**Historias de usuario:**

**Profesor como creador de Learning Path**

1. **Historia 1: Crear un nuevo Learning Path**  
   **Como** profesor,  
   **quiero** poder crear un Learning Path especificando su título, descripción y nivel de dificultad,  
   **para** que los estudiantes tengan una ruta clara para seguir en su aprendizaje.
2. **Historia 2: Agregar actividades al Learning Path**  
   **Como** profesor,  
   **quiero** añadir actividades a un Learning Path existente con detalles como tipo, descripción, duración, nivel de dificultad y dependencias,  
   **para** que las actividades sean organizadas y cumplan con un flujo lógico.
3. **Historia 3: Gestionar actividades existentes**  
   **Como** profesor,  
   **quiero** editar actividades que he creado,  
   **para** mantener actualizado el contenido del Learning Path según sea necesario.
4. **Historia 4: Crear actividades nuevas**  
   **Como** profesor,  
   **quiero** crear actividades nuevas para añadir o no a mis learning paths,  
   **para** crear nuevo contenido para mis necesidades.
5. **Historia 5: Clonar actividades de otros profesores**  
   **Como** profesor,  
   **quiero** clonar actividades de otros Learning Paths,  
   **para** reutilizar contenido relevante y adaptarlo a mis necesidades.
6. **Historia 6: Revisar ratings de las actividades**  
   **Como** profesor,  
   **quiero** ver calificaciones sobre las actividades creadas,  
   **para** mejorar la calidad y relevancia del Learning Path.

**Estudiante**

1. **Historia 1: Registrarse y autenticarse**  
   **Como** estudiante,  
   **quiero** registrarme y acceder al sistema con un login y contraseña,  
   **para** inscribirme en Learning Paths y realizar actividades.
2. **Historia 2: Explorar Learning Paths disponibles**  
   **Como** estudiante,  
   **quiero** ver un listado de Learning Paths con su título, descripción, nivel y duración,  
   **para** elegir el que más me interese.
3. **Historia 3: Inscribirse en un Learning Path**  
   **Como** estudiante,  
   **quiero** inscribirme en un Learning Path,  
   **para** comenzar a seguirlo y completar las actividades.
4. **Historia 4: Realizar actividades**  
   **Como** estudiante,  
   **quiero** completar actividades en el Learning Path, siguiendo las dependencias y recomendaciones,  
   **para** avanzar en mi aprendizaje de manera organizada.
5. **Historia 5: Recibir advertencias sobre prerrequisitos**  
   **Como** estudiante,  
   **quiero** recibir una advertencia al intentar realizar una actividad sin completar sus prerrequisitos,  
   **para** entender las posibles consecuencias de mi decisión.
6. **Historia 6: Dejar feedback en actividades**  
   **Como** estudiante,  
   **quiero** calificar y dejar comentarios en las actividades que realizo,  
   **para** ayudar a mejorar los Learning Paths en el futuro.
7. **Historia 7: Consultar mi progreso**  
   **Como** estudiante,  
   **quiero** ver mi progreso en un Learning Path, incluyendo actividades completadas y porcentaje de avance,  
   **para** evaluar mi desempeño.

**Profesor haciendo seguimiento al progreso de un estudiante en un Learning Path**

1. **Historia 1: Ver estudiantes inscritos en un Learning Path**  
   **Como** profesor,  
   **quiero** ver una lista de estudiantes inscritos en un Learning Path,  
   **para** saber quiénes están participando.
2. **Historia 2: Consultar el progreso de un estudiante**  
   **Como** profesor,  
   **quiero** revisar el progreso de un estudiante en un Learning Path, incluyendo actividades completadas, tiempos y resultados,  
   **para** evaluar su desempeño y detectar áreas de mejora.
3. **Historia 3: Evaluar tareas y exámenes**  
   **Como** profesor,  
   **quiero** revisar y calificar tareas y exámenes enviados por estudiantes,  
   **para** registrar sus resultados en el sistema.

Modificaciones al programa:

Persistencia y entidades:

Las modificaciones a las clases ya existentes relacionadas a persistencia fueron muy pocas, se añadió el id del autor del learning path a la persistencia de los learning paths, lo cual hace que los archivos correspondientes tengan una línea más con este id. Para la parte de revisión de actividades y quizzes, se agregó un apartado correspondiente para guardar la respuesta hecha por el estudiante, haciendo que la respuesta de cada actividad (tal cual como la haya escrito el estudiante) se guarde en un archivo aparte en la carpeta data/estructuras/respuestas, archivos que también se cargan al momento de iniciar la aplicación.

En las clases correspondientes a los distintos tipos de actividades, se añadió un atributo estático final para saber si son un tipo de actividad que requiere respuesta del estudiante, porque, por ejemplo, si es revisar un recurso, no se necesita que el estudiante responda nada. Caso contrario ocurre con quizzes o exámenes, que requieren que el estudiante de una respuesta si o si, respuesta que se guarda tal cual la haya escrito en la consola el estudiante.

Para otras funciones aparte de la persistencia se agregaron los métodos correspondientes que no existían todavía en los gestores, por ejemplo, para revisar la respuesta subida por un estudiante a una actividad (que si requiera respuesta) y este estilo de funciones de conexión que no existían anteriormente.

Consola:

Para la consola se crearon varias clases con distintas funciones, inicialmente son 3 clases abstractas para crear métodos que utilizan varias clases y facilitar el mantenimiento y expansión. Estas clases abstractas son para:

* Manejar el login de un usuario con el gestor
* Manejar el login de un profesor con el gestor (para separar profesor como creador y profesor como revisor)
* Manejar la creación de actividades según el tipo

A partir de esto, se crearon las clases para funciones especificas que heredan de alguna de estas. Además de esto, se crearon clases con utilidades de input y print, utilizadas por todas las clases de la interfaz, estas utilidades incluyen leer enteros, floats, leer una línea o un párrafo, imprimir los detalles de actividades o learning paths, etc.

Para los estudiantes, la clase principal InterfazEstudiante (que hereda de InterfazUsuarioAbstracta) implementa todas las historias de usuario correspondientes a un estudiante y no necesita otras clases externas (además de las clases de utilidades).

Para los profesores hay dos clases principales, una para su perspectiva como creadores de actividades y learning paths y otra como revisores de progreso y actividades hechas por estudiantes en los learning paths. Para la perspectiva como revisor está la clase InterfazProfesorRevisor que no requiere ninguna clase adicional a las de utilidades e implementa todas las historias de usuario de esa perspectiva. Por el otro lado, para profesor como creador está la clase InterfazProfesorCreador, que implementa todas las historias de usuario de esta perspectiva con ayuda de varias clases auxiliares.

Para la perspectiva de profesor creador, la clase principal utiliza la clase ActividadesDisponibles, que funciona como un controlador que lo lleva al creador indicado para el tipo de actividad que desea crear o modificar. Hay un creador por cada tipo de actividad (lo cual facilita mantenimiento y mejora del programa) y todos heredan de la clase abstracta CreadorActividad, para facilitar la interacción e implementación de otras clases adicionales, ya que por si misma, se encarga de pedir los elementos comunes para todos los tipos de actividades (que son los atributos que se encuentran en la clase Actividad).