

# **Proyecto2: Learning Paths**

Luis Angel Aristizabal Correa  
202314935

Andrés Felipe Martínez Tello  
201913197

Natalia Isabela Erazo González  
202213680

## Contenido

Contexto .....	3
Glosario .....	4
Historias de Usuario. ....	5
Pruebas Unitarias .....	9
Pruebas de Integración. ....	20
Diseño Consola .....	22
Menú Principal .....	22
Diseño de la Consola para Profesores .....	22
Menú Principal de Profesores .....	22
Opciones específicas para Profesores .....	23
Diseño de la Consola para Estudiantes .....	25
Menú Principal de Estudiantes .....	25
Opciones específicas para Estudiantes .....	25
Observaciones .....	26

## Contexto

El proyecto Learning Paths es un sistema de rutas de aprendizaje diseñado para facilitar el desarrollo de habilidades en estudiantes mediante actividades estructuradas, ofreciendo a profesores la capacidad de crear y gestionar rutas de aprendizaje personalizadas que se ajusten a las necesidades y progresos individuales de los estudiantes. Los estudiantes, por su parte, pueden inscribirse en estas rutas, acceder a actividades recomendadas, seguir su propio avance y dejar retroalimentación a los profesores sobre las actividades completadas.

En la primera etapa de desarrollo, se implementaron las bases del sistema, enfocándose en la lógica de negocio y la estructura interna. Esto incluyó la gestión de usuarios (registro y autenticación), la creación y administración de rutas de aprendizaje, la configuración de actividades y la medición de progresos. Además, se definió un sistema de notificaciones y de reseñas, proporcionando una experiencia estructurada de aprendizaje que también incluía un componente de retroalimentación.

Para esta entrega actual, el proyecto profundizará en la funcionalidad y usabilidad del sistema, integrando una interfaz de usuario en consola que permitirá a profesores y estudiantes acceder de manera más interactiva y amigable a todas las funcionalidades del sistema, con validaciones en las entradas. También se implementarán pruebas automatizadas para garantizar el correcto funcionamiento del sistema, mediante pruebas unitarias y de integración para los componentes clave. A nivel de contenido, se incluirá un nuevo tipo de actividad en los Learning Paths: quices de verdadero/falso. Asimismo, se optimizará la gestión de datos, permitiendo que el sistema cargue y guarde automáticamente la información en archivos, sin requerir la intervención del usuario para el manejo de persistencia. Con estas mejoras, se busca avanzar hacia una aplicación funcional y robusta, capaz de soportar una experiencia de aprendizaje digital más completa y efectiva.

## Glosario

- **Usuario:** Persona que interactúa con el sistema, ya sea como profesor o estudiante. Accede a las funcionalidades de la plataforma de Learning Paths según su rol.
- **Profesor:** Usuario que puede crear y gestionar rutas de aprendizaje, diseñar actividades, revisar evaluaciones y dar seguimiento al progreso de los estudiantes.
- **Estudiante:** Usuario que se inscribe en los Learning Paths creados por los profesores, completa actividades, realiza tareas y exámenes, y tiene la opción de dejar reseñas sobre las actividades realizadas.
- **Learning Path:** Secuencia estructurada de actividades y recursos educativos creada por un profesor. Su objetivo es guiar a los estudiantes en la adquisición de conocimientos o habilidades sobre un tema específico. Un Learning Path incluye tareas, quizzes, exámenes y recursos educativos, y el progreso del estudiante es registrado en el sistema.
- **Actividad:** Componente de un Learning Path que los estudiantes deben completar. Puede incluir tareas, quizzes, exámenes o la interacción con recursos educativos.
- **Quiz:** Actividad de evaluación compuesta por preguntas de verdadero/falso o de opción múltiple, que se califica automáticamente según los parámetros establecidos por el profesor.
- **Tarea:** Actividad que el estudiante debe completar y enviar para su posterior revisión y calificación por parte del profesor.
- **Examen:** Evaluación que el estudiante debe completar, y que, al igual que la tarea, requiere revisión y calificación del profesor para determinar si fue aprobada.
- **Recurso Educativo:** Material de apoyo, como archivos, videos, sitios web o libros, que el estudiante puede consultar como parte del Learning Path. Se considera completado cuando el estudiante lo marca como tal.
- **ProgresoPath:** Registro general del avance del estudiante dentro de un Learning Path, basado en el número de actividades completadas y su desempeño en estas.
- **ProgresoActividad:** Registro específico del avance del estudiante en cada actividad de un Learning Path, indicando su estado (en curso, completada, aprobada, etc.) y los resultados obtenidos.
- **Reseña:** Opinión o evaluación que el estudiante puede dejar sobre una actividad completada, proporcionando retroalimentación sobre su experiencia en el Learning Path.
- **Registro de Usuario:** Función del sistema encargada de gestionar el proceso de autenticación y registro de usuarios, permitiendo la creación de cuentas, validación de credenciales (login y password) y mantenimiento de un registro de los usuarios activos.

- **Persistencia de Datos:** Almacenamiento y recuperación de la información del sistema en archivos o bases de datos, garantizando que los datos sean accesibles y actualizables de manera segura y eficiente.

## Historias de Usuario.

*Tabla 1 Registro y Autenticación de Usuarios*

HU001	User Story
Como	Usuario
Quiero	Registrarme, acceder e interactuar en la plataforma
Para	Acceder a las funcionalidades del sistema dependiendo del usuario, como acceder a mi cuenta y mis Learning Paths.
Detalles adicionales	
El usuario debe contar con login y contraseña, además de que al registrarse debe añadir todos sus datos personales; su nombre, tipo de usuario, etc.	
Criterios de aceptación	
<p>El sistema debe permitir el registro de nuevos usuarios con un nombre de usuario único y una contraseña.</p> <p>En caso de que existan irregularidades durante el registro: Se mostrará un mensaje informando que el usuario no se pudo registrar de forma correcta e invitándolo a que lo intente de nuevo.</p> <p>El sistema debe permitir a los usuarios iniciar sesión con su nombre de usuario y contraseña.</p> <p>En caso de que existan irregularidades durante la autenticación: Se debe mostrar mensajes de error claros si el nombre de usuario o la contraseña son incorrectos.</p>	

*Tabla 2 Creación y Gestión de Learning Paths*

HU002	User Story
Como	Profesor
Quiero	Crear un nuevo Learning Path y administrar varios Learning Path
Para	Desarrollar herramientas de aprendizajes para mis estudiantes.
Detalles adicionales	
Estrictamente el Usuario debe ser un profesor para acceder a estas funciones.	

<b>Criterios de aceptación</b>
El sistema debe permitir a los profesores crear un Learning Path con un título, descripción, objetivos, nivel de dificultad, duración estimada y rating. En caso de no conseguirse: Se debe mostrar mensajes de error, si ocurrió un fallo en la creación del Learning Path. El sistema debe permitir a los profesores añadir, editar y eliminar actividades dentro de un Learning Path.  El sistema debe permitir a los profesores clonar actividades existentes y editarlas en caso de que no la hayan creado ellos.

*Tabla 3 Gestión de Actividades*

HU003	User Story
Como	Profesor
Quiero	Crear diferentes tipos de actividades (revisión de recursos, tareas, quizzes, exámenes, encuestas)
Para	Para diversificar las experiencias de aprendizaje.
<b>Detalles adicionales</b>	
El sistema debe permitir la creación de actividades dentro de un learning path con una descripción, objetivo, nivel de dificultad y duración esperada.	
<b>Criterios de aceptación</b>	
La actividad debe pertenecer a alguno de los tipos disponibles. El sistema debe permitir a los estudiantes ver detalles de cada actividad.	

*Tabla 4 Progreso y Seguimiento*

HU004	User Story
Como	Estudiante
Quiero	Ver mi progreso en un Learning Path
Para	Saber cuánto he avanzado y mi tasa de éxito.
<b>Detalles adicionales</b>	
Esto a través de número de actividades totales dentro del Learning Path, cantidad de actividades realizadas, cantidad de actividades exitosas del estudiante y cantidad de actividades fallidas del estudiante.	
<b>Criterios de aceptación</b>	
El sistema debe mostrar el progreso del estudiante como un porcentaje de las actividades obligatorias completadas.	

--

*Tabla 5 Feedback y Rating*

HU005	User Story
Como	Usuario
Quiero	Dejar reseñas y ratings en las actividades
Para	Compartir mi experiencia
Detalles adicionales	
Como usuario no se está obligado a dejar reseña sobre las actividades, siendo una función a decisión.	
Criterios de aceptación	
El sistema debe permitir a los usuarios dejar reseñas a las actividades.	

*Tabla 6 Persistencia de Datos*

HU006	User Story
Como	Desarrollador
Quiero	La información de los usuarios del sistema almacenada de manera persistente.
Para	Hay que asegurar que los datos no se pierdan y puedan ser recuperados en cualquier momento.
Detalles adicionales	
Criterios de aceptación	
El sistema debe almacenar la información de los usuarios creados en archivos persistentes fuera de la carpeta del código fuente.	

*Tabla 7 Notificaciones*

HU007	User Story
Como	Estudiante
Quiero	Recibir información de fechas límite de entrega.
Para	Lograr cumplir con todas las actividades propuestas dentro del plazo estipulado.
Detalles adicionales	
La fecha límite depende de la actividad del learning path terminada antes.	
Criterios de aceptación	

El sistema manda una notificación de la fecha límite recomendada para la actividad asignada.



# Pruebas Unitarias

Para garantizar una experiencia confiable y robusta para sus usuarios (profesores y estudiantes). Para esto, se implementará una serie de pruebas unitarias que validen el correcto funcionamiento de cada módulo y componente del sistema. Estas pruebas estarán diseñadas para abarcar los siguientes aspectos clave:

## 1. Gestión de usuarios:

- Registro y autenticación:
  - Validar que los usuarios puedan registrarse correctamente con datos válidos.
  - Asegurar que la autenticación funciona para usuarios registrados y falla para credenciales incorrectas.
  - Verificar restricciones como unicidad de correos electrónicos y longitud de contraseñas.

## 2. Rutas de aprendizaje:

- Creación y edición:
  - Confirmar que los profesores pueden crear rutas de aprendizaje con actividades válidas.
  - Verificar que la edición de rutas actualiza correctamente la información almacenada.
- Asignación y progreso:
  - Validar que los estudiantes puedan inscribirse en rutas existentes.
  - Probar el registro del progreso al completar actividades.
  - Asegurar que el sistema refleja correctamente que una actividad se completó.

## 3. Actividades:

- Verificar que las actividades se están creando correctamente.
- Verificar que la edición de actividades funciona correctamente.
- Intentar crear una actividad sin que exista un Learning Path.
- Quices de verdadero/falso:
  - Verificar que los quices se configuren correctamente (preguntas válidas, opciones disponibles).
  - Validar la corrección automática de respuestas y la retroalimentación asociada.

#### 4. Sistema de retroalimentación:

- Confirmar que los estudiantes pueden enviar reseñas solo para actividades completadas.
- Probar que los profesores reciben las reseñas y que estas se almacenan de manera persistente.

#### 5. Persistencia de datos:

- Probar la carga y el guardado automático de datos en archivos:
  - Validar que los datos se guardan correctamente al realizar cambios (nuevos usuarios, rutas creadas, progreso actualizado).
  - Asegurar que los datos se cargan de manera íntegra al iniciar el sistema.

#### 6. Notificaciones:

- Validar que el sistema genera notificaciones relevantes para los usuarios.
- Probar que las notificaciones se actualizan correctamente al realizar acciones en el sistema.

#### 7. Interfaz de usuario en consola:

- Validaciones de entrada:
  - Probar que la consola rechaza entradas inválidas (como campos vacíos o valores fuera de rango).
  - Asegurar que los mensajes de error sean claros y orienten al usuario sobre cómo corregir su entrada.
- Interacciones:
  - Verificar que todas las opciones del menú funcionan correctamente y llevan al flujo esperado.
  - Probar que el flujo de navegación permite regresar y avanzar entre menús sin errores.

De forma más específica, se espera implementar para cada clase una serie de pruebas unitarias que garanticen su funcionamiento. A continuación, se presenta el tipo de prueba, y lo que se espera obtener a raíz de su implementación:

### Clase Actividad:

Funcionalidad	Método	Tipo de Assert usado
---------------	--------	----------------------

Establecer y obtener descripción	setDescripcion() / getDescripcion()	assertEquals
Establecer y obtener objetivo	setObjetivo() / getObjetivo()	assertEquals
Establecer y obtener nivel de dificultad	setNivelDificultad() / getNivel()	assertEquals
Establecer y obtener duración esperada	setDuracionEsperada() / getDuracionEsperada()	assertEquals
Establecer y obtener prerrequisitos	setPrerrequisitos() / getPrerrequisitos()	assertEquals
Establecer y obtener fecha límite	setFechaLimiteBasadaEnActividadA nterior() / getFechaLimite()	assertEquals
Establecer y obtener obligatorio/opcional	setObligatorioOpcional() / isObligatorioOpcional()	assertEquals
Verificar actividad completada	isCompletada()	assertTrue / assertFalse
Advertencia sobre prerrequisitos	advertenciaPrerrequisitos()	assertFalse
Obtener resultado de la actividad	obtenerResultado()	assertEquals
Modificar creador no permitido	setCreador()	assertThrows
Modificar suma de rating no permitido	setSumaRating()	assertThrows
Modificar nombre no permitido	setNombre()	assertThrows
Cálculo correcto del rating promedio	setRating() / getRating()	assertEquals
Suma acumulada de ratings	setSumaRating() / getSumaRating()	assertEquals
Agregar prerrequisito correctamente	agregarPrerrequisito()	assertTrue (contiene el prerrequisito agregado)
Manejo de prerrequisitos vacíos	agregarPrerrequisito()	assertFalse (prerrequisito no debería ser vacío)

## Clase Encuesta:

Funcionalidad	Método	Tipo de Assert usado
Obtener lista de preguntas	getPreguntas()	assertEquals
Agregar una pregunta	agregarPregunta()	assertTrue (contiene) / assertEquals (tamaño)
Formato de detalles de encuesta	obtenerDetalles()	assertTrue (cadena contiene elementos esperados)
Descripción (constructor)	getDescripcion()	assertEquals
Objetivo (constructor)	getObjetivo()	assertEquals

Verificar actividad completada	isCompletada()	assertTrue / assertFalse
Obtener resultado sobrescrito	obtenerResultado()	assertEquals
Inicializar con preguntas	Constructor	assertEquals (preguntas iniciales)
Validar preguntas vacías	Constructor	assertTrue (lista está vacía)
Agregar pregunta nula o vacía	agregarPregunta()	assertThrows (manejar excepciones)
Formato correcto del detalle	obtenerDetalles()	assertTrue (contienen valores esperados como preguntas, descripción, objetivo)
Detalle con lista vacía de preguntas	obtenerDetalles()	assertTrue

## Clase Examen

Funcionalidad	Método	Tipo de Assert usado
Obtener lista de preguntas	getPreguntas()	assertEquals
Agregar una pregunta	agregarPregunta()	assertTrue (contiene) / assertEquals (tamaño)
Formato de detalles del examen	obtenerDetalles()	assertTrue (cadena contiene elementos esperados)
Descripción (constructor)	getDescripcion()	assertEquals
Objetivo (constructor)	getObjetivo()	assertEquals
Verificar actividad completada	isCompletada()	assertTrue / assertFalse
Obtener resultado sobrescrito	obtenerResultado()	assertEquals
Inicializar con lista vacía	Constructor	assertTrue (lista está vacía)
Agregar pregunta nula o vacía	agregarPregunta()	assertThrows (manejar excepciones)
Longitud máxima de una pregunta	agregarPregunta()	assertFalse (rechazo de preguntas largas)

Formato correcto del detalle completo	obtenerDetalles()	assertTrue (detalles contienen valores esperados como preguntas, descripción, objetivo)
Detalle con lista vacía de preguntas	obtenerDetalles()	assertTrue (cadena contiene solo atributos heredados)
Resultado del examen completado	obtenerResultado()	assertEquals (mensaje esperado según estado de completado)

## Clase LearningPath

Funcionalidad	Método	Tipo de Assert usado
Modificar atributos editables	modificarLearningPath()	assertEquals (nuevos valores)
Fecha de modificación actualizada	modificarLearningPath()	assertNotEquals (fecha inicial ≠ final)
Agregar una actividad	agregarActividad()	assertTrue (actividad en lista)
Actualizar duración al agregar	actualizarDuracion()	assertEquals (suma de duraciones)
Eliminar una actividad	eliminarActividad()	assertFalse (actividad no en lista)
Actualizar duración al eliminar	actualizarDuracion()	assertEquals
Manejo de error al eliminar	eliminarActividad()	assertThrows o validar mensaje
Agregar feedback	agregarFeedback()	assertTrue (feedback en lista)
Calcular rating promedio	calcularRating()	assertEquals (promedio esperado)
Manejo de feedback vacío	calcularRating()	assertEquals (0.0 en lista vacía)
Obtener lista de actividades	getActividades()	assertEquals (lista esperada)
Obtener lista de feedbacks	getFeedback()	assertEquals (lista esperada)
Obtener detalles heredados	Métodos como getTitulo()	assertEquals (valores esperados)

Duración total	getDuracionTotal()	assertEquals (valor esperado)
Manejo de actividad duplicada	agregarActividad()	assertTrue (solo una instancia)
Manejo de feedback duplicado	agregarFeedback()	assertTrue (solo un feedback igual)
Error en eliminación no existente	eliminarActividad()	Validar mensaje o assertFalse
Inicializar con listas vacías	Constructor	AssertTrue
Inicializar atributos heredados	Constructor	assertEquals (valores iniciales)

## Clase Quiz

Funcionalidad	Método probado	Tipo de assert usado
Inicializar atributos correctamente	Constructor de Quiz	assertEquals, assertNotNull
Validar herencia y asignación del tipo (tipo)	Constructor de Quiz	assertEquals
Agregar una pregunta al quiz	agregarPregunta	assertEquals, assertTrue, assertNotNull
Calcular la nota obtenida correctamente	realizarQuiz	assertEquals
Determinar resultado (aprobado/reprobado)	realizarQuiz	assertEquals, assertTrue, assertFalse
Registrar progreso de actividad	realizar	assertEquals, assertNotNull
Verificar permisos de edición (creador correcto)	editar	assertTrue
Verificar permisos de edición (creador incorrecto)	editar	assertFalse
Verificar fecha límite se calcula correctamente	establecerFechaLimite	assertEquals

Recordar actividad antes de la fecha límite	RecordatorioActividad	assertTrue, assertNotNull
Agregar una actividad de seguimiento	agregarActividadSeguimiento	assertTrue, assertFalse
Recomendar actividad según progreso	recomendarActividad	assertNotNull, assertEquals
Clonar actividad y verificar atributos	clonarActividad	assertNotNull, assertEquals
Agregar y calcular promedio de reseñas	agregarReseña, calcularPromedioRating	assertEquals, assertNotNull

## Clase Tarea

Funcionalidad	Método probado	Tipo de assert usado
Inicializar atributos correctamente	Constructor de Tarea	assertEquals, assertNotNull
Validar herencia y asignación del tipo	Constructor de Tarea	assertEquals
Enviar tarea y registrar medio de entrega	enviarTarea	assertEquals, assertNotNull
Registrar progreso al realizar la tarea	realizar	assertNotNull, assertEquals
Verificar permisos de edición (creador correcto)	editar	assertTrue
Verificar permisos de edición (creador incorrecto)	editar	assertFalse
Cambiar descripción de la actividad	cambios	assertEquals
Cambiar objetivo de la actividad	cambios	assertEquals
Verificar manejo de opción inválida en edición	cambios	assertThrows, assertEquals
Establecer y verificar fecha límite	establecerFechaLimite	assertEquals

Recordar actividad antes de la fecha límite	RecordatorioActividad	assertTrue, assertNotNull
Agregar actividad de seguimiento	agregarActividadSeguimiento	assertTrue, assertFalse
Recomendar actividad según progreso	recomendarActividad	assertNotNull, assertEquals
Clonar actividad y verificar atributos	clonarActividad	assertNotNull, assertEquals
Agregar y calcular promedio de reseñas	agregarReseña, calcularPromedioRating	assertEquals, assertNotNull

## Clase RecursoEducativo

Funcionalidad	Método probado	Tipo de assert usado
Inicializar atributos correctamente	Constructor de RecursoEducativo	assertEquals, assertNotNull
Validar herencia y asignación de atributos específicos	Constructor de RecursoEducativo	assertEquals
Cambiar tipo de recurso	cambios	assertEquals
Cambiar enlace del recurso	cambios	assertEquals
Verificar manejo de opción inválida en edición	cambios	assertThrows, assertEquals
Verificar permisos de edición (creador correcto)	editar	assertTrue
Verificar permisos de edición (creador incorrecto)	editar	assertFalse
Marcar recurso como completado	realizar	assertTrue, assertEquals
Registrar progreso como aprobado	realizar	assertEquals, assertNotNull
Establecer y verificar fecha límite	establecerFechaLimite	assertEquals
Agregar y calcular promedio de reseñas	agregarReseña, calcularPromedioRating	assertEquals, assertNotNull



## Clase PreguntaMultiple

Funcionalidad	Método probado	Tipo de assert usado
Crear una pregunta con 4 opciones y 4 explicaciones	Constructor	assertEquals, assertNotNull
Lanzar excepción si hay menos/más de 4 opciones	Constructor	assertThrows
Mostrar la pregunta y opciones	mostrarYResolver	assertTrue, assertEquals
Validar entrada del usuario (opciones válidas 1-4)	mostrarYResolver	assertTrue, assertEquals
Manejar entrada inválida (fuera de rango o no numérica)	mostrarYResolver	assertThrows, assertTrue
Determinar respuesta correcta	mostrarYResolver	assertTrue, assertFalse
Mostrar explicaciones relacionadas	mostrarYResolver	assertEquals

## Clase ProgresoActividad

Funcionalidad	Método probado	Tipo de assert usado
Inicialización correcta de atributos en el constructor	Constructor	assertEquals, assertFalse, assertNotNull
Calcular tiempo dedicado entre dos fechas	calcularTiempoDedicado	assertEquals, assertTrue
Marcar una actividad como realizada	marcarRealizada	assertTrue, assertEquals
Completar una actividad asignando un resultado	completarActividad	assertEquals
Manejo de fechas de inicio y fin	setFechaInicio, setFechaFin	assertEquals, assertNotNull

## Clase ProgresoPath

Funcionalidad	Método probado	Tipo de assert usado
---------------	----------------	----------------------

Inicialización correcta de atributos en el constructor	Constructor	assertEquals, assertFalse, assertNotNull
Agregar una actividad a la lista de realizadas	agregarActividadRealizada	assertTrue
Calcular el porcentaje de progreso en el Learning Path	calcularProgreso	assertEquals, assertNotEquals
Marcar el Learning Path como completado si se realizan todas las actividades obligatorias	marcarCompletado	assertTrue, assertNotNull
Calcular tasas de éxito y fracaso de las actividades realizadas	actualizarTasas	assertEquals, assertTrue

## Clase Usuario

Funcionalidad	Método probado	Tipo de assert
Inicialización de los atributos del usuario (nombre, correo, contraseña)	Constructor Usuario	assertEquals
Obtener el nombre del usuario	getNombre()	assertEquals
Obtener el correo del usuario	getCorreo()	assertEquals
Obtener la contraseña del usuario	getContrasena()	assertEquals
Ver los Learning Paths (Método abstracto)	verLearningPaths()	assertNotNull

Obtener el tipo de usuario (Método abstracto)	getTipoUsuario()	assertNotNull o assertEquals
Dar una reseña a una actividad	darReseñaActividad()	assertNotNull o assertTrue o assertEquals

## Clase Profesor

Funcionalidad	Método probado	Tipo de Assert
Creación de un Learning Path	crearLearningPath()	assertNotNull()
Creación de un Recurso Educativo	crearRecurso()	assertTrue()
Creación de una Encuesta	crearEncuesta()	assertEquals()
Asignación de una actividad a un Learning Path	asignarActividadALearningPath()	assertNotNull()
Verificación de un recurso agregado a Learning Path	obtenerRecursosPorLearningPath()	assertTrue()
Verificación de un Learning Path por título	obtenerLearningPathPorTitulo()	assertNotNull()
Modificación de un Recurso Educativo	modificarRecurso()	assertEquals()
Eliminación de un Learning Path	eliminarLearningPath()	assertTrue()
Asignación de tarea o proyecto	asignarTarea()	assertNotNull()
Verificación de la lista de tareas asignadas	obtenerTareasAsignadas()	assertTrue()

## Clase Estudiante

Funcionalidad	Método probado	Tipo de Assert
Verificar el tipo de usuario	getTipoUsuario()	assertEquals()

Verificación de Learning Paths inscritos	verLearningPaths()	assertNotNull()
Inscripción en un Learning Path	inscribirseEnLearningPath()	assertNotNull()
Verificación de inscripción en un Learning Path	inscripcion()	assertTrue()
Dar reseña a una actividad completada	darReseñaActividad()	assertTrue()
Selección de una actividad en un Learning Path	seleccionarActividad()	assertNotNull()
Iniciar actividad (verificación de estado)	iniciarActividad()	assertTrue()
Verificación de prerequisites de una actividad	faltanPrerrequisitos()	assertTrue()
Realizar una actividad	realizarActividad()	assertTrue()
Solicitar recomendación de actividad	pedirRecomendacionActividad()	assertTrue()
Solicitar progreso de un Learning Path	pedirProgresoPath()	assertTrue()

## Pruebas de Integración.

Funcionalidad	Método usado	Tipo de Assert usado
Registrar profesor en el sistema	Registro.registrarProfesor	assertTrue (verificar que el profesor fue añadido correctamente)
Registrar estudiante en el sistema	Registro.registrarEstudiante	assertTrue (verificar que el estudiante fue añadido correctamente)
Iniciar sesión de un profesor	Registro.loginProfesor	assertEqual (verificar que devuelve la instancia del profesor correcto)
Iniciar sesión de un estudiante	Registro.loginEstudiante	assertEqual (verificar que devuelve la instancia del estudiante correcto)
Crear un Learning Path por un profesor	Profesor.crearLearningPath	assertIn (verificar que el Learning Path fue añadido a la

		lista del profesor y del sistema)
Inscribir estudiante en un Learning Path	Estudiante.inscribirseEnLearningPath	assertIn (verificar que el Learning Path está en la lista del estudiante)
Realizar una actividad (Encuesta)	Encuesta.realizar	assertTrue (verificar que la actividad fue completada correctamente y marcada como realizada)
Agregar reseña a una actividad	Estudiante.darReseñaActividad	assertTrue (verificar que la reseña fue añadida a la actividad)
Calificar una actividad por un profesor	Profesor.calificarActividad	assertEqual (verificar que el resultado de la actividad fue actualizado correctamente)
Actualizar progreso de un Learning Path	ProgresoPath.calcularProgreso	assertEqual (verificar que el porcentaje de progreso es correcto)
Añadir prerequisites a una actividad	Profesor.agregarPrerrequisitoActividad	assertIn (verificar que el prerequisite fue añadido correctamente)
Clonar una actividad	Profesor.clonarActividad	assertNotEqual (verificar que la actividad clonada es una instancia nueva y diferente)
Calcular promedio de rating de un Learning Path	LearningPath.calcularPromedioRating	assertEqual (verificar que el promedio calculado es correcto)
Registrar un usuario (Profesor o Estudiante)	Registro.registrarProfesor, Registro.registrarEstudiante	assertIn (verificar que el usuario fue añadido a la lista correspondiente)
Crear un Learning Path	Profesor.crearLearningPath	assertIn (verificar que el Learning Path fue añadido al sistema y al profesor)
Crear actividades	Profesor.crearActividad	assertIn (verificar que la actividad fue añadida al Learning Path correspondiente)

Registrar progreso en un Learning Path	ProgresoPath.calcularProgreso	assertEqual (verificar que el progreso y las tasas de éxito/fracaso son correctas)
Agregar reseñas de actividades	Reseña.hacerReseña	assertIn (verificar que la reseña fue añadida a la lista de la actividad)
Recomendar actividades	Actividad.recomendarActividad	assertEqual (verificar que la actividad recomendada corresponde al progreso actual)
Notificaciones de fechas límite	Actividad.RecordatorioActividad	assertTrue (verificar que la notificación se genera si corresponde)
Calificar actividades	Profesor.calificarActividad	assertEqual (verificar que la calificación fue actualizada correctamente)

## Diseño Consola.

### Menú Principal

El sistema debe presentar un menú principal que permita al usuario seleccionar su tipo de rol:

1. Profesor.
2. Estudiante.
3. Salir.

Al seleccionar una opción, se redirige al menú correspondiente para Profesores o Estudiantes.

### Diseño de la Consola para Profesores

#### Menú Principal de Profesores

1. Ver los Learning Paths existentes.
2. Crear un nuevo Learning Path.
3. Modificar un Learning Path existente.
4. Crear una nueva actividad.
5. Clonar una actividad existente.

6. Salir.

## Opciones específicas para Profesores

### 1. Ver los Learning Paths existentes\*\*

- Muestra una lista con los títulos de los Learning Paths creados por el profesor.
- Si no tiene Learning Paths creados, se muestra: **"No tiene Learning Paths creados. Puede crear uno seleccionando la opción 2 del menú principal."**

### 2. Crear un nuevo Learning Path

Solicita al usuario los siguientes datos:

- Título del Learning Path.
- Descripción breve.
- Objetivo general.
- Nivel de dificultad (Bajo, Medio, Alto).
- Duración estimada en horas.
- Confirmación: "Learning Path creado exitosamente."

### 3. Modificar un Learning Path existente

- Muestra los Learning Paths creados y permite seleccionar uno.
- Opciones para modificar:
  1. Cambiar el título.
  2. Actualizar la descripción.
  3. Modificar el objetivo.
  4. Cambiar el nivel de dificultad.
  5. Modificar la duración estimada.
  6. Agregar o eliminar actividades.
- Confirmación tras cada cambio: **"Learning Path actualizado exitosamente."**

### 4. Crear una nueva actividad

Solicita seleccionar el tipo de actividad:

1. Recurso educativo.
2. Tarea.
3. Quiz.
4. Encuesta.
5. Examen.

Le solicitan información adicional según el tipo de actividad:\*\*

**Recurso educativo:**

- Tipo de recurso (Ejemplo: Artículo, Video, PDF).
- Enlace del recurso.

**Tarea:**

- Medio de entrega (Ejemplo: Plataforma virtual, Correo electrónico).

**Quiz:**

- Nota mínima de aprobación (decimal).
- Preguntas: Se ingresan mediante la función `agregarPregunta`. Para cada pregunta:
  - Texto.
  - Cuatro opciones de respuesta.
  - Explicación de cada opción.
  - Identificación de la opción correcta.

**Encuesta:**

- Preguntas: Se ingresan mediante la función `agregarPregunta`. Para cada pregunta:
  - Texto.

**Examen:**

- Preguntas abiertas: Se ingresan mediante la función `agregarPregunta`. Para cada pregunta:
  - Texto.

Confirmación: \*"Actividad creada exitosamente."\*

**5. Clonar una actividad existente**

- Muestra una lista de actividades creadas por el profesor.
- Solicita seleccionar una actividad para clonar.
- Permite modificar atributos como nombre, descripción o nivel de dificultad.
- Confirmación tras la clonación: \*"Actividad clonada exitosamente."\*

**6. Salir**

Finaliza la sesión del profesor y regresa al menú principal.



# Diseño de la Consola para Estudiantes

## Menú Principal de Estudiantes

1. Ver Learning Paths disponibles.
2. Inscribirse en un Learning Path.
3. Ver progreso en Learning Paths.
4. Ver progreso en actividades.
5. Seleccionar y realizar una actividad.
6. Dar reseña de una actividad.
7. Salir.

## Opciones específicas para Estudiantes

### 1. Ver Learning Paths disponibles

- Muestra una lista de todos los Learning Paths disponibles en el sistema.
- Si no hay Learning Paths disponibles, muestra: `"No hay Learning Paths disponibles en este momento."`

### 2. Inscribirse en un Learning Path

- Solicita seleccionar un Learning Path de la lista.
- Confirmación: `"Inscripción al Learning Path 'NOMBRE_LP' realizada exitosamente."`

### 3. Ver progreso en Learning Paths

- Muestra una lista de los Learning Paths inscritos y su progreso:
- Título del Learning Path.
- Porcentaje completado.
- Estado (En curso o Completado).
- Ejemplo:

Learning Path: Introducción a Java

Progreso: 60%

Estado: En curso

### 4. Ver progreso en actividades

- Muestra una lista de actividades realizadas y en progreso:
- Nombre de la actividad.
- Estado (Completada/En progreso).
- Tiempo dedicado (en horas).
- Fecha de inicio y finalización.
- Ejemplo:

Actividad: Tarea 1 - Fundamentos

Estado: Completada

Tiempo dedicado: 2 horas

Fecha de inicio: 10-11-2024

Fecha de finalización: 11-11-2024

## 5. Seleccionar y realizar una actividad

- Muestra las actividades disponibles dentro del Learning Path seleccionado.
- Solicita seleccionar una actividad.
- Confirmación tras completarla: `*"Actividad 'NOMBRE_ACTIVIDAD' completada exitosamente."*`

## 6. Dar reseña de una actividad

- Solicita seleccionar una actividad de la lista de actividades completadas.
- Solicita:

Calificación (de 1 a 10).

-Comentario.

- Confirmación: `*"Reseña para 'NOMBRE_ACTIVIDAD' guardada exitosamente."*`

## 7. Salir

Finaliza la sesión del estudiante y regresa al menú principal.

## Observaciones

El diseño asegura una experiencia fluida y permite que tanto profesores como estudiantes accedan a las funcionalidades principales del sistema de manera interactiva y organizada.