reseñas



### Descripción del Diagrama:

1. Estudiante deja una reseña:
   1. El estudiante llama a la función dejarResena(), pasando el ID del Learning Path, el ID de la actividad, su comentario y la calificación.
2. Verificación de la Actividad:
   1. El LearningPathController verifica si la actividad existe en la base de datos.
3. Guardar la reseña:
   1. Si la actividad es válida, el LearningPathController guarda la reseña en la base de datos, asociando la reseña con el Learning Path, la actividad, y el ID del estudiante.
4. Confirmación:
   1. Finalmente, el sistema confirma que la reseña ha sido guardada exitosamente.

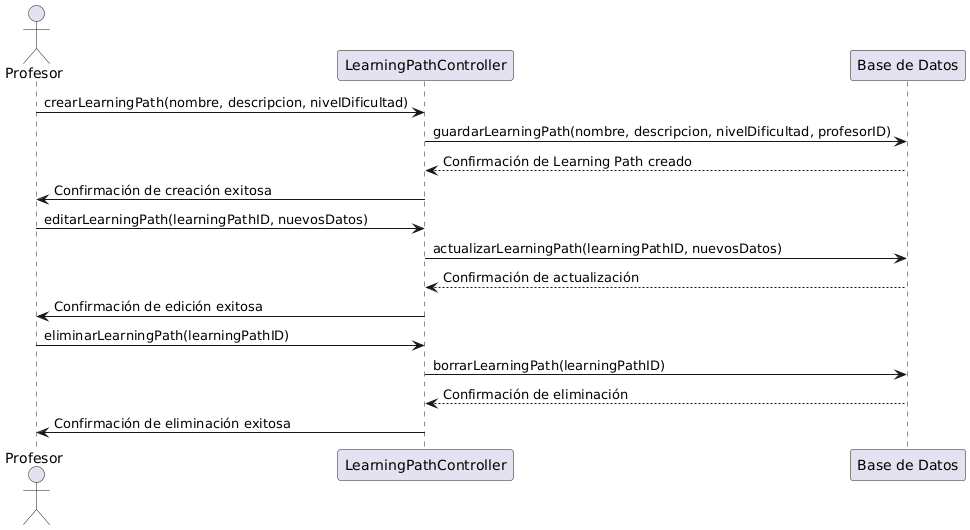
Requerimiento 5: revisión de progreso



### Descripción del Diagrama:

1. Solicitud del Estudiante:
   1. El estudiante llama al método verProgreso(), pasando el ID del Learning Path y su ID.
2. Obtención de Progreso:
   1. El LearningPathController delega la consulta del progreso al ProgresoController.
3. Consulta a la Base de Datos:
   1. El ProgresoController consulta la base de datos para obtener el progreso del estudiante en el Learning Path.
4. Devolución de Progreso:
   1. Los datos del progreso (porcentaje completado, tiempo dedicado, etc.) se devuelven al LearningPathController y luego al estudiante.

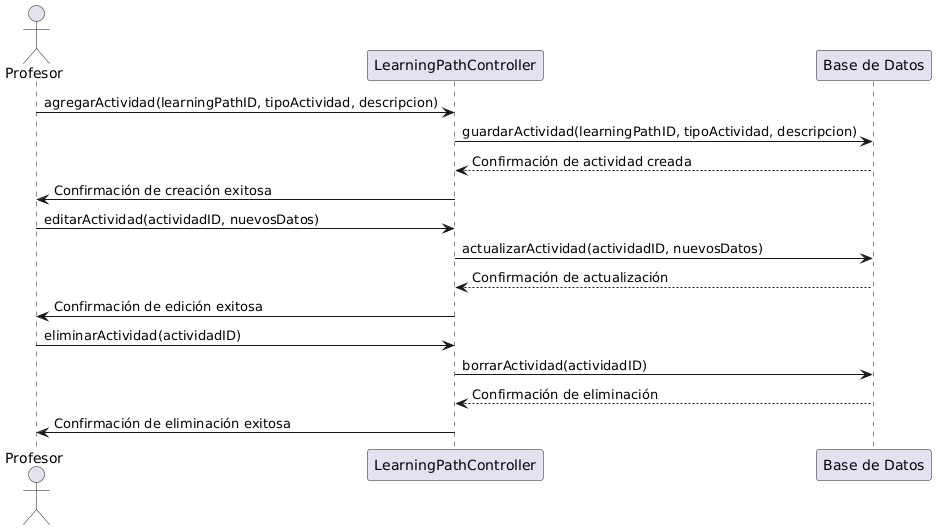
Requerimiento 6: Gestion de learning path



### Descripción del Diagrama:

1. Creación del Learning Path:
   1. El profesor solicita la creación de un nuevo Learning Path.
   2. El LearningPathController guarda el Learning Path en la base de datos y confirma al profesor.
2. Edición del Learning Path:
   1. El profesor edita los detalles de un Learning Path existente.
   2. El controlador actualiza la información en la base de datos y confirma la actualización.
3. Eliminación del Learning Path:
   1. El profesor solicita eliminar un Learning Path.
   2. El LearningPathController borra el Learning Path de la base de datos y confirma la eliminación.

Requerimiento 7: Gestión de actividades



### Descripción del Diagrama:

1. Agregar una Actividad:
   1. El profesor solicita agregar una nueva actividad a un Learning Path.
   2. El LearningPathController guarda la actividad en la base de datos y confirma la creación exitosa al profesor.
2. Editar una Actividad:
   1. El profesor edita los detalles de una actividad existente.
   2. El controlador actualiza la actividad en la base de datos y confirma la edición.
3. Eliminar una Actividad:
   1. El profesor solicita eliminar una actividad.
   2. El LearningPathController borra la actividad de la base de datos y confirma la eliminación.

Requerimiento 8: Calificación

### Descripción del Diagrama:

1. Solicitud de Calificación:
   1. El profesor llama al método calificarActividad(), proporcionando el ID de la actividad, el ID del estudiante, y la calificación.
2. Obtención de Datos de la Actividad:
   1. El LearningPathController obtiene los detalles de la actividad de la base de datos.
3. Actualización de la Calificación:
   1. La calificación es actualizada en la base de datos asociando la calificación con el estudiante y la actividad específica.
4. Confirmación:
   1. El sistema confirma al profesor que la calificación ha sido guardada exitosamente.

Requerimiento 9: Acceso a estadísticas.

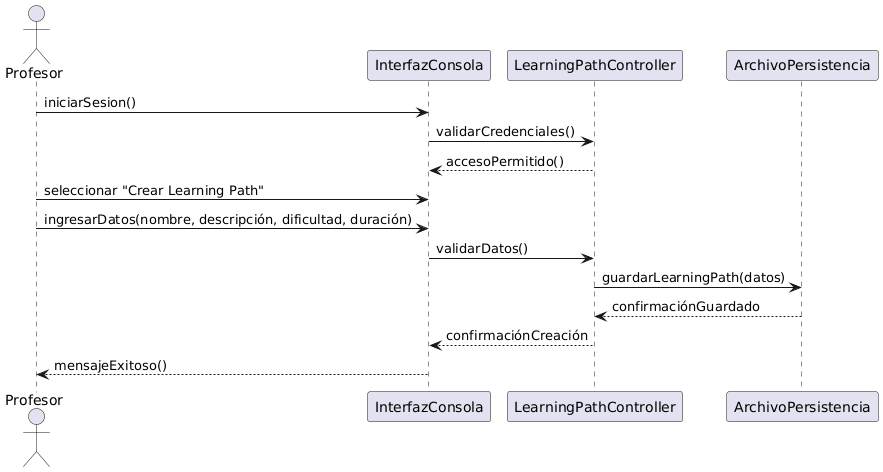
### Descripción del Diagrama:

1. Solicitud de Estadísticas:
   1. El profesor solicita ver las estadísticas de un Learning Path utilizando el método verEstadisticas().
2. Consulta a la Base de Datos:
   1. El LearningPathController consulta la base de datos para obtener las estadísticas de los estudiantes en el Learning Path.
3. Datos Devueltos:
   1. La base de datos devuelve las estadísticas, incluyendo el progreso y el rendimiento de los estudiantes.
4. Presentación de las Estadísticas:
   1. El LearningPathController muestra las estadísticas solicitadas al profesor.

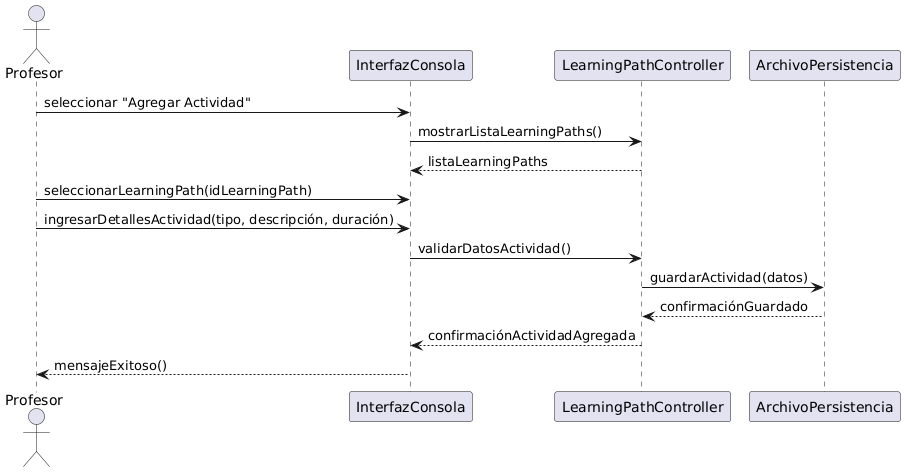
**Historias de Usuario**

### **Rol: Profesor**

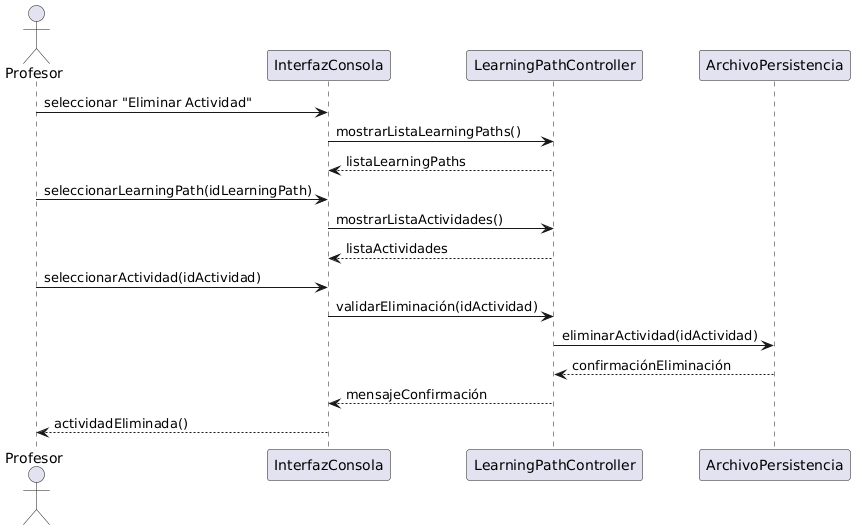
1. **Crear un Learning Path**
   1. **Historia**: Como profesor, quiero crear un Learning Path con actividades para estructurar el aprendizaje de mis estudiantes.
   2. **Entradas**:
      1. Nombre del Learning Path.
      2. Descripción.
      3. Nivel de dificultad.
      4. Duración estimada.
   3. **Acciones**:
      1. Iniciar sesión en el sistema.
      2. Seleccionar la opción "Crear Learning Path".
      3. Proporcionar los detalles y guardar el Learning Path.
      4. Confirmar la creación.
   4. **Validaciones y restricciones:**
      1. Todos los campos son obligatorios.
      2. No se pueden duplicar nombres de Learning Paths.
   5. **Salidas**:
      1. Confirmación de que el Learning Path fue creado exitosamente.
      2. Mensaje de error si faltan datos o si ya existe un Learning Path con el mismo nombre.



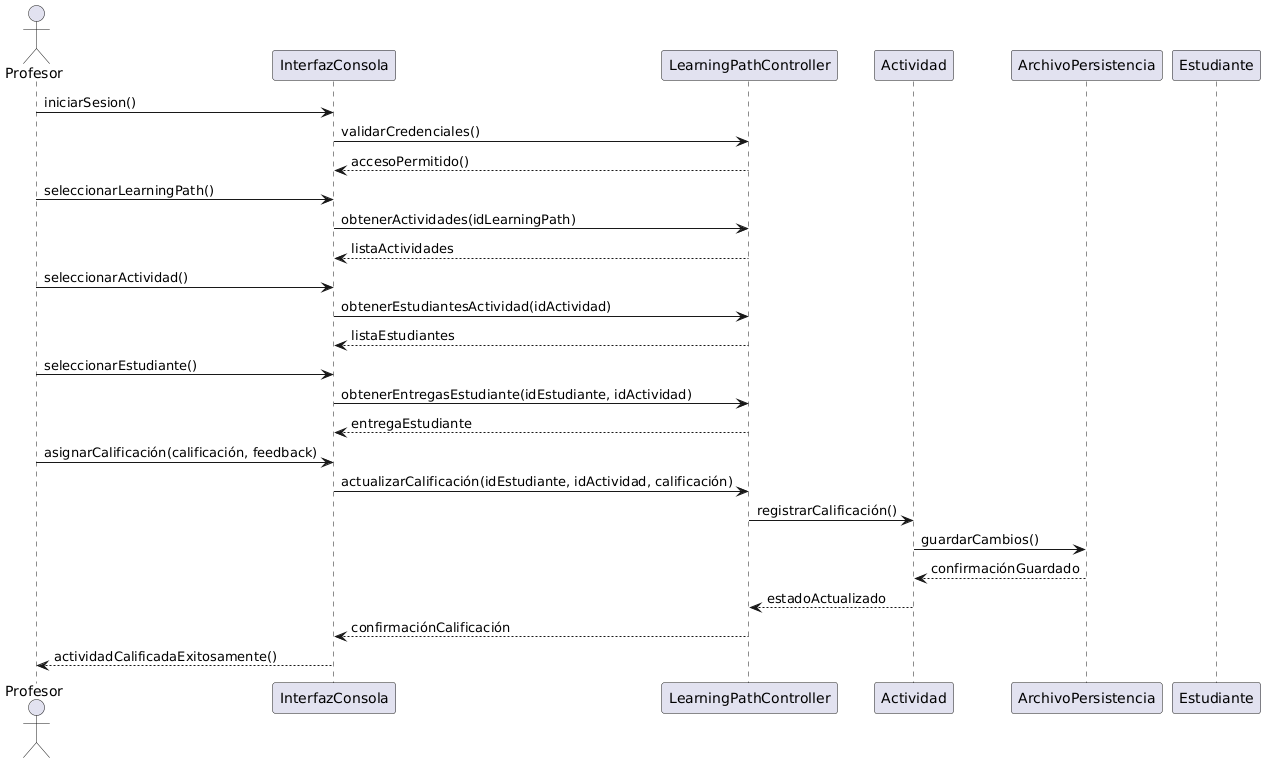
1. **Agregar una actividad al Learning Path**
   1. **Historia**: Como profesor, quiero agregar actividades como quices, tareas o exámenes a un Learning Path para enriquecer el contenido.
   2. **Entradas**:
      1. Selección del Learning Path.
      2. Tipo de actividad (quiz, tarea, examen, etc.).
      3. Detalles de la actividad (descripción, objetivo, nivel de dificultad, duración, etc.).
   3. **Acciones**:
      1. Seleccionar un Learning Path existente.
      2. Configurar la nueva actividad (llenar los detalles).
      3. Confirmar y guardar la actividad.
   4. **Validaciones y restricciones:**
      1. Las actividades deben tener un tipo válido.
      2. Las actividades obligatorias deben estar bien configuradas con un objetivo claro.
   5. **Salidas**:
      1. Confirmación de que la actividad fue agregada al Learning Path.
      2. Visualización de la actividad en la lista del Learning Path.
      3. Mensaje de error si falta algún dato.



1. **Eliminar una actividad del Learning Path**
   1. **Historia**: Como profesor, quiero eliminar una actividad que ya no es relevante para mantener actualizado el contenido.
   2. **Entradas**:
      1. Selección del Learning Path.
      2. Selección de la actividad a eliminar.
   3. **Acciones**:
      1. Seleccionar el Learning Path.
      2. Seleccionar la actividad específica.
      3. Confirmar la eliminación de la actividad.
   4. Validaciones y restricciones:
      1. Confirmación adicional para evitar eliminaciones accidentales.
      2. Verificar que la actividad no sea un prerrequisito de otra actividade antes de eliminarla.
   5. **Salidas**:
      1. Mensaje de éxito indicando que la actividad fue eliminada.
      2. Mensaje de error si la actividad no puede eliminarse debido a dependencias.

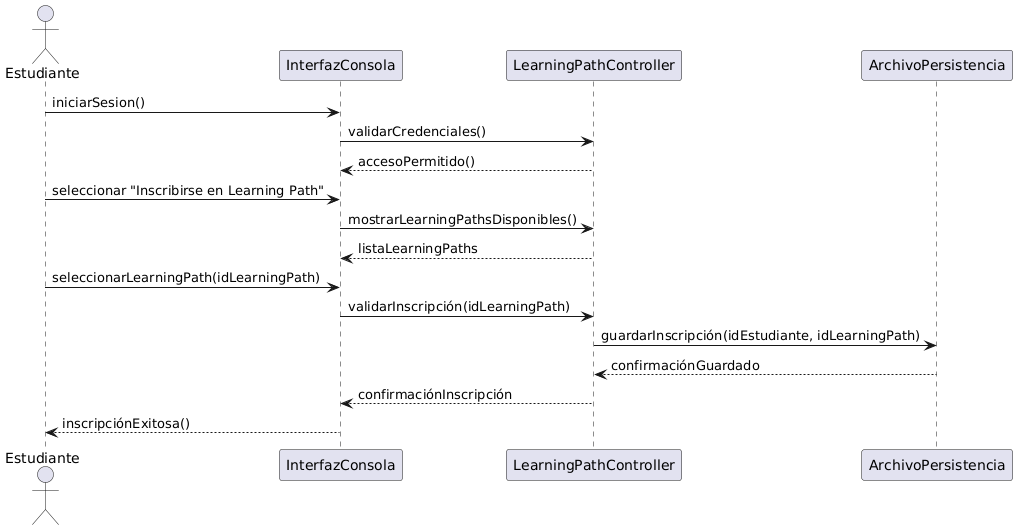


1. **Calificar una actividad.**
   1. **Historia:** Como profesor, quiero calificar las actividades realizadas por los estudiantes para darles retroalimentación y actualizar su progreso en el Learning Path.
   2. **Entradas:**
      1. Selección del Learning Path.
      2. Selección de la actividad a calificar.
      3. Selección del estudiante.
      4. Respuestas del estudiante o entregas realizadas.
      5. Calificación asignada (nota o estado: aprobado/no aprobado).
   3. **Acciones:**
      1. Iniciar sesión en el sistema.
      2. Seleccionar un Learning Path de la lista de los creados por el profesor.
      3. Seleccionar la actividad a calificar dentro del Learning Path.
      4. Elegir al estudiante que realizó la actividad.
      5. Revisar las respuestas o entregas del estudiante.
      6. Asignar una calificación o marcar la actividad como completada.
      7. Guardar los cambios.
   4. **Validaciones y restricciones:**
      1. El profesor solo puede calificar actividades de Learning Paths que él creó.
      2. Validar que el estudiante haya enviado una respuesta o entrega antes de permitir la calificación.
      3. Si la calificación es numérica, debe estar dentro del rango permitido (por ejemplo, de 0 a 100).
   5. **Salidas:**
      1. Confirmación de que la calificación fue registrada exitosamente.
      2. Actualización del estado de la actividad como "completada" o "reprobada".
      3. Actualización del progreso del estudiante en el Learning Path.
      4. Mensaje de error si no hay entrega o si ya se calificó la actividad previamente.

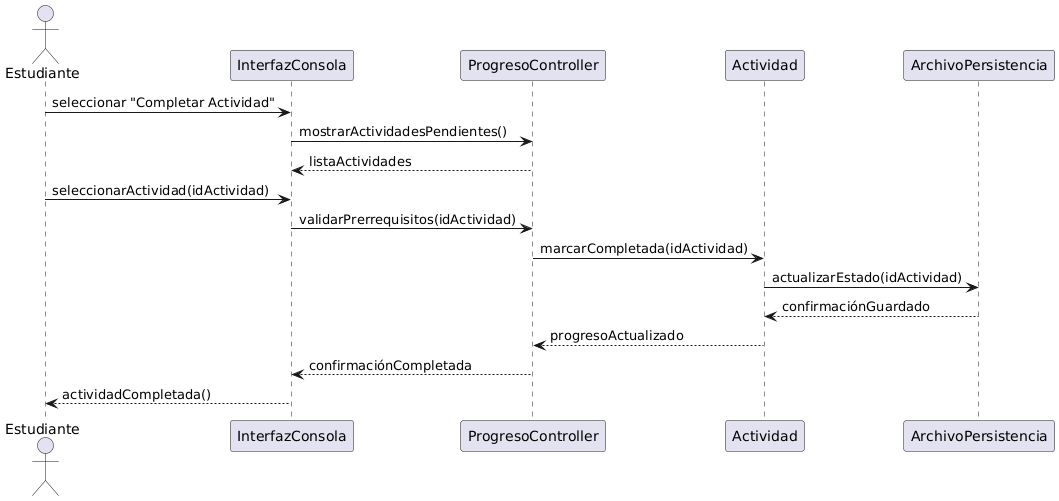


### **Rol: Estudiante**

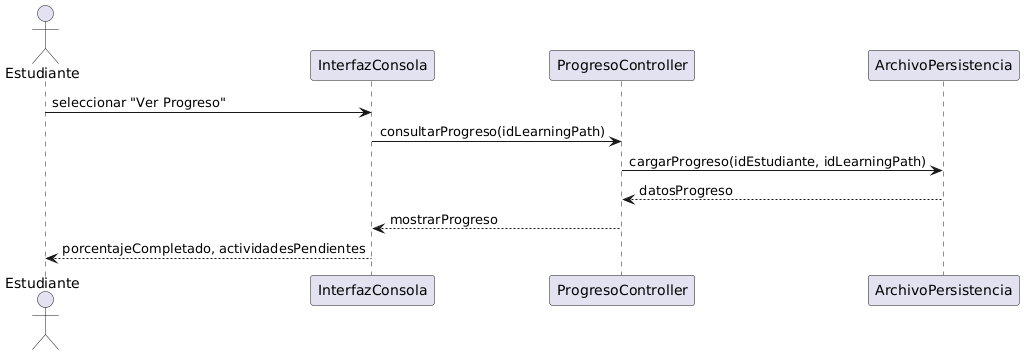
1. **Inscribirse en un Learning Path**
   1. **Historia**: Como estudiante, quiero inscribirme en un Learning Path para seguir su contenido y mejorar mis conocimientos.
   2. **Entradas**:
      1. Selección del Learning Path disponible.
   3. **Acciones**:
      1. Iniciar sesión en el sistema.
      2. Navegar por los Learning Paths disponibles.
      3. Seleccionar uno e inscribirse.
   4. **Validaciones y restricciones:**
      1. No puede inscribirse si ya está registrado en el Learning Path.
      2. No puede inscribirse en Learning Paths completados.
   5. **Salidas**:
      1. Confirmación de inscripción.
      2. Visualización de las actividades del Learning Path.
      3. Mensaje de error si ya está inscrito o no cumple los criterios.



1. **Completar una actividad**
   1. **Historia**: Como estudiante, quiero completar una actividad dentro del Learning Path para avanzar en mi progreso.
   2. **Entradas**:
      1. Selección de una actividad del Learning Path.
      2. Respuestas (en caso de quices o exámenes).
   3. **Acciones**:
      1. Seleccionar una actividad pendiente.
      2. Completar los requisitos de la actividad (como enviar respuestas).
      3. Marcar la actividad como completada.
   4. **Validaciones y restricciones:**
      1. Cumplir los prerrequisitos de la actividad.
      2. Validar el formato de las respuestas.
      3. Respetar los límites de tiempo para actividades con duración establecida.
   5. **Salidas**:
      1. Confirmación de actividad completada.
      2. Actualización del progreso general del Learning Path.
      3. Mensaje de error si no se cumplen los prerrequisitos.



1. **Ver progreso en un Learning Path**
   1. **Historia**: Como estudiante, quiero ver mi progreso en un Learning Path para conocer cuánto me falta por completar.
   2. **Entradas**:
      1. Selección del Learning Path.
   3. **Acciones**:
      1. Acceder a la sección de progreso.
      2. Consultar el estado del Learning Path.
   4. **Validaciones y restricciones.**
      1. Solo se muestra progreso de Learning Paths en los que está inscrito.
   5. **Salidas**:
      1. Porcentaje de progreso.
      2. Lista de actividades completadas y pendientes.



**Rol común: Todos los Usuarios**

1. **Registro de Usuarios.**
   1. **Historia**: Como usuario (profesor o estudiante), quiero registrarme en el sistema según mi rol para acceder a las funcionalidades que necesito.
   2. **Entradas**:
      1. Nombre.
      2. Correo electrónico.
      3. Contraseña.
      4. Selección de rol (profesor o estudiante).
   3. **Acciones**:
      1. Ingresar los datos requeridos en el formulario de registro.
      2. Confirmar el registro.
   4. **Validaciones y restricciones**:
      1. Todos los campos son obligatorios.
      2. El correo electrónico debe ser único.
      3. La contraseña debe cumplir criterios de seguridad (longitud mínima, caracteres especiales, etc.).
   5. **Salidas**:
      1. Confirmación del registro exitoso.
      2. Mensaje de error si los datos no son válidos o si el correo ya está registrado.

