

DOCUMENTO PROYECTO 2

REQUERIMIENTOS, LIMITACIONES Y ASUNCIONES:

- La aplicación debe estar hecha en Java
- Es una plataforma independiente
- No hay análisis ni tablas de datos para facilitar la digestión de los datos de los estudiantes
- Toda la información tiene que ser persistente
- La información se tiene que guardar en archivos dentro de una carpeta única para el programa
- La carpeta no puede ser la misma donde se encuentra el código fuente de la aplicación
- Los Profesores deben poder crear *Learning Paths*
- Los Profesores deben poder editar *Learning Paths*
- Los Profesores deben poder crear Actividades
- Los Profesores deben poder editar sus Actividades creadas
- Los Profesores deben poder clonar Actividades
- Los Profesores deben poder añadir Actividades existentes a un *Learning Path*
- Los *Learning Paths* deben contar con título
- Los *Learning Paths* deben contar con una descripción general
- Los *Learning Paths* deben contar con una lista de objetivos
- Los *Learning Paths* deben contar un nivel de dificultad
- Los *Learning Paths* deben contar una secuencia de Actividades
- Los *Learning Paths* deben contar con una duración calculada en minutos
- Los *Learning Paths* deben contar con un rating
- Los *Learning Paths* deben contar con Metadatos como fecha de creación
- Los *Learning Paths* deben contar con Metadatos como fecha de modificación
- Los *Learning Paths* deben contar con Metadatos como versión
- Las Actividades deben tener una descripción
- Las Actividades deben tener un objetivo
- Las Actividades deben tener una duración esperada
- Las Actividades deben tener una fecha límite basada en la Actividad anterior
- Las Actividades deben tener una Actividad de seguimiento
- Las Actividades deben tener un resultado
- Las Actividades deben ser obligatorias u opcionales
- Los Estudiantes deben poder dejar reseñas
- Los Estudiantes deben poder ver la estructura del *Learning Path*
- Los Estudiantes deben ver la descripción de la actividad
- Los Estudiantes solo pueden tener una (1) Actividad a la vez
- Los Estudiantes deben tener un progreso del *Learning Path*, calculado desde el porcentaje de actividades obligatorias completadas exitosamente.
- El Sistema debe llevar un registro del progreso de los estudiantes
- El Sistema debe registrar cuando un Estudiante inicia un *Learning Path*
- El Sistema debe registrar cuando un Estudiante acaba un *Learning Path*
- El Sistema debe registrar el tiempo dedicado por actividad de un Estudiante
- El Sistema debe registrar la tasa de éxito por Actividades de un Estudiante
- El Sistema debe registrar la tasa de fracaso por Actividades de un Estudiante
- El Sistema debe registrar cuando un Estudiante completa una Actividad
- El Sistema debe contar con registro de usuarios

- El Sistema debe contar con autenticación
- El Sistema debe contar con un sistema de *feedback* con ratings
- Todos los Usuarios del Sistema deben tener un *login* y un *password*
- El Sistema debe soportar diferentes tipos de Actividades
- El primer tipo de Actividad es de revisar un Recurso Educativo
- El resultado de los Recursos Educativos siempre es exitoso al completarse
- El segundo tipo de Actividad es una Tarea.
- Al Estudiante realizar una Tarea, registra en el sistema que fue entregada. La tarea tendrá estado de “enviada” hasta ser revisada por el Profesor responsable y sea marcada como exitosa o no exitosa.
- El tercer tipo de Actividad es un Quiz.
- El Quiz es una serie de preguntas cerradas de única respuesta, cada pregunta debe tener una explicación.
- El Quiz tiene calificación mínima.
- El cuarto tipo de Actividad es un Examen.
- El Examen es una serie de preguntas abiertas. Permanece “enviado” hasta que el profesor califica las respuestas de los estudiantes.
- El quinto tipo de Actividad es una Encuesta.
- Las Encuestas son series de preguntas abiertas al Estudiante. Las encuestas se completan después de que el Estudiante las envíe.

Consola:

Para la entrega se finalizó la integración de las funcionalidades deseadas de la aplicación en la consola. Actualmente la aplicación es capaz de iniciar sesión, como un usuario de estudiante o profesor. El profesor tiene la capacidad de crear y editar Learning Paths, así como crear, clonar y editar las actividades creadas por si mismos. Los estudiantes, por otro lado, tienen la oportunidad de interactuar con el sistema mediante la inscripción a Learning Paths, el inicio de actividades, y la creación de reseñas para los Learning Paths completados. El sistema tiene la capacidad de crear y guardar su biblioteca de información con todos los Learning Paths, Actividades, Preguntas y Usuarios creados que hacen parte del sistema general. El proceso de guardado en la base central al funcionamiento de la aplicación se basa en el almacenamiento de los objetos desde códigos identificadores que son llamados entre sí. La interfaz de la consola fue desarrollada en mente como se desea traducir el trabajo a la interfaz gráfica 2D, cada acción esta pensada dentro de su propia ventana de opciones que puede ser diseñada y anidada en otra, por lo que la lógica de funcionamiento esta gran parte implementada y falta la traducción grafica de la misma.

Flujo General de los Tests

- Setup Inicial: Se inicializan los objetos requeridos para cada prueba, como Estudiante, Profesor, LearningPath, y actividades. Se configuran listas, mapas, y dependencias necesarias.
- Ejecución de Funcionalidades: Se prueban métodos clave como validarUsuario, crearLearningPath, crearReseña, y persistencia de datos.
- Validación de Resultados: Aserciones (assertEquals, assertTrue, assertFalse, etc.) aseguran que los resultados reales coincidan con los esperados.

- Limpieza Final: Se restablece el estado global (EstadoGlobal) o se eliminan archivos temporales creados durante las pruebas.

Cobertura Total del Sistema Entidades Probadas: EstadoGlobal, Profesor, Estudiante, LearningPath, Reseña, RecursoEducativo, Tarea, Quiz, Encuesta, Examen. Aspectos Validados:

- Gestión de usuarios (creación, autenticación, acceso global).
- Gestión y persistencia de LearningPaths.
- Interacción con actividades específicas.
- Creación, edición, y manejo de reseñas.
- Integridad y consistencia de datos.

Pruebas para EstadoGlobal

- EstadoGlobal es el contenedor principal que gestiona usuarios y LearningPaths. Las pruebas aseguran que las funcionalidades globales se comporten correctamente.

testValidarUsuario:

- Valida que se pueda verificar la existencia de usuarios en el sistema global.
- Asegura que los usuarios no existentes se rechacen.

testValidarLogin:

- Comprueba la autenticación de usuarios con credenciales correctas.
- Asegura que contraseñas incorrectas o usuarios no existentes sean rechazados.

testCrearLearningPath:

- Simula la creación de un LearningPath por un Profesor.
- Verifica que el LearningPath se almacene en el contenedor global.

testEditarLearningPath:

- Valida que los LearningPaths almacenados puedan ser modificados correctamente.

testCargarSistemaGlobal:

- Simula la inicialización del sistema con datos predefinidos.
- Asegura que los contenedores globales no estén vacíos tras la carga.

Pruebas para Profesor

- La clase Profesor permite a los usuarios gestionar LearningPaths y actividades.

testCrearLearningPath:

- Verifica que un profesor pueda crear un nuevo LearningPath.
- Valida que los LearningPaths creados se añadan a su lista personal.

testGetLearningPathbyNombre:

- Prueba la búsqueda de LearningPaths por su nombre.
- Asegura que los LearningPaths no existentes retornen null.

testGetActividadesCreadas:

- Valida la gestión de actividades creadas por un profesor.
- Asegura que estas actividades puedan ser consultadas correctamente.

Pruebas para Estudiante

- Los tests de Estudiante aseguran que las interacciones del estudiante con LearningPaths, actividades, y reseñas sean correctas.

Inscripción y Finalización de LearningPaths:

- testInscribirLearningPath: Verifica que un estudiante pueda inscribirse en un LearningPath.
- testTerminarLearningPathCompletado: Valida que el estudiante pueda completar un LearningPath con progreso al 100%.
- testTerminarLearningPathNoCompletado: Asegura que los LearningPaths no completados no puedan ser finalizados.

Gestión de Reseñas:

- testCrearReseñaParaActividadCompletada: Comprueba que los estudiantes puedan crear reseñas para actividades completadas.
- testCrearReseñaParaActividadNoCompletada: Asegura que no se puedan crear reseñas para actividades no completadas.
- testEditarReseña: Valida que las reseñas existentes puedan ser editadas.
- testReseñasMúltiples: Verifica que un estudiante pueda gestionar varias reseñas simultáneamente.

Pruebas para LearningPath

- LearningPath es una clase central que organiza las actividades del sistema. Las pruebas validan la gestión de actividades y su persistencia.

testConstructorAndGetters: Valida los atributos iniciales de un LearningPath.

testPersistenciaSalvarYCargarLearningPath:

- Asegura que un LearningPath se pueda guardar y recuperar correctamente.
- Actividades Específicas: Cada tipo de actividad tiene pruebas personalizadas:

Test de Actividades:

- RecursoEducativo: Validación de atributos como tipo, URL, y título.

- Tarea: Gestión del estado de envío (No entregado, Entregado, Exitoso).
- Quiz: Validación de selección de opciones correctas e incorrectas.
- Encuesta: Manejo de preguntas abiertas y estado de completado.
- Examen: Gestión del estado de calificación y manejo de preguntas abiertas.

Pruebas para Reseña: Las pruebas de Reseña aseguran que esta clase funcione correctamente como una entidad independiente.

- testConstructorAndGetters: Valida que los atributos como comentario, rating, creador y fecha se inicialicen correctamente.
- testEditarComentario: Comprueba que se puedan editar los comentarios de una reseña.
- testEditarRating: Verifica que se pueda modificar la calificación (rating) de una reseña.

Persistencia:

La persistencia se hizo en dos partes la primera se utilizó la clase learning path la cual contiene la mayor parte de toda la info necesaria para la persistencia, además de esta clase se usó la clase Usuario la cual es ligada con la persistencia de learning path para poder guardar toda la información de los diferentes usuarios la diferencia de esto al anterior proyecto es que en este caso si esta su implementación sin un error de syntax en todo el código.

También se le implemento a estas excepciones para que se prevengan errores que pueden llegar a pasar y que el código no se detenga por completo.

Interfaz:

Se le añadió un jtestframe para que se puede empezar a implementar una interfaz de usuario al programa, para este se utiliza swing y la extensión WindowBuilder.

Para la realización del Diagrama UML se diseñó el programa con *Learning Paths* como estructura principal que contuviera las actividades y diferentes preguntas que la conforman. Dentro de los demás sistemas principales planeados se encuentra el sistema de usuarios, responsable de realizar el registro, ingreso y autenticación para la plataforma. Por otro lado, se desea crear una consola específica para los profesores que les permita crear, gestionar y editar los *Learning Paths* disponibles en la plataforma, así como las actividades y preguntas, a las cuales se les asignó un ID único con el objetivo de lograr usar preguntas y actividades antiguas o modificadas para la creación o modificación de los *Learning Paths*. De la misma manera, se diseñó el sistema de reseñas para que sea accesible a todos los usuarios. Finalmente, se desarrolló un sistema de persistencia para guardar la información general del sistema relevante, correspondiente a la de los *Learning Paths*, así como la de los usuarios para poder mantener la información registrada y guardada.

UML MODELO

UML PERSISTENCIA