**Diseño e implementación: Grupo 4**

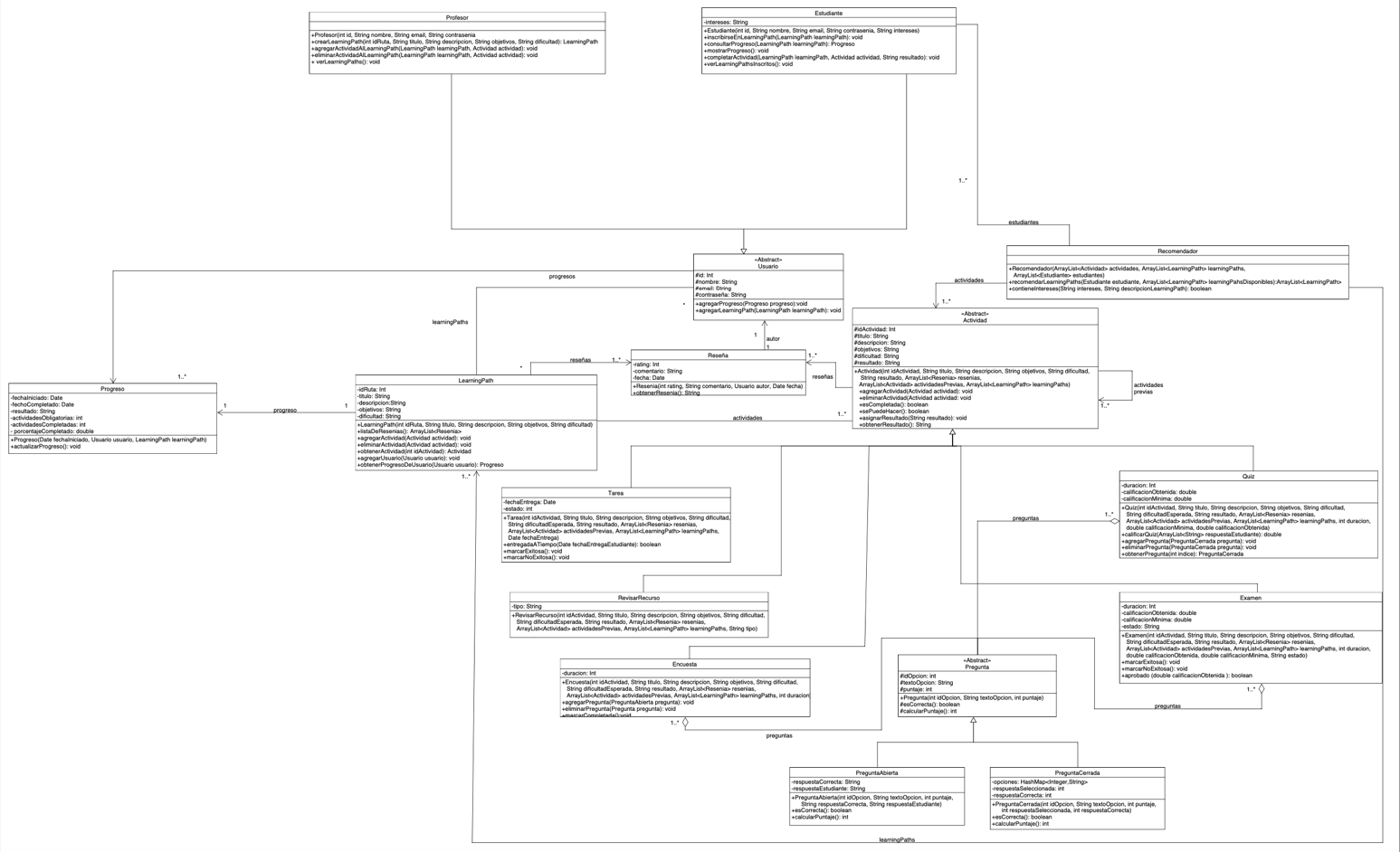
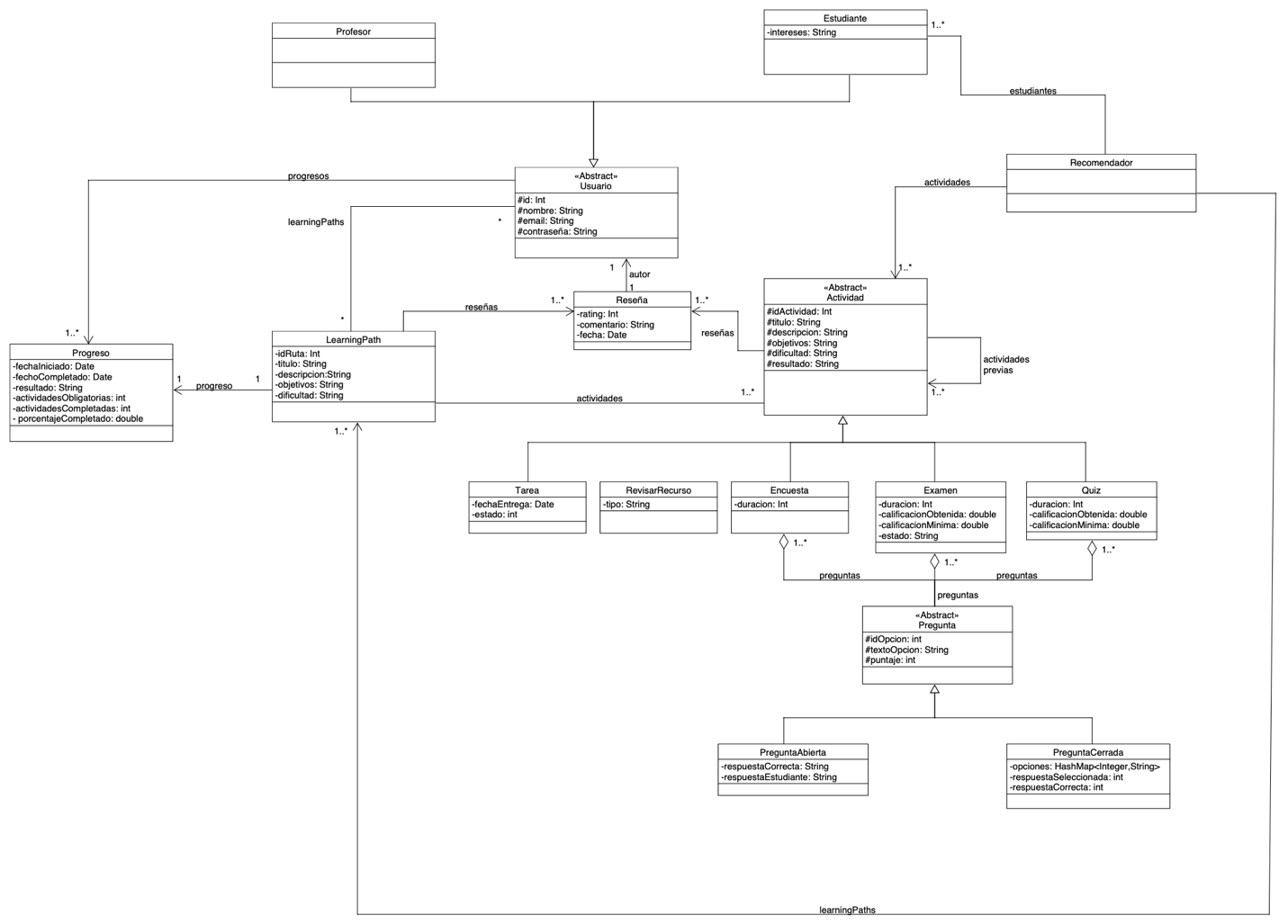
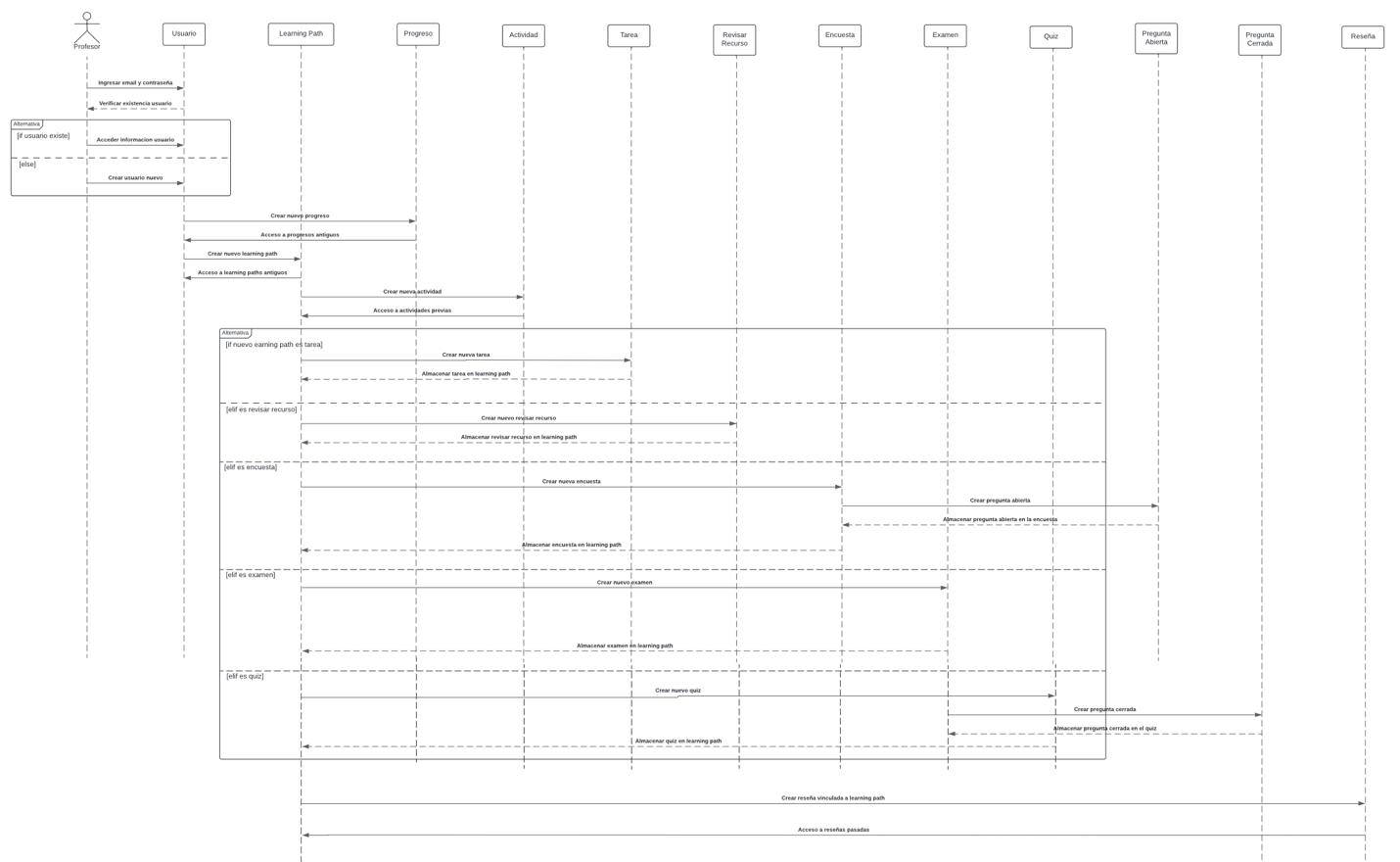
Diagrama de clases de diseño

Diagrama de clases de alto nivel

Diagramas de secuencia

Profesor:

Estudiante:

Justificación por clases

Usuario:

Métodos:

* agregarProgreso(…): Genera una nueva instancia para el registro de un progreso, registrando los datos necesarios de la clase Progreso.
* agregarLearningPath(…): Genera una nueva instancia para el registro de un Learning Path, registrando los datos necesarios de la clase LearningPath.

Profesor:

Métodos:

* Profesor(…): Registra un nuevo usuario profesor.
* crearLearningPath(…): Genera un nuevo Learning Path con la informacion requerida, registrada por el usuario profesor.
* agregarActividadAlLearningPath(…): Agrega la instancia de actividad nueva creada al Learning Path relacionado al profesor.
* eliminarlActividadAlLearningPath(…): Elimina la instancia de actividad referenciada en el LearningPath relacionado al profesor.
* verLearningPaths(…): Muestra los Learning Paths vinculados al usuario professor.

Estudiante:

Métodos:

* Estudiante(…): Registra un nuevo usuario estudiante.
* inscribirseEnLearningPath(…): Inscribe al usuario estudiante al Learning Path de elección.
* consultarProgreso(…): Muestra el progreso en el Learning Path relacionado al usuario estudiante.
* mostrarProgreso(…): Muestra el progreso en el Learning Path relacionado al usuario estudiante.
* completarActividad(…): Genera acceso a la actividad que el usuario estudiante desea completar dentro de un Learning Path definido.
* verLearningPathsInscritos(…): Muestra todos los Learning Paths relacionados con el usuario estudiante.

Recomendador:

Métodos:

* Recomendador(…): Genera una nueva instancia de un estudiante recomendador.
* recomendarLearningPaths(…): Muestra las recomendaciones de Learning Paths sugeridas al estudiante de acuerdo a los progresos e intereses.
* contieneIntereses(…): Booleano que indica si el Learning Path contiene los intereses del estudiante.

Progreso:

Métodos:

* Progreso(…): Genera una nueva instancia de progreso.
* actualizarProgreso(…): Actualiza la información del progreso.

LearningPath:

Métodos:

* LearningPath(…): Crea un nuevo Learning Path con toda la información necesaria ingresada por el usuario.
* ListaDeResenias(…): Muestra todas las reseñas existentes.
* agregarActividad(…): Genera una nueva instancia para una actividad.
* eliminarActividad(…): Elimina la actividad seleccionada.
* obtenerActividad(…): Obtiene la información relevante a la actividad en búsqueda.
* agregarUsuario(…): Crea un nuevo vinculo de acceso de un usuario a este Learning Path.
* obtenerProgresoDeUsuario(…): De acuerdo a la búsqueda ingresada, se obtiene el progreso relacionado a ese usuario.

Reseña:

Métodos:

* Resenia(…): Crea una nueva reseña con la información otorgada por el usuario.
* obtenerResenia(…): Obtiene la información relevante a la reseña en búsqueda.

Actividad:

Métodos:

* Actividad(…): Crea una nueva actividad con toda la información necesaria ingresada por el usuario.
* agregarActividad(…): Genera una nueva instancia para una actividad.
* eliminarActividad(…): Elimina la actividad seleccionada.
* esCompletada(): Booleano que indica si la pregunta fue completada.
* sePuedeHacer(): Booleano que indica si la pregunta está disponible para que el usuario estudiante pueda completar.
* asignarResultado(…): Asigna un resultado en escrito por el profesor de lo actividad.
* obtenerResultado(…): Se refleja el resultado previo.

Tarea:

Métodos:

* Tarea(…): Crea una nueva tarea con toda la información necesaria ingresada por el usuario.
* entregadaATiempo(…): Booleano que indica si fue entregada la tarea a tiempo o no.
* marcarExitosa(): Booleano que indica si la pregunta fue aprobada.
* marcarNoExitosa(): Booleano que indica que la pregunta no fue aprobada.

RevisarRecurso:

Métodos:

* revisarRecursos(…):

Encuesta:

Métodos:

* Encuesta(): Crea una nueva encuesta con toda la información necesaria ingresada por el usuario.
* agregarPregunta(): Genera una instancia para adicionar una nueva pregunta.
* eliminarPregunta():
* marcarCompletada(): Booleano que indica que la encuesta fue completada.

Quiz:

Métodos:

* Quiz(…): Crea un nuevo quiz con toda la información necesaria ingresada por el usuario.
* calificarQuiz(): Calcula el puntaje y retorna el entero correspondiente.
* agregarPregunta(): Genera una instancia para adicionar una nueva pregunta.
* eliminarPregunta(): Elimina la pregunta referenciada.
* obtenerPregunta(): Muestra la pregunta referenciada.

Examen:

Métodos:

* Examen(…): Crea un nuevo examen con toda la información necesaria ingresada por el usuario.
* marcarExitosa(): Booleano que indica si la pregunta fue aprobada.
* marcarNoExitosa(): Booleano que indica que la pregunta no fue aprobada.
* aprobado(): Booleano que indica si el examen fue aprobado o no.

Pregunta:

Métodos:

* Pregunta(…): Crea una nueva pregunta con la información ingresada por el usuario.
* esCorrecta(): Booleano que indica si la respuesta es correcta o no.
* calcularPuntaje(..): Calcula el puntaje y retorna el entero correspondiente.

PreguntaAbierta:

Métodos:

* PreguntaAbierta(…): Crea una nueva pregunta con la información ingresada por el usuario.
* esCorrecta(): Booleano que indica si la respuesta es correcta o no.
* calcularPuntaje(): Calcula el puntaje y retorna el entero correspondiente.

PreguntaCerrada:

Métodos:

* PreguntaCerrada(…): Crea una nueva pregunta con la información ingresada por el usuario.
* esCorrecta(): Booleano que indica si la respuesta es correcta o no.
* calcularPuntaje(): Calcula el puntaje y retorna el entero correspondiente.