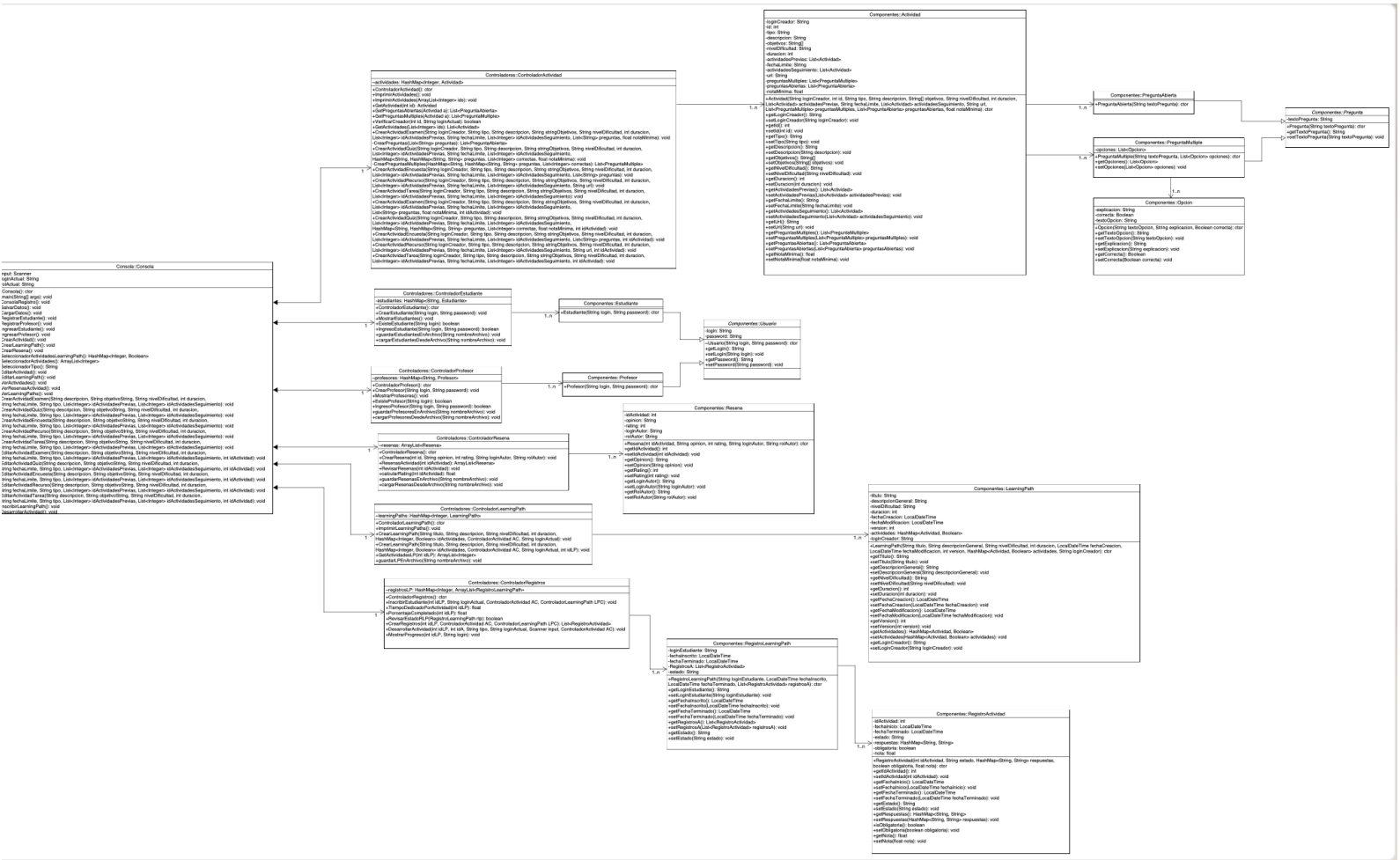
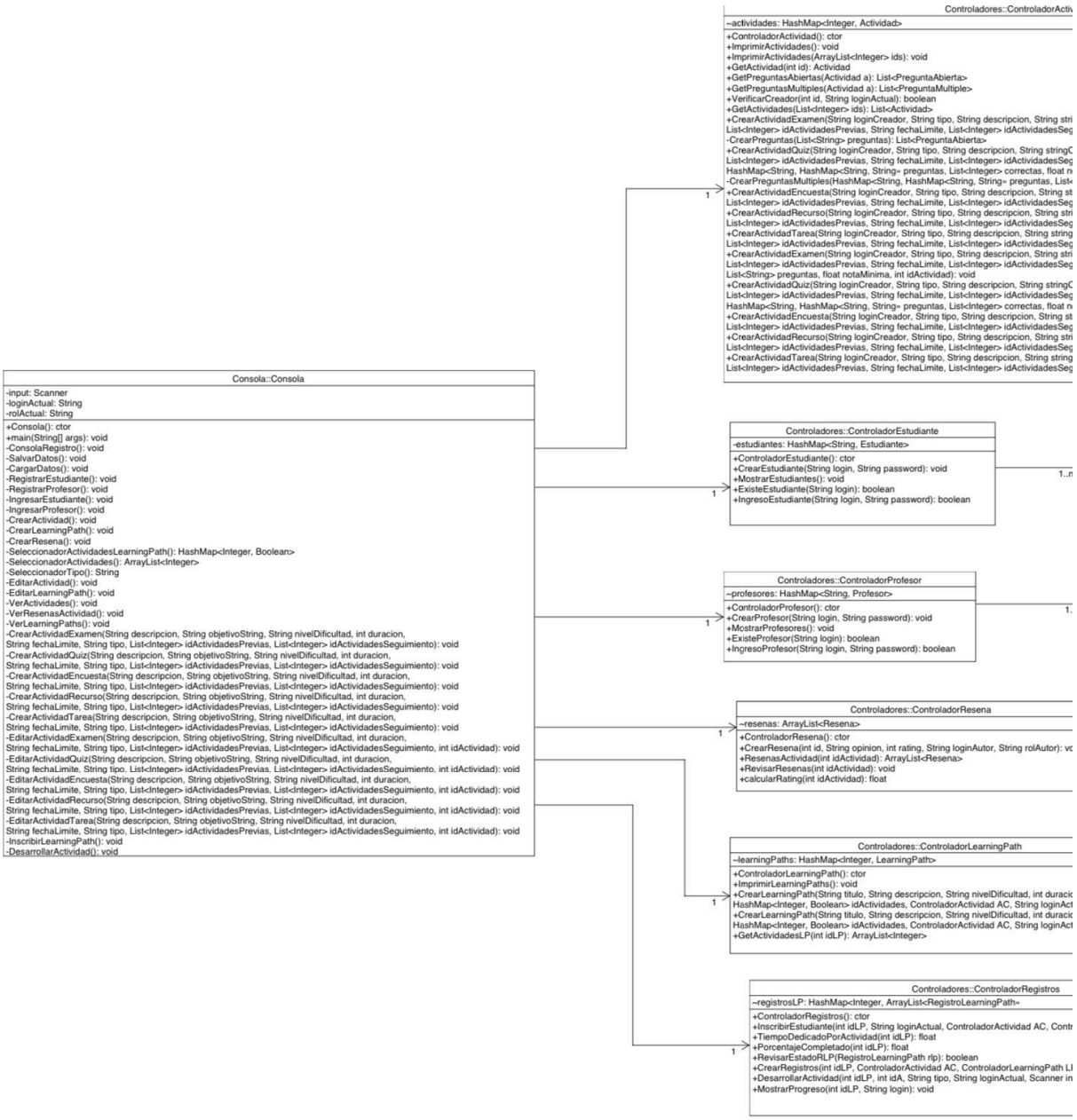
Angela Bárcenas -202320484
Tomas Hernandez 202320326
Nicolas Toro

A. Un diagrama de clases de diseño que incluya todas las clases, incluyendo sus



relaciones, atributos y métodos.

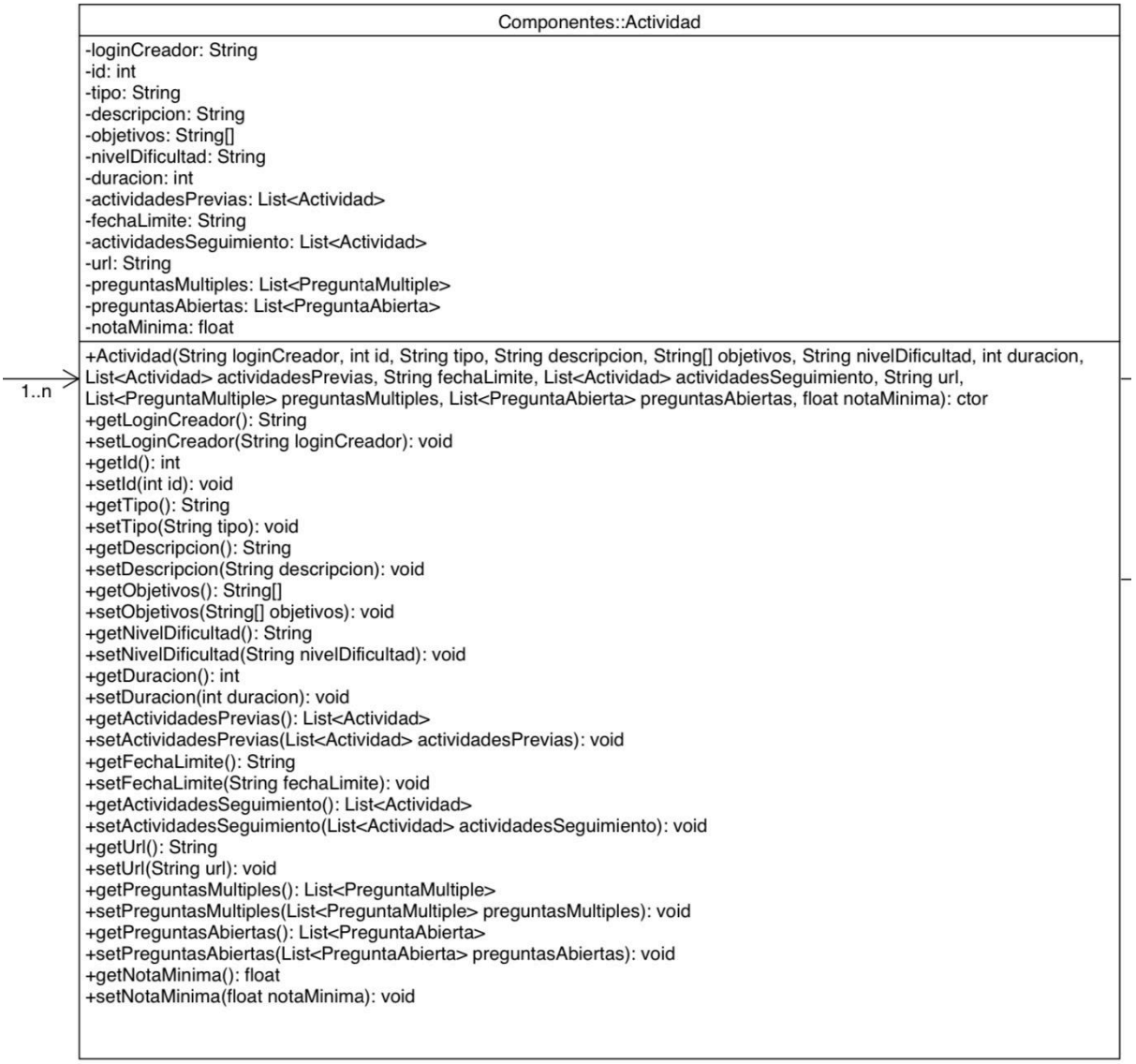
A continuación vamos a descomponer por partes el diagrama anterior, con el fin de entenderlo mejor y poder visualizar de manera clara cada clase con sus métodos y atributos.



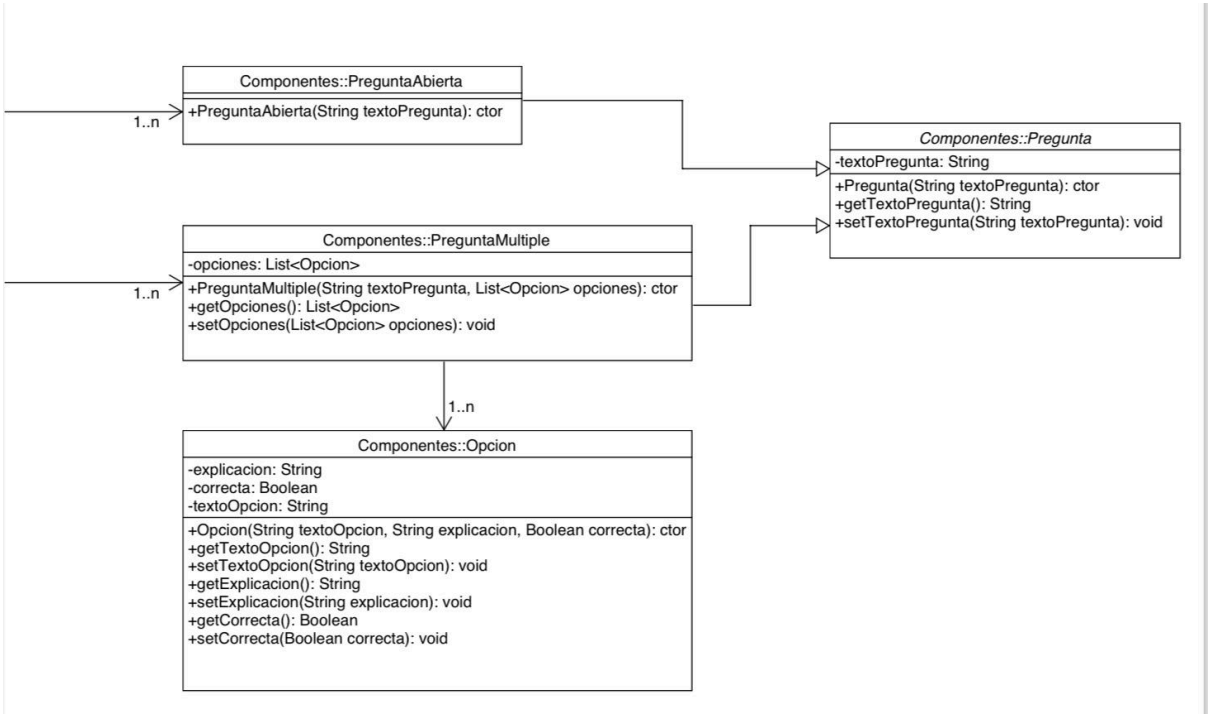
La Consola, siendo la presentación de todo el proyecto se relaciona directamente con los controladores de cada una de las clase(existe solamente un controlador por cada clase) , como se puede observar en la imagen anterior; mediante esta se implementan los métodos que se relacionan directamente con cada usuario y que se desprenden del controlador de cada clase.

Actividades:

Controlador actividades; tiene como responsabilidad principal la creación , administración y verificar las diferentes actividades del sistema.

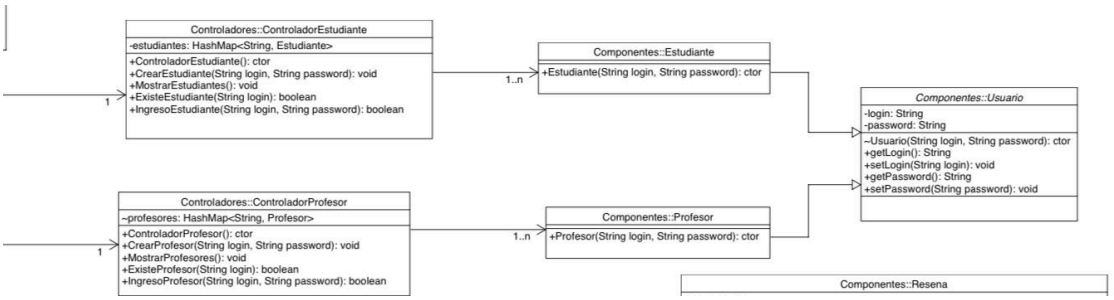


La clase Actividad modela una actividad educativa dentro del sistema, con atributos clave como el creador (loginCreador), identificador (id), tipo, descripción, objetivos, nivel de dificultad, duración, y URLs asociadas. Además, incluye colecciones de actividades previas y de seguimiento (List<Actividad>), así como preguntas abiertas y de opción múltiple asociadas a la actividad. Los métodos de la clase son principalmente getters y setters que permiten modificar y acceder a los atributos. También gestiona atributos como la fecha límite (fechaLimite), el nivel de dificultad, y la nota mínima requerida (notaMinima), lo que sugiere que está diseñada para actividades evaluativas.



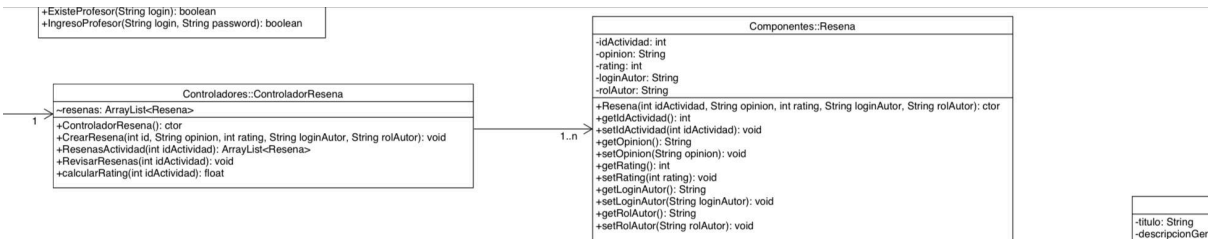
Las clases componentes heredan los atributos y métodos de la clase componentes pregunta, de la cual se desprenden las tres opciones de pregunta existentes en el sistema y que varían en su forma de calificar y en la forma de ser respondidas.

Usuarios:



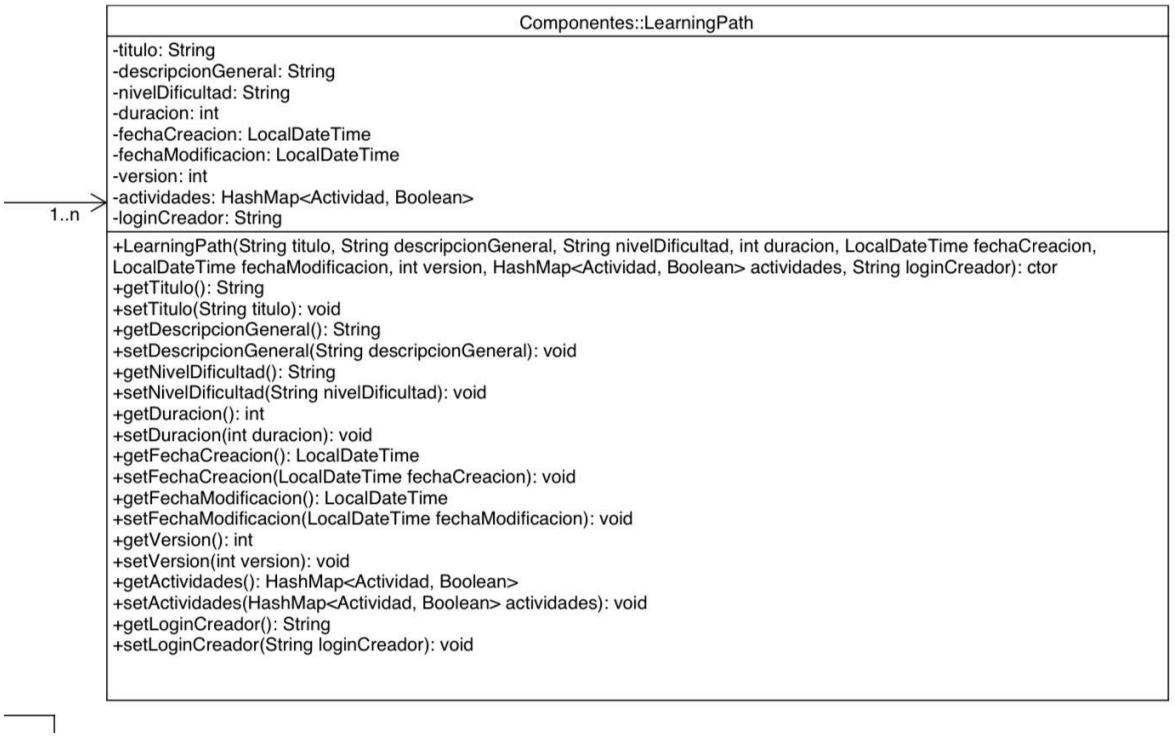
La clase usuario es una clase abstracta de la ua heredan las clases profesor y estudiante; esto debido a que ambas clases comprate atributos, sin embargo sus métodos, permisos y su uso del sistema es totalmente diferente, puesto que los profesores cuenta con más permisos como por ejemplo para crear y editar los learning paths. A pesar de esto tanto profesor como estudiante, son usuarios que necesitan registrarse e ingresar con un login y una contraseña.

Reseñas:



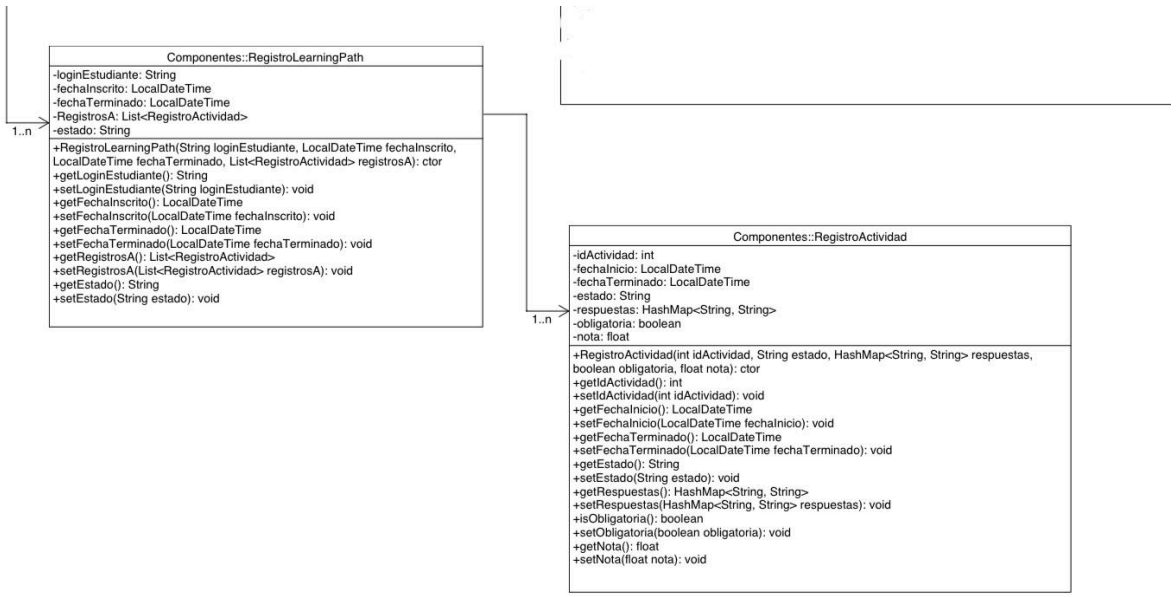
Reseñas son aquellas opiniones que pueden crear los estudiantes acerca de las actividades y los Learning paths, el controlador se encarga de permitir al estudiante crear, revisar y leer las reseñas que tiene que cumplir con los criterios estipulados de la clase reseña.

Learning Paths:



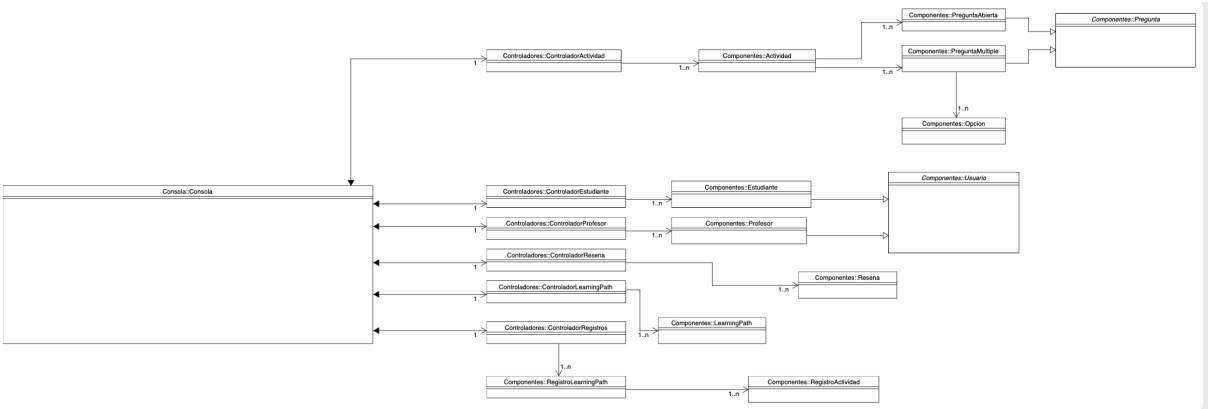
Los learning paths, fundamentales en este sistema vienen ligados con actividades, dado que se componen directamente de ellas, un learning oath está conformado por varias actividades p, pueden ser creados únicamente por profesores y deben tener información básica que se estipula en los atributos de la clase.

Registros:



Por último los componentes de registro permiten almacenar la información de cada learning path, que guarda la información de cada actividad que lo compone, es decir en estas clases, se implementa el almacenamiento del avance de cada estudiante en cada una de las actividades, así como su calificación,la fecha en la que lo realizó, el tiempo que le tomó y demás información relevante para el profesor y para el estudiante. Esta clase permite guardar los avances de cada usuario para así facilitar sacar promedios o información tanto genérica como individual de cada uno de los learning paths.

B. Un diagrama de clases de alto nivel, que incluya todas las clases y sus relaciones, pero no todos los métodos ni atributos. Este diagrama facilitará entender las relaciones entre clases.



Las relaciones entre clases, son básicamente asociaciones que nos indican la existencia de una conexión lógica e incluso funcional entre ellas, es decir nos muestra cómo los objetos de una clase se relacionan con la otra. En nuestro diagrama;

Consola:

La Consola actúa como la interfaz donde los usuarios interactúan con el sistema, mientras que los controladores gestionan la lógica y los datos.

- la consola es unidireccional es decir; solo necesita interactuar con los controladores para realizar tareas específicas
- la consola depende de los controladores para poder ejecutar las funcionalidades del sistema

Controladores:

Cada controlador gestiona un conjunto de datos específicos, como ControladorEstudiante gestiona un HashMap de Estudiante y ControladorProfesor un HashMap de Profesor.

- La relación entre cada controlador y su conjunto de datos (estudiantes, profesores, etc.) es de composición, ya que el controlador crea, modifica y borra estos datos, y si se destruye el controlador, también lo hacen sus datos.

Controlador Learnig Path y Actividades:

Para construir y administrar los "Learning Paths", el ControladorLearningPath requiere interactuar con el ControladorActividad, dado que un Learning Path se compone de diversas actividades.

- LearningPath Controlador incluye un HashMap de actividades. Esto nos muestra que se compone de diversas actividades que se vinculan de forma lógica a través del controlador.

Controlador Reseñas y las reseñas:

ControladorResena tiene una agregación con la clase Resena, debido que contiene una lista de Resena (ArrayList<Resena>). Esto implica que el controlador gestiona las reseñas, pero las reseñas pueden existir independientemente del controlador.

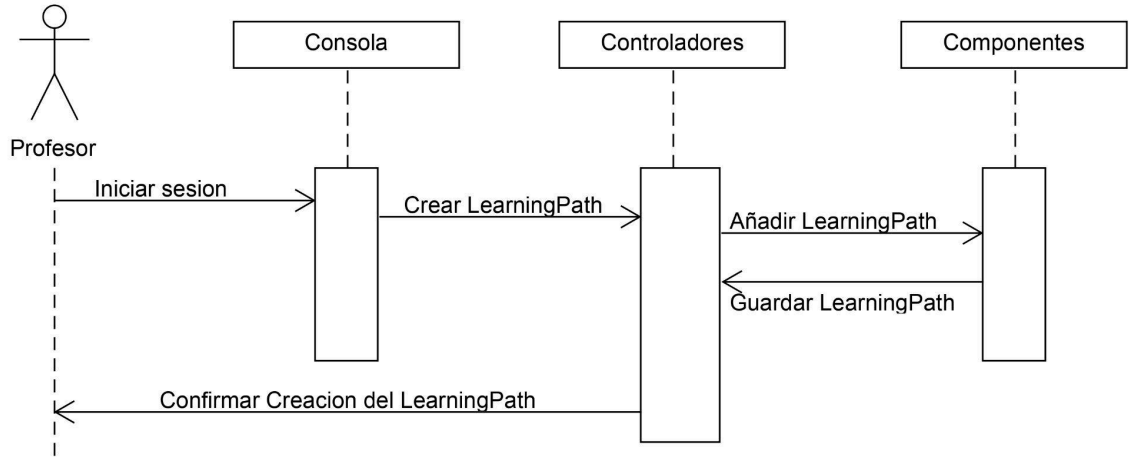
- Un ControladorResena puede tener muchas reseñas, como se indica en el uso de una ArrayList.

ControladorRegistros y **RegistroLearningPath**:

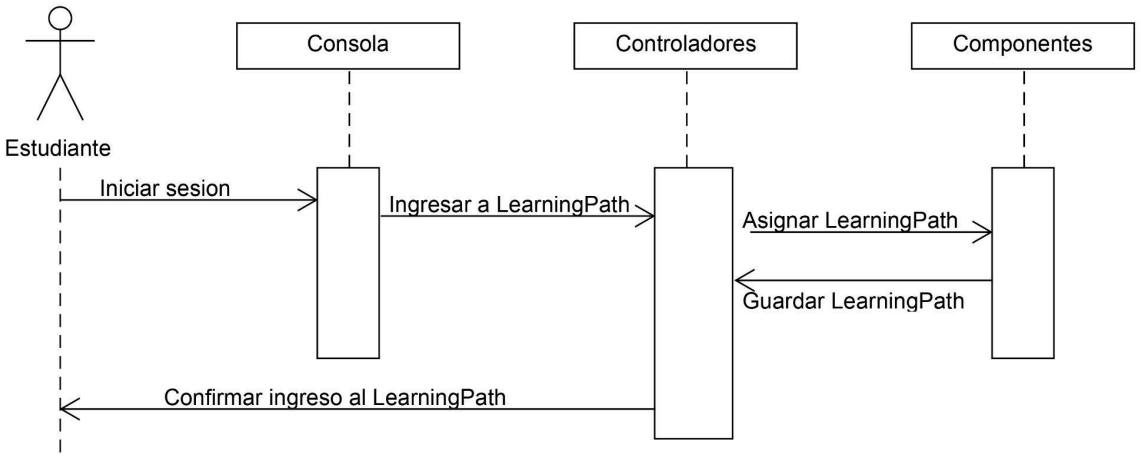
- ControladorRegistros gestiona registros de estudiantes inscritos en Learning Paths a través de un HashMap<Integer, ArrayList<RegistroLearningPath>>. Esto indica una relación fuerte en la cual múltiples registros pueden existir para cada Learning Path.

C. Diagramas de secuencia para las funcionalidades que usted considere críticas:

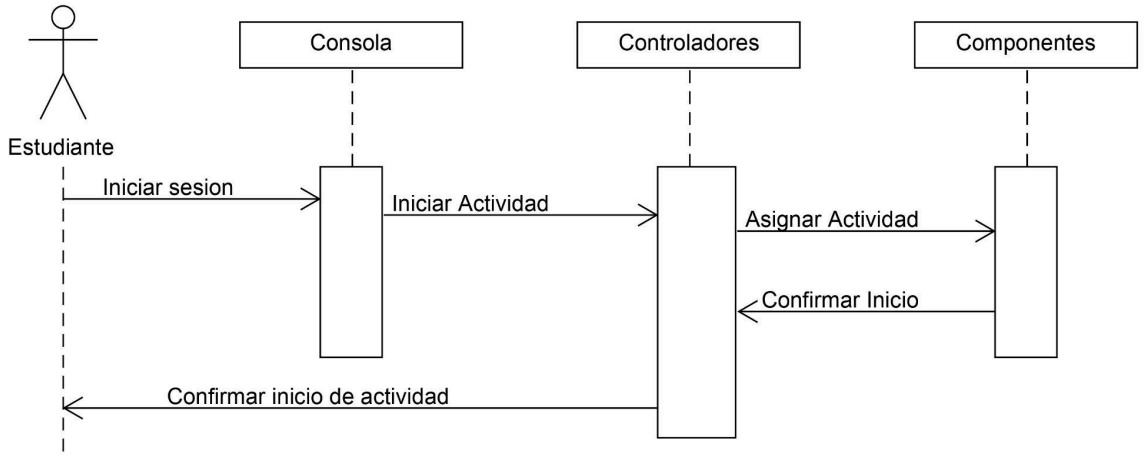
1.
Este diagrama muestra cómo debe interactuar un usuario de tipo profesor con el sistema para poder crear un learning Path, esto se hace de la siguiente manera, el profesor inicia sesión en el sistema para que pueda crear el learning Path, luego da toda la información necesaria para crear un learning Path, esta información se le pasa a los controladores y finalmente a los componentes, donde finalmente le envía la instrucción de guardar el Learning Path a los controladores, y estos se encargan de enviar una confirmación directa al usuario de tipo profesor que el Learning Path fue creado.



2.
Este diagrama de secuencia ilustra como un usuario de tipo Estudiante interactúa con el sistema para ingresar a un learning Path que le interesa, el estudiante inicia sesión en el sistema para que pueda ingresar al learning Path, esta instrucción se le pasa a los controladores y finalmente a los componentes, donde finalmente le envía la instrucción de guardar el Learning Path a los controladores, y estos se encargan de enviar una confirmación directa al usuario de tipo profesor que el Learning Path fue creado.



3.
En este diagrama se ilustra cómo el usuario de tipo estudiante interactúa con el sistema al momento de iniciar una actividad, para iniciar el estudiante debe iniciar sesión para que pueda dar la instrucción al controlador de iniciar una actividad, este controlador envía una instrucción a los componentes que da inicio a la actividad, el componente envía una confirmación de inicio de actividad al controlador, y este controlador se encarga de informarle al estudiante que la actividad ha sido iniciada.



4.
En este diagrama se evidencia como un usuario de tipo Profesor debe interactuar con el sistema para poder agregar una actividad nueva, esto se hace de la siguiente manera, el profesor inicia sesión en el sistema para que pueda agregar la actividad, luego da toda la información necesaria para agregarla, esta información se le pasa a los controladores y finalmente a los componentes, donde finalmente le envía la instrucción de agregar la actividad a los controladores, y estos se encargan de enviar una confirmación directa al usuario de tipo profesor que la actividad fue creada.