

**Universidad Nacional de Misiones**

**Carrera: Analista en Sistemas de Computación**

**Materia:** Proyecto FInal

**Profesor**

**Trabajo final**

**Integrante**

| Foto | APELLIDO y Nombre | correo electrónico |
| --- | --- | --- |
|  | Aguayo, Martin Eduardo | martineduardoaguayo@gmail.com |

**Año lectivo:** 2023

**Propuesta del trabajo final:**

El proyecto lo solicitó un cliente que trabaja en una academia y administra a los que enseñan en la academia (instructores). Cada instructor trabaja en una determinada localidad de misiones con su propio curso de enseñanza y tiene a sus propios alumnos.

Los alumnos cada mes, semana, o día abonan una cuota dependiendo del curso que estén realizando. Los instructores de los cursos cobran las cuotas, registran nuevos alumnos o quitan alumnos, dan el ok para que el administrador certifique a los que ya terminaron y comienzan nuevos cursos.

El cliente nos solicitó una página web en la que se pueda hacer todo esto más automáticamente en el cual nuestra propuesta es poder

* Administrar a los instructores con sus respectivos datos y cursos que pueden realizar
* Otorgar roles a las personas registradas en la página web (si es un instructor es instructor, si es un alumno el rol es de alumno etc)
* Permitir que los instructores se registren en la página web
* Permitir que los alumnos se registren en la página web
* Crear nuevos cursos y asignarle los instructores a cargo
* Emitir certificados
* Cobrar cuotas mensuales, semanales, diarias dependiendo del curso a realizarse.
* Ver los cursos más realizados
* Ver la cantidad de alumnos en la academia
* Otorgar descuentos a las personas inscriptas en más de un curso
* etc…

# Planificación

## ***Planificación de Entrevistas***

Para el proyecto se realizará el relevamiento mediante entrevistas con el cliente ya que es la persona que más sabe cómo necesita que funcione la aplicación web que le servirá para el proyecto y haremos preguntas cerradas y abiertas.

***Planificación de Actividades***

| **Actividad** | **Fecha de inicio** | **Fecha de finalización** |
| --- | --- | --- |
| **Planificación** |  |  |
| * Análisis de requerimientos | 03/04/2023 | 05/04/2023 |
| * Identificación de objetivos | 06/04/2023 | 07/04/2023 |
| **Requisitos** |  |  |
| * Descripción del proyecto | 08/04/2023 | 09/04/2023 |
| * Identificación de problemas | 08/04/2023 | 09/04/2023 |
| **Análisis** |  |  |
| * Diseño de casos de uso | 10/04/2023 | 18/04/2023 |
| * Diseño de casos de uso extendidos | 19/04/2023 | 09/05/2023 |
| * Diseño de modelo de dominio | 10/05/2023 | 12/05/2023 |
| * Diseño de DSS | 13/05/2023 | 27/05/2023 |
| * Diseño de contratos | 28/05/2023 | 05/06/2023 |
| **Diseño** |  |  |
| * DSD | 06/06/2023 | 20/06/2023 |
| * Casos de uso reales | 21/06/2023 | 29/06/2023 |
| * Diagrama de clases | 29/06/2023 | 30/06/2023 |
| **Programación** |  |  |
| * Desarrollo de los módulos | 01/07/2023 | 31/08/2023 |
| **Pruebas** |  |  |
| * Desarrollo de las pruebas | 01/07/2023 | 31/08/2023 |
| **Documentación** | 03/04/2023 | 31/08/2023 |
| **Manual de usuario.** | 01/07/2023 | 31/08/2023 |
| **Duración total del proyecto en días:** | | 150 |

## **Estudio de Factibilidad**

## **Factibilidad operacional**

Tanto la dueña de la academia como los instructores han manifestado que están conscientes de que es necesario implementar un sistema informático mecanizado, esto de acuerdo a las entrevistas realizadas anteriormente.

La dueña de la academia cree factible el tener una página web con un sistema sólido ya que actualmente no tienen un mecanismo eficiente para administrar y gestionar todos los procesos en la academia.

El sistema está constituido por los siguientes componentes principales

* Usuarios
* Gestor de Base de datos
* Aplicación web
* Sistema operativo
* Hardware
* Manuales de Usuario
* Reportes
* Pruebas

**Usuarios:**

Se ha agrupado en conjunto a toda persona que va a interactuar de forma directa con el sistema final ya sean estos administradores del sistema o usuarios finales

| REQUERIDO | DISPONIBLE |
| --- | --- |
| Administrador del Sistema | Administrador del Sistema |
| Administrador de base de datos | Administrador de base de datos |
| Alumnos | Alumnos |
| Instructores | Instructores |

Tabla: Usuarios destinados a operar el sistema propuesto

**Sistema Gestor de Base de Datos:**

Para el manejo claro, sencillo y ordenado de los datos se ha determinado que un sistema gestor de bases de datos es la herramienta adecuada para convertir los datos en información.

| REQUERIDO | DISPONIBLE |
| --- | --- |
| Base de datos relacional | PostgreSQL |

Tabla: Gestor de Base de Datos para la operación del sistema propuesto

**Aplicación web:**

Debido a que el sistema propuesto es para utilizarlo en una página web ha permitido decidir que la aplicación se desarrolle en windows. Se especifica la tecnología del lado del servidor que tiene mayor importancia y además algunas tecnologías del lado del cliente.

| REQUERIDO | DISPONIBLE |
| --- | --- |
| Librería React | Librería React |
| Lenguaje de programación interpretado JavaScript | Lenguaje de programación interpretado JavaScript |
| Lenguaje de programación HTML | Lenguaje de programación HTML |
| Lenguaje de programación CSS | Lenguaje de programación CSS |
| Framework | Next js |
|  |  |

Tabla: Aplicación web para la operación del sistema propuesto

**Sistema Operativo:**

Para administrar los recursos del servidor, el servidor web y el gestor de la base de datos se debe utilizar un sistema operativo compatible con estas tecnologías. Para que los usuarios finales tengan acceso a la aplicación se debe disponer de un sistema operativo que sea familiar para la mayoría de usuarios.

| REQUERIDO | Disponible |
| --- | --- |
| Sistema operativo Window/Mac Os/Linux | Sistema operativo Window/Mac Os/Linux |

Tabla: Sistemas operativos para la implementación del sistema propuesto

**Hardware:**

El soporte necesario para que el sistema funcione correctamente se compone de los siguientes elementos.

| REQUERIDO | DISPONIBLE |
| --- | --- |
| Servidor Host en línea | Servidor Host en línea |
| Computador personal | Computador personal |

Tabla: Evaluación de hardware

Manuales de Usuario:

Aprender cada parte del nuevo sistema es un punto muy importante para que el sistema final sea utilizado de la mejor forma. Durante el periodo de transición del sistema nuevo se presentan muchas dudas y es necesario un manual que pueda ser útil tanto para usuarios principiantes como también para usuarios avanzados.

| REQUERIDO | DISPONIBLE | POSIBILIDAD DE GESTIONAR |
| --- | --- | --- |
| Ayuda integrada |  | Ayuda integrada |
| Manual Aplicación web: Academia A.L. |  | Manual Aplicación web: Academia A.L. |

Tabla:Ayuda técnica del sistema propuesto para facilitar su operación

Reportes:

La información proporcionada por un sistema informático es el producto final que cualquier nivel de la pirámide organizacional necesita para tomar decisiones.

| REQUERIDO | DISPONIBLE | POSIBILIDAD DE GESTIONAR |
| --- | --- | --- |
| Reportes detallados |  | Reportes detallados |
| Reportes consolidados |  | Reportes consolidados |

Características del Hardware y Software de la OGA

| ELEMENTO | SERVIDOR | PC (1) |
| --- | --- | --- |
| Memoria RAM | 4 GB | 256 MB |
| Disco Duro | 2 discos ATA de 70GB c/u | 120 GB |
| Procesador | 2 procesadores Intel XEON de 3.2 GHz | Intel Pentium IV, 3.6 GHz |
| Motherboard | Intel | Intel |
| Monitor | CRT 19” | Flat 15 “ LCD |
| Tipo Impresora | No | Inyección |
| Velocidad impresor | No | 20 páginas por min |
| Mouse, teclado | Si | No |
| CD-ROM | 56x Max | No |
| DVD-ROM | 16x Max | 16x (formatos +R / -R RW) |
| UPS | 15 KVA | 600 VA |
| Tarjeta de red | 4 tarjetas de 100/1000 Kbps | 10/100 Kbps |
| Acceso a Internet | AMNET de 1 Mbps | Si |
| Topología de Red | Estrella | Estrella |
| Velocidad de Red | 1000 mbps | 100 mbps |
| Cable de Red | UTP categoría 6 | UTP categoría 5e |

Software actual:

| TIPO DE PROGRAMA | SERVIDOR | PC |
| --- | --- | --- |
| Sistema Operativo | Red Hat Linux v4.2 | WINDOW 11 |
| Lenguaje de Programación | JavaScript | N/A |
| SGBD18 | MySQL v5.1 | N/A |
| Servidor Web | Xampp 8.1.6 | N/A |

**Conclusión de Factibilidad Operativa:**

Se concluye que operativamente el desarrollo del proyecto es factible, ya que como desarrollador cuento con la tecnología informática necesaria para la puesta en operación del sistema de información propuesto.

## **Factibilidad Técnica**

Mediante esta factibilidad se establece si el sistema propuesto puede desarrollarse con los recursos técnicos con que cuenta el equipo de desarrollo; esto se hace considerando la disponibilidad de los recursos existentes en términos de hardware, software y recurso humano, o sea la existencia de la tecnología y el conocimiento necesario para establecer que sea factible técnicamente el desarrollo del proyecto.

**Sistema Operativo**

Este elemento es de los más importantes ya que debe cumplir con las características de estabilidad, administración, velocidad, facilidad de uso, seguridad, multiusuario y escalabilidad para soportar la instalación del sistema informático y a la vez brindar velocidad de conexión a las bases de datos y seguridad a los usuarios. El sistema operativo con el que vamos a iniciar que cumple con las características necesarias e indispensables para el buen funcionamiento del sistema propuesto es Window 11..

**Lenguaje de Desarrollo**

El lenguaje de desarrollo debe cumplir con las siguientes características:

* Soporte a gran cantidad de bases de datos
* Facilidad de desarrollo de sistemas
* En continua mejora
* Fácil de administrar
* Estable y ampliamente usado en ambiente web,

A continuación se presentan los diferentes lenguajes de desarrollo que cumplen con las características arriba mencionadas

* PHP
* JavaScript
* Python
* Java
* C#

**Sistema Gestor de Base de Datos:** Este es un factor muy importante ya que determinará la manera en que se guardará la información, la velocidad de procesamiento, respaldo de los datos y la seguridad.

El sistema gestor de base de datos debe cumplir con las siguientes características:

* Estable
* Seguro
* Escalable
* Soporte de grandes cantidades de información
* Conexión con diferentes lenguajes de programación vía ODBC
* En continua mejora

A continuación se presentan bases de datos que cumplen con las características arriba mencionadas

* MySQL
* Oracle
* PostreSQL

**Características del Hardware Disponible para el Desarrollo**

Las características de los equipos de cómputo con que se dispone actualmente para el desarrollo del sistema informático, se muestran a continuación:

| EQUIPO | ELEMENTO | CAPACIDAD |
| --- | --- | --- |
| Computadora de Escritorio | Memoria RAM  Disco Solido NVME  Disco SSD  Disco Duro  Procesador | 16GB (3600 MHZ)  240GB  240GB  500 Gb  INTEL CORE I7 9700 4.0 GHZ |

Con lo anterior podemos decir que el equipo de desarrollo cuenta con las tecnologías lo suficientemente robustas y maduras para desarrollar y soportar la aplicación, además está disponible para realizar el proyecto.

**Experiencia y Conocimiento del Equipo de Desarrollo**

El Recurso Humano, experiencia y conocimientos del equipo de desarrollo se especifican a continuación en el caso de que sea más de un desarrollador y no para el trabajo final de ASC

| RECURSO HUMANO | Project Manager  Diseñador  Desarrollador backend(O desarrollador Full Stack)  Desarrollador front end (O desarrollador Full Stack)  Equipo de Marketing |
| --- | --- |
| EXPERIENCIA | Desarrollo de Sistemas.  Análisis y Diseño de sistemas  Administración de proyectos informáticos.  Programación de sistemas  Trabajo en equipo. |
| CONOCIMIENTOS | Lenguajes de programación en ambiente Web.  Manejador de Base de Datos SQL.  Base de Datos Relacionales  Unified Modeling Language UML |

En la tabla anterior se detalla el recurso humano del que se dispone para el desarrollo del proyecto

**Conclusión de Factibilidad Técnica**

Se cuenta con el equipo necesario para el desarrollo del sistema informático tanto en hardware como en software, así mismo el equipo de desarrollo está capacitado ya que poseen los conocimientos y experiencia necesarios para que el desarrollo de cada una de las etapas se realice de manera satisfactoria, brindando los resultados esperados. Por lo cual se concluye que el desarrollo del Sistema Informático para la aplicación Asistente de Estacionamiento es técnicamente factible.

# **Relevamiento**

Para el proyecto que lleve adelante decidí realizar la recolección de información a través de entrevista cara a cara con el cliente,

Para la recolección de datos provenientes de los clientes decidí la utilización del método entrevista, que me permitió la correcta comprensión y delimitación de las características y funcionalidades planteadas por el cliente.

***Conclusiones del Relevamiento. Determinación de problemas, necesidades / requerimientos.***

Las conclusiones lógicas a las que podemos abordar luego del relevamiento realizado son:

* La necesidad de contar con un sistema de respuesta óptimo que permita un el acceso fácil y rápido a las funcionalidades de la aplicación.
* La necesidad de brindar un sistema que cumpla con la seguridad y eficiencia para el manejo de los datos ingresados en él.
* La necesidad de contar con un sistema bien optimizado para lograr el ahorro de recursos computacionales.

# **Análisis**

Análisis y Especificación de Requisitos conforme ERS 830

## **Descripción del escenario**

Tema: **Academia A.L.**

Descripción: Actualmente el cliente (dueña de una Academia de capacitaciones) gestiona todos los cursos, instructores que imparten dichos cursos y a los alumnos de cada curso mediante papel,wsp u otros medio de contacto. Los alumnos compran los cursos mediante publicaciones en facebook, instagram y generalmente se comunican por whatsapp o facebook para informarse más sobre los cursos que hay, los precios, duración, ubicación etc. Registra manualmente a todos los alumnos en un papel, los cursos, los pagos, crea y registra uno por uno los certificados para cada alumno.

Se busca poder hacer un proceso más fácil, productivo y satisfactorio para poder realizar dicha gestión y por supuesto también poder automatizar los pasos a realizar..

## **Objetivos del sistema**

- El producto software a desarrollar se denomina: “ACADEMIA A.L.” en el cual nos solicitan una aplicación web para que tanto:

**-** Sus instructores (profesores) puedan crear sus cursos y anotar a sus alumnos o que sus mismos alumnos se anoten.

- Los alumnos pueden ver la información de sus cursos, las fechas en la que se dictan, los próximos cursos a salir en el futuro que resulte de su interés etc.

- También que la propia persona dueña de la academia (administrador) pueda administrar y tener un control de todos los cursos, instructores y de los alumnos para poder emitir los certificados al finalizar dichos curso, esto se realizaría cuando un alumno haya aprobado un curso y el instructor lo apruebe y se lo dé por finalizado.

## **Requisitos Funcionales**

* Modulo de Administrador
  + El sistema debe permitir que un administrador elimine un curso.
  + El sistema debe permitir que un administrador cree un curso y asocie a un instructor al curso.
  + El sistema debe permitir que un administrador agregue a un alumno a un curso
  + El sistema debe permitir que un administrador administre los feedbacks asociados a un curso
  + El sistema debe permitir que un administrador administre los FAQS del sistema
  + El sistema debe permitir que un administrador cree un usuario y le asocie un rol.
  + El sistema debe permitir que un administrador elimine a un usuario del sistema.
  + El sistema debe permitir que un administrador edite a un usuario.
  + El sistema debe permitir que un Administrador cargue un Certificado en el sistema asociado a un curso.
  + El sistema debe permitir que un Administrador otorgue roles a los usuarios.
* Modulo de Alumno
  + El sistema debe permitir que un alumno sea dado de alta en el sistema.
  + El sistema debe permitir que un alumno borre su cuenta en el sistema.
  + El sistema debe permitir que un alumno se inscriba a un curso si cumple con los requisitos del curso.,
  + El sistema debe permitir que un alumno deje su feedback de un curso finalizado.
  + El sistema debe permitir que un alumno haga preguntas a un chatbot en el sistema (predeterminadas o libres).
  + El sistema debe permitir que un alumno pague un curso.
  + El sistema debe permitir que un alumno se dé de baja de un curso.
  + El sistema debe permitir que un alumno se inscriba a más de un curso.
* Módulo de Instructor (Módulo de alta interno)
  + El sistema debe permitir que un Instructor cree un curso
  + El sistema debe permitir que un Instructor suba contenido en un curso.
  + El sistema debe permitir que un Instructor Elimine su curso.
  + El sistema debe permitir que un Instructor edite su Curso.
  + El sistema debe permitir que un Instructor cambie de rol a Alumno.
  + El sistema debe permitir que un Instructor vea los cursos disponibles
  + El sistema debe permitir que un Instructor edite sus datos.
  + El sistema debe permitir que un Instructor elimine a su usuario.
  + El sistema debe permitir que un Instructor dé de baja a un alumno en su curso.
  + El sistema debe permitir que un Instructor otorgue notas a sus alumnos en su curso.
* Módulo de Curso
  + El sistema debe permitir que un curso tenga una descripción.
  + El sistema debe permitir que un curso tenga una fecha de inicio y final.
  + El sistema debe permitir que un curso sea gratis o tenga un precio.
  + El sistema debe permitir que se elimine un Curso.
  + El sistema debe permitir que se cree un Curso.
  + El sistema debe permitir que un curso tenga Alumnos asociados.
  + El sistema debe permitir que un curso tenga un Instructor a cargo.
  + El sistema debe permitir que un Curso sea de poca duración o de larga duración.
  + El sistema debe permitir que un Curso sea sincrónico o asincrónico.
  + El sistema debe permitir que un curso sea Presencial-Semipresencial-Virtual.
  + El sistema debe permitir que un Curso tenga información de que cosas se necesitan para realizar la misma.
* Módulo de Certificado
  + El sistema debe permitir que se puedan subir certificados.
  + El sistema debe permitir asociar un certificado a una Categoría.
  + El sistema debe permitir otorgarle a un alumno que finaliza un certificado.
  + El sistema debe permitir diferenciar entre certificados provinciales y nacionales.
  + El sistema debe permitir que un certificado sea de la OME.
  + El sistema debe permitir que se elimine un modelo de certificado del sistema.
* Módulo de Clase (clase 1, clase 2 del curso “X”)
  + El sistema debe permitir que una clase tenga una descripción.
  + El sistema debe permitir que una clase tenga una fecha de inicio y final.
  + El sistema debe permitir que una clase tenga contenido asociado.
  + El sistema debe permitir que una clase tenga
* Módulo de Nota
  + El sistema debe permitir que un Instructor otorgue una calificación a un alumno inscripto en su curso.
  + El sistema debe permitir que un Instructor decida la modalidad de las notas en el sistema (1 nota, 2 notas, sin nota solo aprobado o desaprobado).
* Módulo de Feedback
  + El sistema debe permitir que un Alumno que realizó un curso y lo terminó deje un feedback del mismo.
  + El sistema debe permitir que un Administrador gestione los feedback recibidos de los alumnos de los cursos que finalizaron.
  + El sistema debe permitir que los Instructores puedan visualizar los feedback recibidos.
* Módulo de Pago
  + El sistema debe permitir que un Alumno pague el precio para acceder a un curso de pago en donde el monto fue establecido por el Instructor.
  + El sistema debe permitir que el pago sea por la api de mercadopago.

## **Requisitos no funcionales**

* Módulo de Auditoría
  + El sistema debe permitir que el módulo de auditoría registre todas las acciones realizadas por los usuarios incluyendo el inicio y cierre de sesión, creación, modificación y eliminación de la información almacenada.
* Módulo de Estadística
  + El sistema debe proporcionar herramientas de análisis y visualización de datos estadísticos para facilitar la comprensión de los mismos

# **Modelado de Análisis**

## ***Diagrama de casos de uso***

## Casos de uso extendido

## 

| Caso de Uso: Registrarse | |
| --- | --- |
| Actor/es: Alumno | |
| Objetivo: Registrarse en el sistema. | |
| Resumen: El Alumno se registra en la plataforma para poder iniciar sesión y acceder a sus funcionalidades | |
| Precondición: El Alumno no tiene una cuenta en el sistema; | |
| Poscondición: El Alumno queda registrado en la base de datos del sistema. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un alumno se va a registrar en el sistema. |  |
|
|
| 2.- El alumno selecciona la opción Registrarse; |  |
|  | 3.- El sistema le muestra todos los campos a rellenar con su información personal, correo electrónico y contraseña. |
| 4.- El alumno rellena todos los campos obligatorios y envía la información al sistema. |  |
|  | 5.- El sistema valida si los datos ingresados son correctos |
|  | 6- El sistema confirma el registro del Alumno y le indica que ya puede iniciar sesión con la cuenta que se creó previamente. |
| **Curso alternativo:** | |
| Paso 5: Algunos datos del registro son incorrectos o no se han ingresado. Volver al paso 4 o finalizar el caso de uso.. | |

| Caso de Uso: Iniciar Sesión. | |
| --- | --- |
| Actor/es: Usuario | |
| Objetivo: Iniciar sesión en el sistema. | |
| Resumen: El Usuario inicia sesión en el sistema para poder usar la aplicación con su cuenta. | |
| Precondición: El usuario no inició sesión en el sistema. | |
| Poscondición: El usuario inició sesión en el sistema. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un usuario quiere iniciar sesión en su cuenta. |  |
|
|
| 2.- El actor selecciona Iniciar sesión. |  |
|  | 3.- El sistema le pide que ingrese correo y contraseña |
| 4.- El usuario ingresa su usuario y contraseña. |  |
|  | 5.- El sistema valida si los datos ingresados son correctos y autentifica al actor con éxito. |
| 6.- El usuario se encuentra en la GUI de su cuenta. |  |
| **Curso alternativo:** | |
| Paso 5: Los datos del usuario, contraseña o ambos no son válidos. Se vuelve al paso 4 o finaliza el caso de uso. | |

| Caso de Uso: Visualizar cursos | |
| --- | --- |
| Actor/es: Alumno | |
| Objetivo: Visualizar cursos de su interés/disponibles | |
| Resumen: El alumno va a la sección de cursos para poder ver todo los estudios que ofrece el Sistema. | |
| Precondición: El alumno se encuentra logueado en el sistema. | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un alumno desea visualizar los cursos que tiene disponible la academia. |  |
|
|
| 2.- El alumno selecciona la opción Cursos. |  |
|  | 3.- El sistema le muestra todos los cursos disponibles en el sistema en los que se puede inscribirse, los que están en proceso y los que están finalizados. |
| **Curso alternativo:** | |

| Caso de Uso: Visualizar un curso | |
| --- | --- |
| Actor/es: Alumno, Instructores | |
| Objetivo: Visualizar un curso de su interés que esté disponible | |
| Resumen: El alumno o instructor dentro de la sección de cursos ingresa a un curso para poder obtener más información. | |
| Precondición: El alumno o instructor se encuentran logueado en el sistema. | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un alumno o instructor desea visualizar un curso de los que tiene disponible la academia. |  |
|
|
| 2.- Selecciona la opción Cursos. |  |
| 3.- Selecciona un curso de los cursos disponibles |  |
|  | 4- El sistema muestra la información asociada al curso. |
| **Curso alternativo:** | |

| Caso de Uso: Inscribirse en Curso | |
| --- | --- |
| Actor/es: Alumno | |
| Objetivo: Darse de alta en un curso | |
| Resumen: El alumno dentro de un curso selecciona la opción para adquirir el curso y completa los pasos requeridos. | |
| Precondición: El curso se encuentra disponible para el alumno y cumplen con los requisitos para poder acceder al curso. | |
| Poscondición: El alumno se da de alta en el curso. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un alumno ingresa a un curso para inscribirse. |  |
|
|
| 2.- El alumno selecciona el curso en el que desea inscribirse. |  |
|  | 3.- El sistema muestra la información detallada del curso y la opción para inscribirse. |
| 4- El alumno selecciona la opción para inscribirse |  |
|  | 5- El sistema le muestra todos los requerimientos para poder finalizar con la inscripción. |
| 6- El alumno completa los requisitos y avanza con la inscripción . |  |
|  | 7- El sistema valida los datos otorgados. |
|  | 8- El sistema confirma la inscripción y da de alta al alumno. |
| 9- El alumno recibe un cartel de confirmación de alta en el curso y puede acceder a la misma. |  |
| **Curso alternativo:** | |
| 8- El sistema no puede confirmar la inscripción y notifica que no se pudo realizar la misma. : | |

| Caso de Uso: Crear Curso | |
| --- | --- |
| Actor/es: Instructor | |
| Objetivo: Crear un nuevo curso en la plataforma | |
| Resumen: El Instructor crea un curso en el plataforma. | |
| Precondición:El instructor debe estar autenticado en la plataforma | |
| Poscondición:El nuevo curso queda registrado en la plataforma y disponible para que los estudiantes se inscriban en él. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1- El instructor ingresa a la plataforma y se dirige a la sección de creación de cursos. |  |
|
|
|  | 2- El sistema muestra un formulario de creación de cursos, el cual incluye los campos necesarios para la información del curso, como el nombre del curso, la descripción, la fecha de inicio y fin, la categoría, el precio, el número de cupos disponibles, entre otros. |
| 3- El instructor completa los campos del formulario con la información del curso requerida previamente. |  |
| 4. El instructor revisa la información del curso y confirma la creación del mismo. |  |
|  | 5. El sistema guarda la información del curso en la base de datos y muestra un cartel de que se creó el curso. |
| **Curso alternativo:** | |
| 3a. Si el instructor encuentra un error en la información del curso, puede modificar los campos del formulario y volver a confirmar la creación del curso. | |
| 5a. Si el sistema detecta un error en la información del curso, muestra un mensaje de error y regresa al paso 2 para corregir los datos del formulario. | |

## 

| Caso de Uso: Pagar Curso | |
| --- | --- |
| Actor/es: Alumno, Plataforma de Pago | |
| Objetivo: Realizar el pago de un curso en la plataforma | |
| Resumen: Este caso de uso extendido describe el proceso de pago de un curso en la plataforma de la academia, detallando las acciones que deben realizar el alumno y la plataforma de pago para llevar a cabo la transacción. | |
| Precondición: El alumno debe estar registrado y logueado en la plataforma, haber seleccionado el curso que desea pagar y contar con los medios de pago aceptados por la plataforma de pago | |
| Poscondición:El alumno queda inscripto en el curso y puede acceder al contenido del mismo. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1- El alumno selecciona el curso que desea pagar. |  |
|
|
| . | 2- La plataforma de pago muestra el detalle del curso y el monto a pagar.. |
| 3- El alumno selecciona el medio de pago y completa la información requerida (número de tarjeta, fecha de vencimiento, código de seguridad, etc.). |  |
|  | 5. La plataforma de la academia recibe la confirmación del pago y actualiza el estado del curso a "pagado" para el alumno. |
| **Curso alternativo:** | |
| 1a. Si el pago es rechazado, se muestra un mensaje de error y se permite al alumno volver a intentar el pago con otro medio de pago. | |
| 3a. Si el alumno selecciona pagar con tarjeta de crédito, se verifica la disponibilidad de límite en la tarjeta y se realiza la autorización del monto del curso. Si no hay disponibilidad, se muestra un mensaje de error y se permite al alumno volver a intentar el pago con otro medio de pago. | |
| 4a. Si la transacción no se puede procesar, se muestra un mensaje de error y se permite al alumno volver a intentar el pago con otro medio de pago. | |

| Caso de Uso: Realiza Feedback | |
| --- | --- |
| Actor/es: Alumno | |
| Actor/es Secundarios: Administrador, Instructor | |
| Objetivo: Permitir al alumno compartir su opinión y experiencia con el curso y/o instructor | |
| Resumen: El alumno, una vez finalizado el curso, puede dejar un comentario y una calificación sobre su experiencia en el mismo y con el instructor. El feedback puede ser revisado por el instructor y el administrador, y se mostrará públicamente en la página del curso siempre y cuando haya sido aprobado por el administrador. | |
| Precondición: El alumno debe haber finalizado el curso | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1 - El alumno accede a la página del curso |  |
|
|
| 2- El alumno deja un comentario sobre el curso y/o instructor. |  |
|  | 3-El feedback es enviado al sistema |
| 4- El feedback es revisado por el instructor y el administrador. | . |
|  | 5. Si el feedback es aprobado por el administrador, se muestra públicamente en la página del curso. |
| **Curso alternativo:** | |
|  | |

## 

| Caso de Uso: Darse de Baja | |
| --- | --- |
| Actor/es: Alumno | |
| Objetivo: Darse de baja de un Curso. | |
| Resumen: El alumno decide darse de baja de un curso por motivos personales. | |
| Precondición: El Alumno debe estar inscripto en el curso. | |
| Poscondición: El alumno ya no se encuentra inscripto en el curso | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1 - El alumno accede a la página del curso |  |
|
|
| 2- El alumno selecciona la Opción “Darse de baja del curso” |  |
|  | 3-El sistema le muestra un cartel preguntando si esta seguro que desea hacerlo. |
| 4- El alumno selecciona la opción “Sí estoy seguro” | . |
|  | 5. El sistema elimina del curso al alumno. |
| **Curso alternativo:** | |
| 4. El Alumno selecciona que “No desea darse de baja” y sigue inscripto en el curso con acceso al contenido del mismo. | |

## 

| Caso de Uso: Inscribe Alumnos en sus cursos | |
| --- | --- |
| Actor/es: Instructor | |
| Objetivo: Inscribir a sus alumnos en sus cursos. | |
| Resumen: El alumno no está familiarizado con la tecnología o no sabe cómo inscribirse en el curso por lo que el instructor le matrícula al alumno en el curso. | |
| Precondición: El Instructor debe tener un curso creado previamente. El Alumno debe tener una cuenta en la plataforma. | |
|  | |
| Poscondición: El alumno queda registrado en el curso. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1 - El Instructor inicia sesión en su cuenta en el sistema. |  |
|
|
| 2-El Instructor selecciona el curso en el que desea inscribir a un alumno. |  |
|  | 3- El sistema muestra la lista de alumnos inscritos en el curso. |
| 4- El Instructor selecciona la opción "Inscribir alumno". | . |
|  | 5- La plataforma muestra una lista de los alumnos en el sistema permitiéndole elegir cuál desea agregar en el curso. |
|  | 7- La plataforma actualiza la lista de alumnos inscritos en el curso con los datos del nuevo alumno. |
| **Curso alternativo:** | |

| Caso de Uso: Calificar Alumnos | |
| --- | --- |
| Actor/es: Instructor, Alumno | |
| Objetivo: Registrar la calificación de los alumnos en el curso que él está impartiendo. | |
| Resumen: El instructor califica al alumno en el curso que está impartiendo. Una vez asignada la calificación el  el sistema guarda la información. | |
| Precondición: Debe existir un curso con alumnos inscriptos. | |
| Poscondición: El alumno ya no se encuentra inscripto en el curso | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1 - El instructor accede al sistema. |  |
|
|
| 2- El instructor selecciona el curso que está impartiendo. |  |
|  | 3- El sistema muestra una lista de los alumnos inscritos en el curso. |
| 4- El instructor selecciona al alumno que desea calificar. | . |
|  | 5. El sistema muestra la sección de calificaciones para ese alumno. |
| 6 El instructor asigna la nota correspondiente. |  |
|  | 7- El sistema guarda la información de la calificación. |
|  | 8- El sistema notifica al alumno que la calificación ha sido asignada. |
| 9 El alumno puede ver su calificación en su perfil de estudiante. |  |
| **Curso alternativo:** | |
| 5a. Si el alumno no ha completado el trabajo/Evaluación que debe ser calificado, el instructor puede marcar al alumno como "ausente" y guardar la información. | |

## 

## 

## 

## 

| Caso de Uso: Administrar Feedbacks | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: Permitir al Administrador gestionar los feedbacks recibidos de los usuarios del sistema. | |
| Resumen: | |
| Precondición: | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- El Administrador selecciona la opción "Administrar Feedbacks" |  |
|
|
|  | 2- El sistema muestra una lista de los feedbacks recibidos ordenados por fecha de creación |
| 3.- El Administrador puede filtrar la lista de feedbacks por curso, nombres de estudiante o nombre de instructor y fecha de creación. |  |
|  | 4- El sistema filtra los feedback según lo que selecciono el Administrador |
| 5- El Administrador puede ver el detalle de un feedback seleccionándolo de la lista |  |
| 6.- El Administrador puede marcar un feedback como leído o no leído, eliminar un feedback, aprobarlo. |  |
|  | 7 - Si se elimina el feedback el sistema lo elimina del sistema, si se aprueba el sistema lo muestra en el sistema, si se marca como leído el sistema lo marca como leído. |
| **Curso alternativo:** En el paso 3, si no hay feedbacks registrados en el sistema, se muestra un mensaje indicando que no hay feedbacks disponibles. | |

## 

| Caso de Uso: Establece FAQs | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: Permitir al Administrador del sistema establecer las preguntas frecuentes (FAQs) y sus respectivas respuestas para que los usuarios puedan resolver sus dudas en el sistema. | |
| Resumen: El Administrador del sistema tiene la capacidad de agregar, modificar o eliminar preguntas frecuentes (FAQs) y sus respectivas respuestas en el sistema para brindar una mejor experiencia a los usuarios en la resolución de sus dudas. | |
| Precondición: | |
| Poscondición:La lista de preguntas frecuentes (FAQs) se actualiza con los cambios realizados por el Administrador del sistema. | |
| Curso típico de eventos | |
| Acción de los Actores | Respuestas del sistema |
| 1.- El Administrador del sistema inicia sesión en el sistema |  |
|
|
|  | 2- El sistema muestra la opción de "Establecer FAQs". |
| 3.- El Administrador del sistema selecciona la opción "Establecer FAQs" |  |
|  | 4- El sistema muestra la lista de preguntas frecuentes (FAQs) existentes en el sistema. |
| 5.-El Administrador del sistema tiene la opción de agregar una nueva pregunta frecuente y su respectiva respuesta o modificar/eliminar una pregunta frecuente existente. |  |
| 6- El Administrador del sistema guarda los cambios realizados. |  |
|  | 7- El sistema actualiza la lista de preguntas frecuentes (FAQs) con los cambios realizados. |

## 

| Caso de Uso: Crear Certificado | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: Permite al Administrador agregar modelos de certificados en el sistema. | |
| Resumen: El Administrador ingresa al sistema y selecciona “Certificados”. Luego, elige la opción “Cargar Modelo de Certificado” en la cual se le solicita la categoria del curso para el cual se destinará el modelo de certificado | |
| Precondición: | |
| Poscondición:El modelo de certificado cargado estará disponible para su uso en la emisión de certificados del curso correspondiente. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- El Administrador ingresa al sistema y selecciona “Certificados” en el menú principal. |  |
|
|
|  | 2- El sistema muestra las opciones de creación de certificados disponibles. |
| 3.-El Administrador selecciona la opción “Cargar Modelo de Certificado”. |  |
|  | 4- El sistema solicita al Administrador el nombre del curso para el cual se destinará el modelo de certificado. |
| 5.-El Administrador ingresa la categoría del curso y selecciona el archivo del modelo de certificado a cargar. |  |
|  | 6- El sistema valida y almacena el modelo de certificado en el sistema. |
| **Curso alternativo:** Si el Administrador no selecciona un archivo de modelo de certificado válido, el sistema mostrará un mensaje de error y solicitará que se seleccione un archivo válido | |

## 

| Caso de Uso: Emitir Certificado | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo:Emitir un certificado para el alumno que ha completado un curso satisfactoriamente. | |
| Resumen:El Administrador ingresa al sistema y selecciona “Certificados”. Luego selecciona la opción “ Emitir Certificado” en el cual obtiene una tabla de los alumnos aprobados, quc curso aprobaron y a que instructor corresponde dicho curso con la fecha del mismo | |
| Precondición: El modelo del certificado se encuentra cargado en el sistema | |
| Poscondición:El Administrador emite un certificado para el alumno seleccionado. | |
| Curso típico de eventos | |
| Acción de los Actores | Respuestas del sistema |
| 1.-El Administrador selecciona la opción “Certificados”. |  |
|
|
|  | 2-El sistema muestra una lista de opciones relacionadas a certificados. |
| 3.- El Administrador selecciona la opción “Emitir Certificado”. |  |
|  | 4- El sistema muestra una tabla con la lista de alumnos aprobados, el curso que aprobaron, el instructor que corresponde a dicho curso y la fecha del mismo. |
| 5.-El Administrador selecciona al alumno del cual desea emitir el certificado. |  |
|  | 6- El sistema genera y muestra el certificado en formato PDF para su descarga o impresión y enviará al perfil del alumno. |

## 

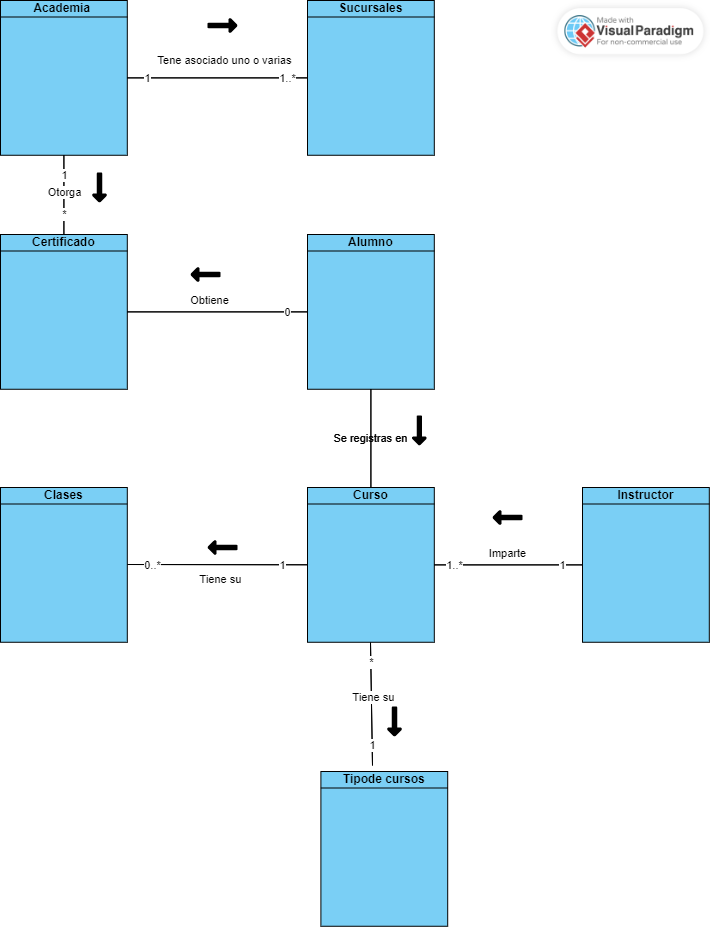
## 

| Caso de Uso: Otorgar Roles | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: Permite al Administrador otorgar roles a los usuarios registrados en el sistema. | |
| Resumen: El administrador ingresa al sistema y accede a la sección de administración de usuarios. Luego selecciona la opción de "Otorgar Roles" y busca al usuario al que desea otorgarle un nuevo rol. Después, selecciona el nuevo rol que se desea asignar y confirma la acción. | |
| Precondición:: El usuario tiene que estar registrado en el sistema. | |
| Poscondición:El usuario seleccionado ha sido asignado con un nuevo rol. El sistema actualiza la información de los roles asignados en la base de datos. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- El administrador ingresa al sistema y accede a la sección de administración de usuarios. |  |
|
|
| 2- El administrador selecciona la opción de "Otorgar Roles". |  |
|  | 3- El sistema muestra una lista de usuarios registrados. |
| 4- El Administrador busca al usuario al que desea otorgar un nuevo rol. |  |
|  | 5 - El sistema muestra una lista de los roles disponibles. |
| 6- El administrador selecciona el nuevo rol que desea asignar al usuario y confirma la acción. |  |
|  | 7 - El sistema actualiza los roles asignados al usuario y muestra un mensaje de confirmación al administrador |
| **Curso alternativo:** | |

## 

## Modelo de dominio

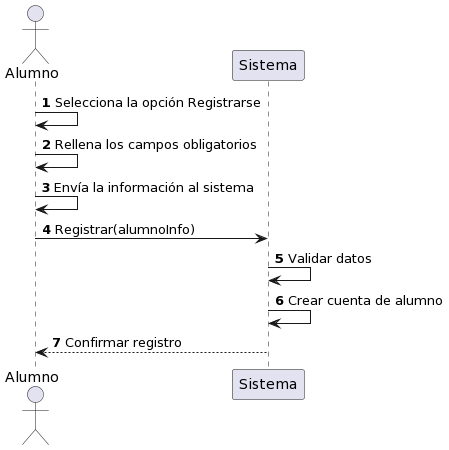
| **Conceptos candidatos** | **Conceptos idóneos** |
| --- | --- |
| Curso | Curso |
| Instructor | Instructor |
| Alumno | Alumno |
| Clases | Clases |
| Pago | Pago |
| Certificado | Certificado |
| Sucursales | Sucursales |
| Tipo de Curso | Tipo de Curso |
| Preguntas Frecuentes | Preguntas Frecuentes |
| Feedbacks | Feedbacks |



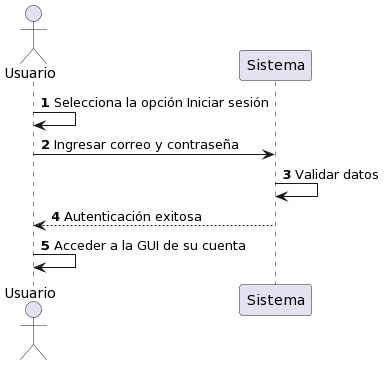
## 

## Diagramas de Secuencia de Sistema

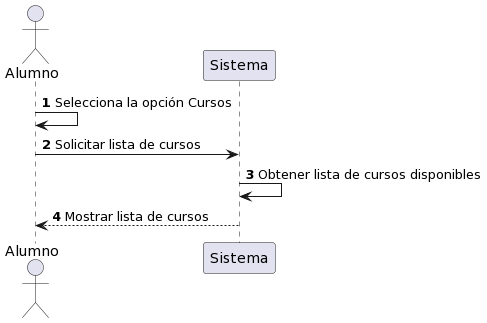
**DSS Registrarse**



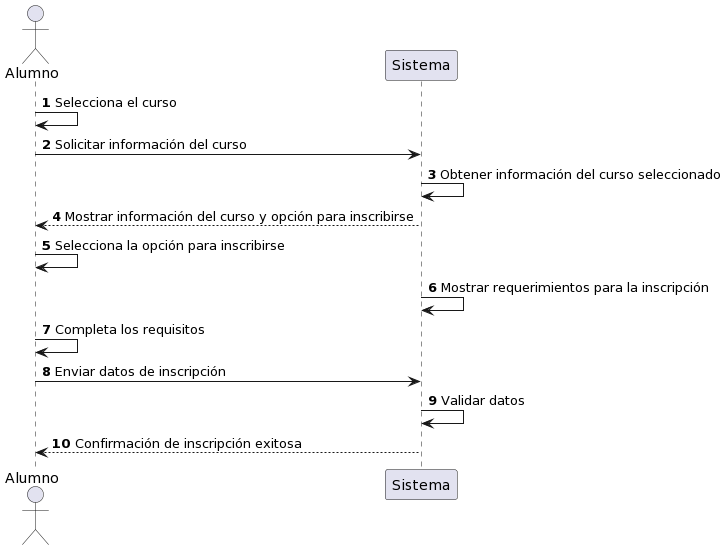
**DSS Iniciar Sesión**

****

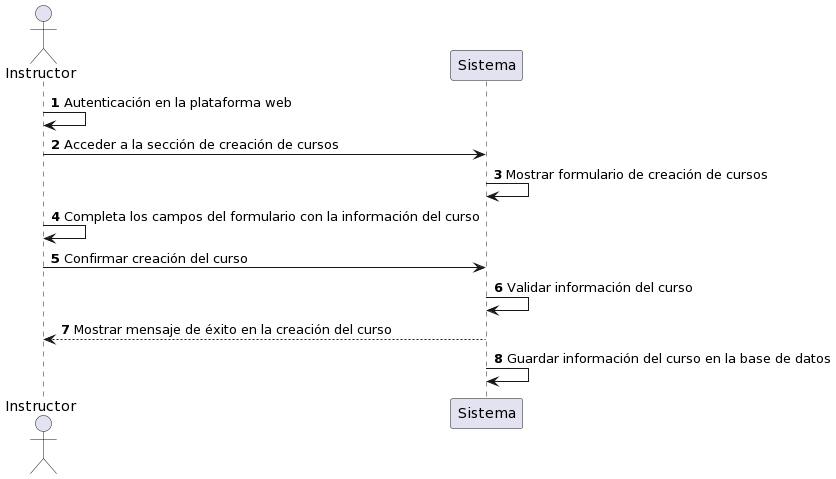
**DSS Visualizar Curso**

****

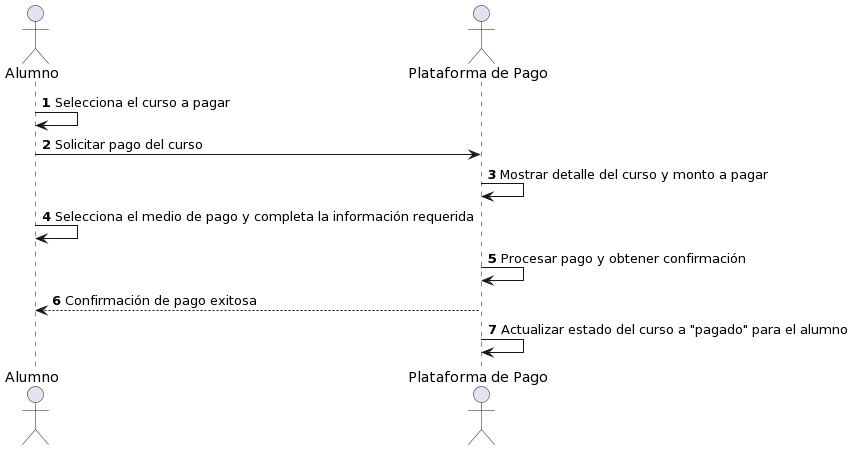
**DSS Inscribirse en un Curso**

****

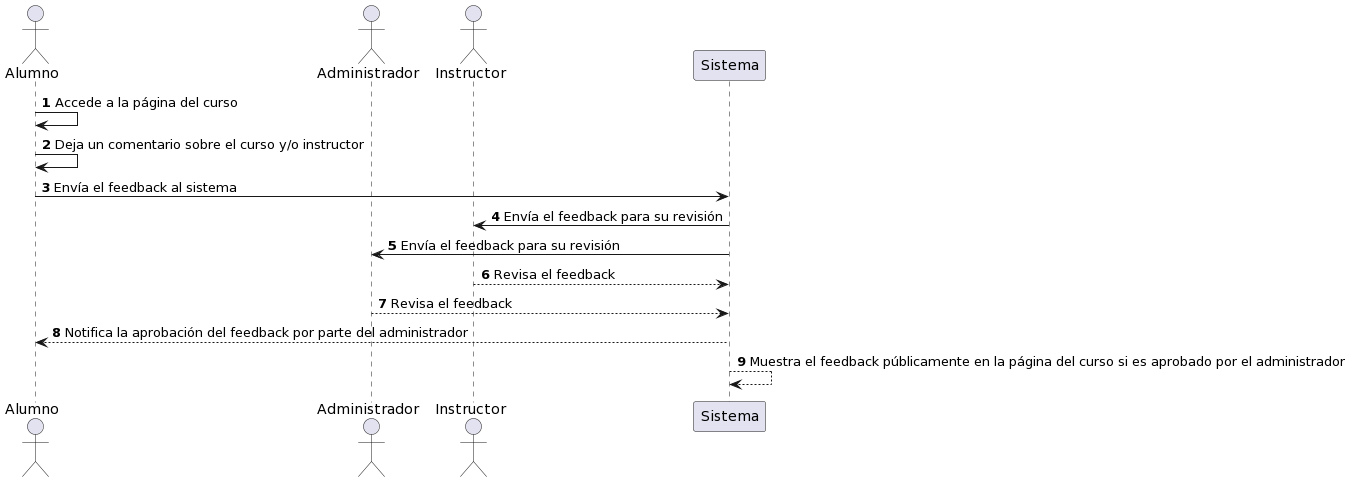
**DSS Crear Curso**

****

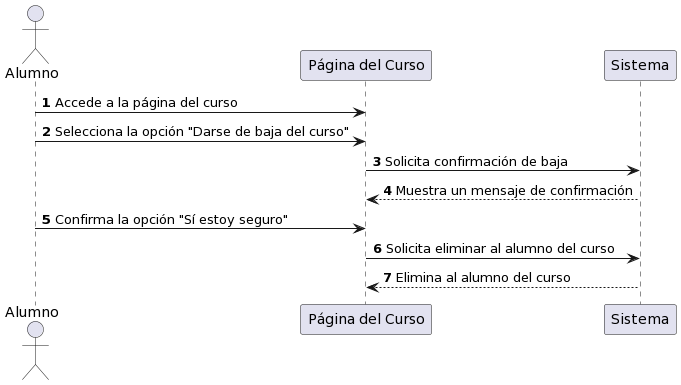
**DSS Pagar Curso**

****

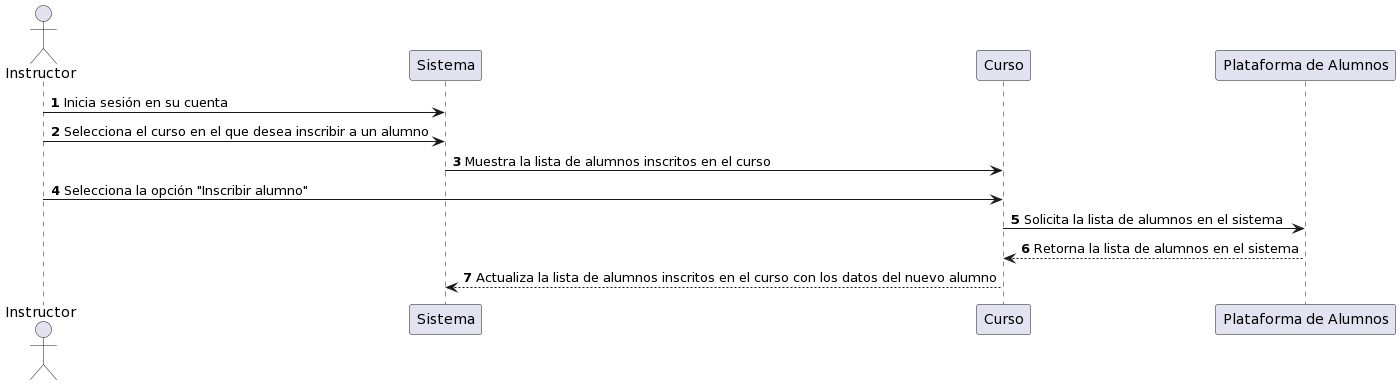
**DSS Emitir Feedback**

****

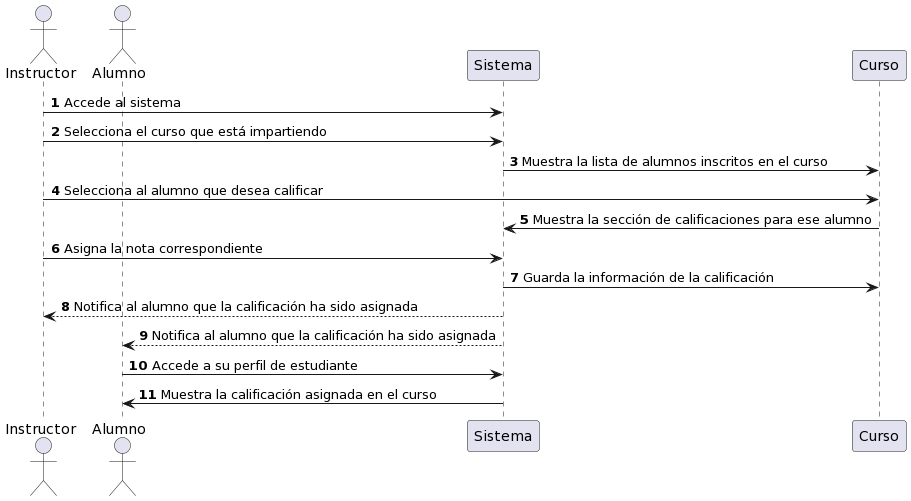
**DSS Darse de baja de un Curso**

****

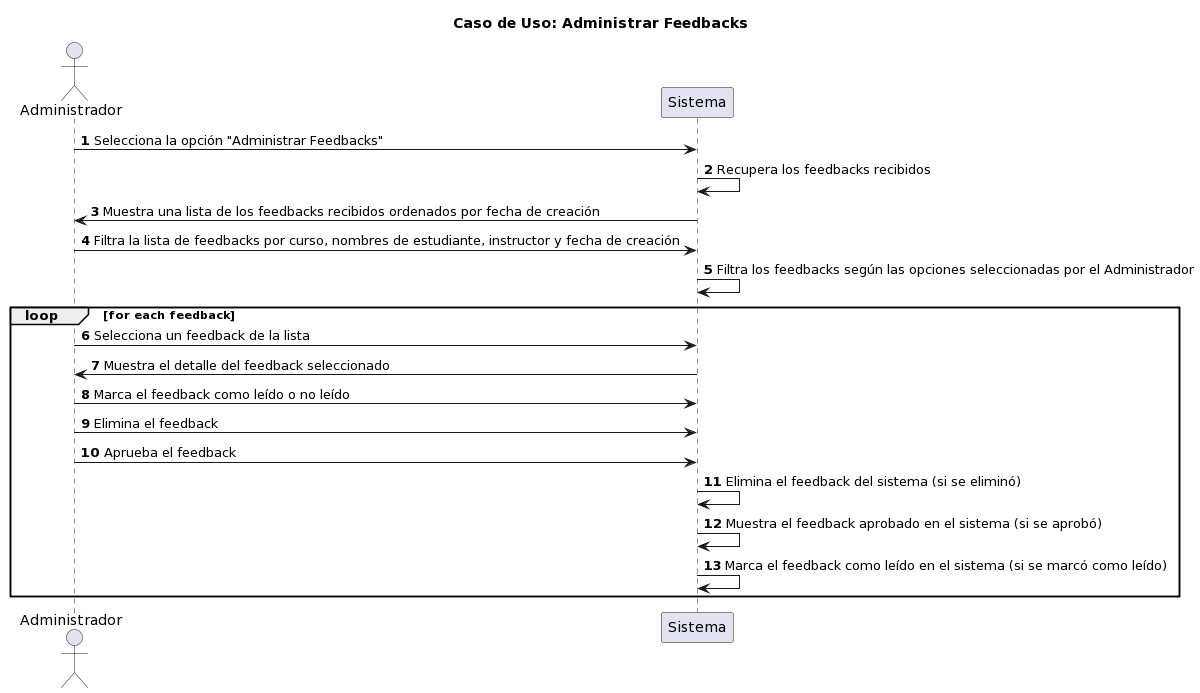
**DSS Inscribir Alumno Curso**

****

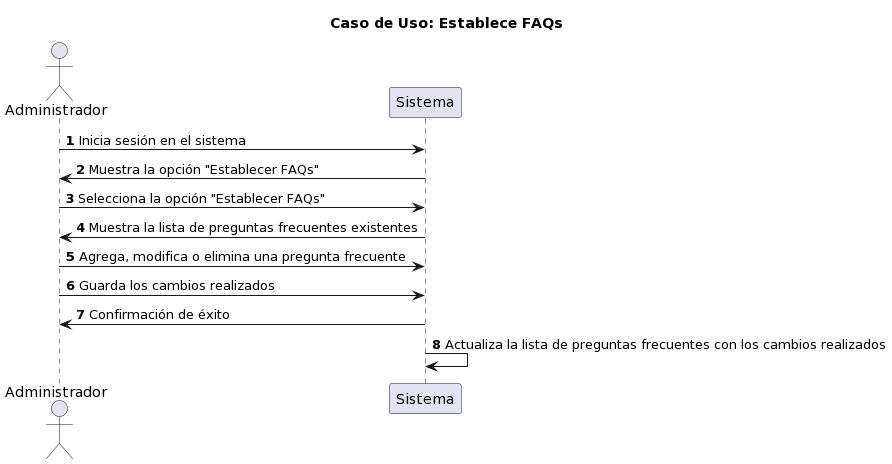
**DSS Calificar Alumno**

****

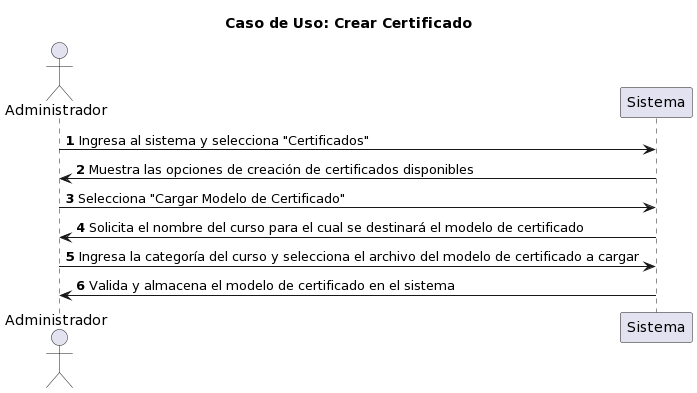
**DSS Administrar Feedback**

****

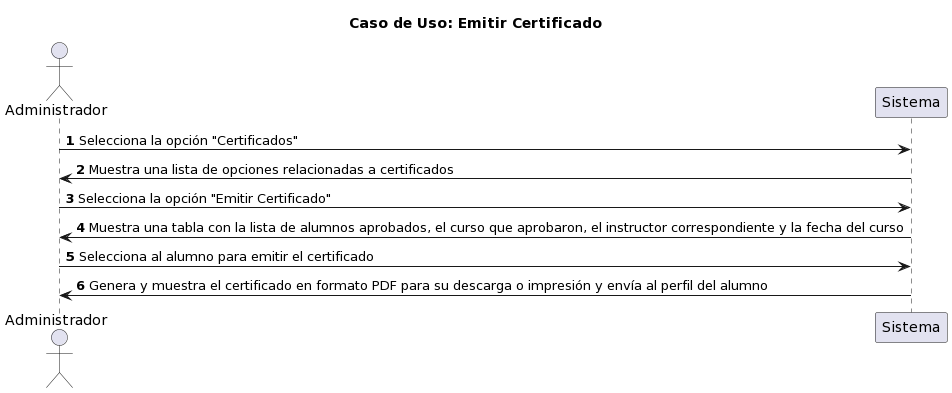
**DSS Administrar FAQs**

****

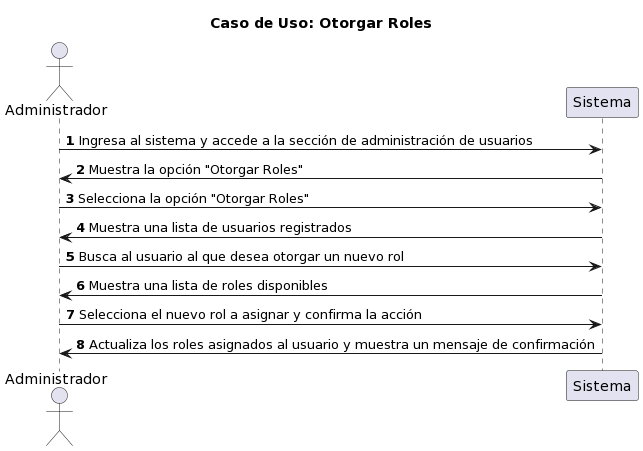
**DSS Cargar Certificado**

****

**DSS Emitir Certificado**

****

**DSS Otorgar Roles**

****

## Contratos

| Nombre | Inscribirse() |
| --- | --- |
| Responsabilidades | El alumno desea inscribirse en un curso |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Inscribirse en un Curso** |
| Notas | El curso puede ser gratis o de pago. |
| Excepciones | Si el curso es de pago el alumno para inscribirse tiene que realizar el pago del mismo para poder inscribirse. |
| Salidas | Inscripción exitosa al curso nombre “xxxxxx”.  Mensaje generado por las excepciones. |
| Precondiciones | * El alumno no se encuentra inscripto en el curso * EL curso tiene que tener cupos disponibles. |
| Poscondiciones | * Se asocia una inscripción al Curso * Se resta la cantidad de cupos disponibles para anotarse al curso |

| Nombre | PagarCurso( ) |
| --- | --- |
| Responsabilidades | Registrar Pago en el sistema. |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Pagar Curso** |
| Notas |  |
| Excepciones |  |
| Salidas | Pago realizado con éxito.  Mensaje generado por las excepciones. |
| Precondiciones |  |
| Poscondiciones | * Se asocia un pago al idUsuario del Alumno * Se asocia una inscripción al Curso * Se resta la cantidad de cupos disponibles para anotarse al curso |

| Nombre | CrearCurso() |
| --- | --- |
| Responsabilidades | El Instructor crea un nuevo curso en el sistema. |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Crear un Curso** |
| Notas |  |
| Excepciones | Si el Instructor no tiene Disponible la creación de un curso en particular recibirá una mensaje indicando la situación. |
| Salidas | * Curso creado con éxito * Mensaje generado por las excepciones. |
| Precondiciones | El instructor tiene disponible la creación de la categoría del curso. |
| Poscondiciones | * Se asocia el curso creado al instructor * Se asocia un curso creado en el sistema |

| Nombre | CargarCertificado() |
| --- | --- |
| Responsabilidades | El Administrador carga un modelo de certificado en el sistema destinado para una categoría. |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Crear Certificado** |
| Notas |  |
| Excepciones |  |
| Salidas | Mensaje de confirmación de carga del certificado. |
| Precondiciones |  |
| Poscondiciones | * Se asocia un certificado a una categoría de curso. * Se guarda en la base de datos el modelo de certificado |

| Nombre | EmitirCertificado() |
| --- | --- |
| Responsabilidades | El Administrador aprueba y emite un certificado para un alumno. |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Emitir Certificado** |
| Notas |  |
| Excepciones |  |
| Salidas | Mensaje de Emisión de certificado con éxito |
| Precondiciones | * El alumno tuvo que aprobar a ver aprobado y finalizado un curso. |
| Poscondiciones | * Se asocia un certificado a un Alumno |

| Nombre | DejarFeedback() |
| --- | --- |
| Responsabilidades | El Alumno al finalizar un curso deja su feedback del mismo. |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Emitir Feedback** |
| Notas |  |
| Excepciones |  |
| Salidas | Mensaje Feedback Realizado con éxito. |
| Precondiciones | * Tuvo que haber finalizado el curso del que se emitirá el feedback. |
| Poscondiciones | * Se asocia el feedback al id del alumno * Se asocia el feedback al curso realizado |

| Nombre | CalificarAlumno() |
| --- | --- |
| Responsabilidades | El Instructor ingresa a un curso que está impartiendo y califica a un alumno |
| Tipo | Sistema |
| Referencias cruzadas | **Caso de Uso: Califica Alumnos** |
| Notas |  |
| Excepciones |  |
| Salidas | Mensaje generado por excepciones. |
| Precondiciones |  |
| Poscondiciones | * Se asocia una nota al Alumno en el curso. |

# Diseño

## Casos de Uso Reales

## Diseño de Entradas y Salidas.

| Caso de Uso: Registrarse | |
| --- | --- |
| Actor/es: Usuario | |
| Objetivo: Registrarse en el sistema. | |
| Resumen: El Usuario se crea una cuenta para poder autenticarse en el sistema. | |
| Precondición: El Usuario no tiene una cuenta en el sistema. | |
| Poscondición: El Usuario se crea una cuenta en el sistema con usuario y contraseña. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un Usuario se va a registrar en el sistema. |  |
|
|
| 2.- El Usuario hace clic en el botón Registrarse. |  |
|  | 3.- El sistema le muestra todos los campos a rellenar con su información personal, email y contraseña. |
| 4.- El Usuario rellena todos los campos obligatorios y presiona el botón Registrarse, el cual envía la información al sistema. |  |
|  | 5.- El sistema valida si los datos ingresados son correctos y confirma el registro del alumno e indica que ya puede iniciar sesión con la cuenta que se creó previamente. |
| **Curso alternativo:** | |
| Paso 5: Algunos datos del registro son incorrectos o no se han ingresado. Volver al paso 4 o finalizar el caso de uso. | |

| Caso de Uso: Iniciar Sesión. | |
| --- | --- |
| Actor/es: Usuario. | |
| Objetivo: Iniciar sesión en el sistema. | |
| Resumen: El Usuario inicia sesión en el sistema para poder usar la aplicación con su cuenta. | |
| Precondición: El Usuario no inició sesión en el sistema. | |
| Poscondición: El Usuario inició sesión en el sistema. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un Usuario quiere iniciar sesión en su cuenta. |  |
|
|
| 2.- El Usuario hace click en el botón Iniciar sesión. |  |
|  | 3.- El sistema le pide que ingrese email y contraseña. |
| 4.- El Usuario ingresa su email y contraseña. |  |
|  | 5.- El sistema valida si los datos ingresados son correctos y autentifica al Usuario con éxito. |
| 6.- El Usuario se encuentra en la GUI de su cuenta. |  |
| **Curso alternativo:** | |
| Paso 5: Los datos del Usuario, correo electrónico, contraseña o ambos no son válidos. Se vuelve al paso 4 o finaliza el caso de uso. | |

| Caso de Uso: Crear Curso. | |
| --- | --- |
| Actor/es: Instructor | |
| Objetivo: Crear un curso en el sistema para que los alumnos tengan la posibilidad de inscribirse por el mismo. | |
| Resumen: El Instructor ingresa a su cuenta y se dirige a la sección de “Crear Curso”, completa todos los datos del mismo y crea un curso. | |
| Precondición: | |
| Poscondición: El curso creado está creado en el sistema. | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando un Instructor desea crear un curso. |  |
|
|
| 2.- El Instructor se dirige a la sección “Crear Curso” |  |
|  | 3.- El sistema le muestra LA GUI y el formulario con inputs a rellenar para crear el curso: Nombre del curso, Descripción, precio, horarios de cursada, fecha de inicio y finalización, días, imagen del curso, categoría, sucursal, que vas a aprender. |
| 4.- El Instructor completa todos los datos requeridos del formulario y hace click en crear curso. |  |
|  | 5.- El sistema valida si los datos ingresados son correctos y crea el curso en el sistema. |
| 6.- El Instructor recibe un mensaje de “Curso creado con éxito” y puede visualizar el mismo. |  |
| **Curso alternativo:** | |
| Paso 5: Los datos ingresados en el formulario del curso no son válidos . Se vuelve al paso 3 o finaliza el caso de uso. | |

| Caso de Uso: Ver Cursos | |
| --- | --- |
| Actor/es: Alumno, Instructor, Administrador | |
| Objetivo: Ver los cursos disponibles en el sistema.. | |
| Resumen: Los actores ingresan a su cuenta y hacen clic en “Ver cursos” para ver los cursos disponibles en el sistema. | |
| Precondición: Estar autenticado en el Sistema. | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando uno de los actores desea ver los cursos disponibles en el sistema. |  |
|
|
| 2.- El Actor se dirige a la sección “Ver Cursos” haciendo click en la misma. |  |
|  | 3.- El sistema le muestra LA GUI y los cursos disponibles en el sistema con su imagen, nombre, un botón para obtener más información del curso, precio, categoría y una opción de filtrado para filtrar por categoría, fecha, gratuitos o de pago, precio. |
| **Curso alternativo:** | |
| Paso 5: Si no existe ningún curso disponible en el sistema se muestra una pantalla de que no hay cursos disponibles. | |

| Caso de Uso: Administrar Categorías | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: CRUD para las categorías de los cursos | |
| Resumen: Permite al Administrador ver las categorías creadas en el sistema con todas las opciones de un CRUD. | |
| Precondición: Tener Rol de Administrador | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando el Administrador desea administrar las categorías. |  |
|
|
| 2.- El Admin se dirige a “Administrar Categorías” |  |
|  | 3.- El sistema le muestra un tabla con todas las categorías actuales en el sistema, opciones de filtrado, un botón para agregar una nueva categoría, otro botón para editar una categoría, otro botón para eliminar una categoría. |
| **Curso alternativo:** | |
| Paso 5: Si no existe ningúna categoría disponible en el sistema se muestra una pantalla de que no hay categorías creadas. | |

| Caso de Uso: Administrar Sucursales | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: CRUD para las sucursales de la Academia | |
| Resumen: Permite al Administrador ver las sucursales creadas en el sistema con todas las opciones de un CRUD. | |
| Precondición: Tener Rol de Administrador | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando el Administrador desea administrar las sucursales. |  |
|
|
| 2.- El Admin se dirige a “Administrar Sucursales” |  |
|  | 3.- El sistema le muestra un tabla con todas las Sucursales actuales en el sistema, opciones de filtrado, un botón para agregar una nueva sucursal, otro botón para editar una sucursal, otro botón para eliminar una sucursal/. |
| **Curso alternativo:** | |
| Paso 5: Si no existe ningúna Sucursal disponible en el sistema se muestra una pantalla de que no hay Sucursales creadas. | |

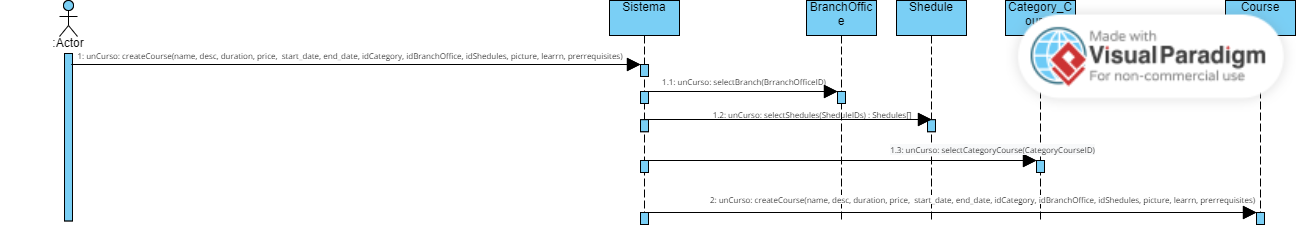
| Caso de Uso: Administrar Categorías | |
| --- | --- |
| Actor/es: Administrador | |
| Objetivo: CRUD para las categorías de los cursos | |
| Resumen: Permite al Administrador ver las categorías creadas en el sistema con todas las opciones de un CRUD. | |
| Precondición: Tener Rol de Administrador | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando el Administrador desea administrar las categorías. |  |
|
|
| 2.- El Admin se dirige a “Administrar Categorías” |  |
|  | 3.- El sistema le muestra un tabla con todas las categorías actuales en el sistema, opciones de filtrado, un botón para agregar una nueva categoría, otro botón para editar una categoría, otro botón para eliminar una categoría. |
| **Curso alternativo:** | |
| Paso 5: Si no existe ningúna categoría disponible en el sistema se muestra una pantalla de que no hay categorías creadas. | |

| Actor/es: Administrador | |
| --- | --- |
| Objetivo: CRUD para las sucursales de la Academia | |
| Resumen: Permite al Administrador ver las sucursales creadas en el sistema con todas las opciones de un CRUD. | |
| Precondición: Tener Rol de Administrador | |
| Poscondición: | |
| **Curso típico de eventos** | |
| **Acción de los Actores** | **Respuestas del sistema** |
| 1.- Este caso de uso comienza cuando el Administrador desea administrar las sucursales. |  |
|
|
| 2.- El Admin se dirige a “Administrar Sucursales” |  |
|  | 3.- El sistema le muestra un tabla con todas las Sucursales actuales en el sistema, opciones de filtrado, un botón para agregar una nueva sucursal, otro botón para editar una sucursal, otro botón para eliminar una sucursal/. |
| **Curso alternativo:** | |
| Paso 5: Si no existe ningúna Sucursal disponible en el sistema se muestra una pantalla de que no hay Sucursales creadas. | |

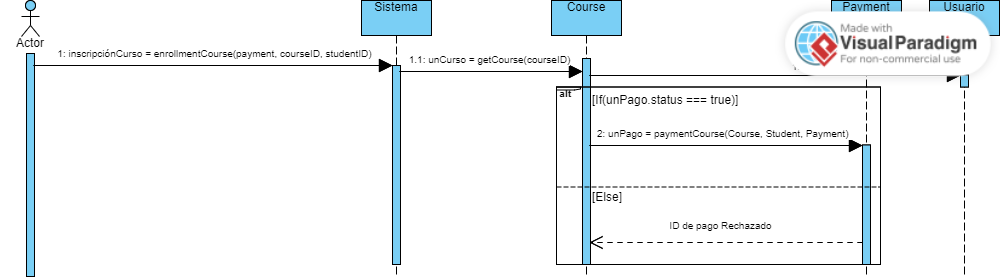
## 

## Diagramas de Secuencia de Diseño

**DSD** createCourse()

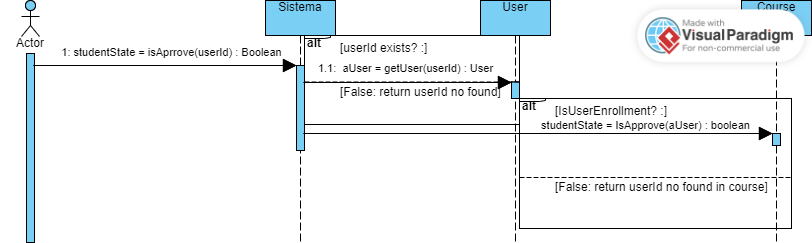


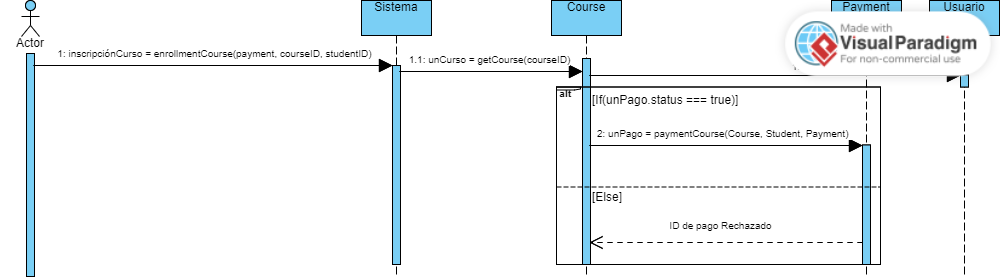
## 

**DSD** enrollmentCourse()

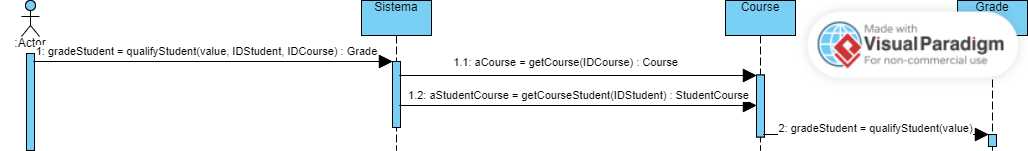
## 

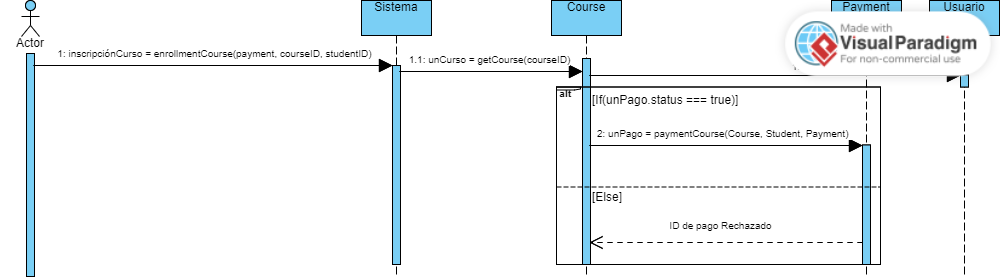
**DSD** aprroveStudent()



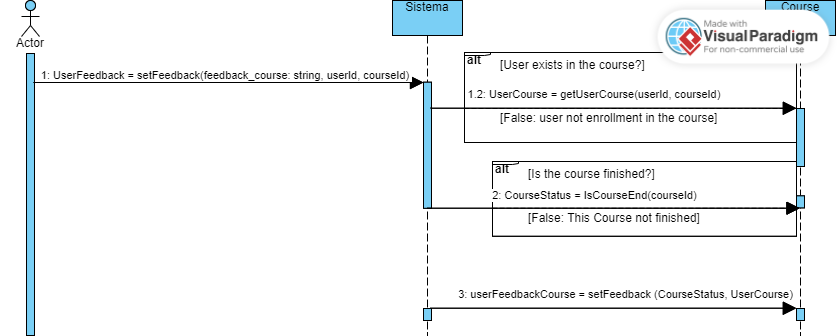


## 

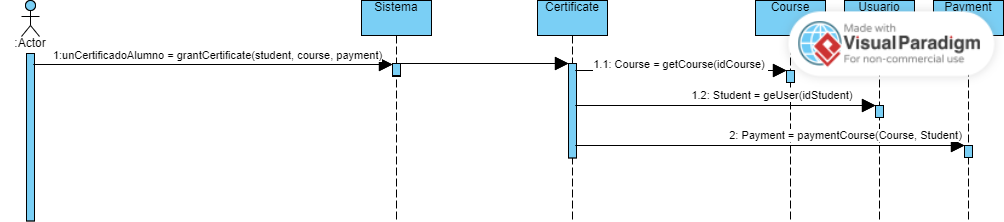
**DSD** qualifyStudent() 



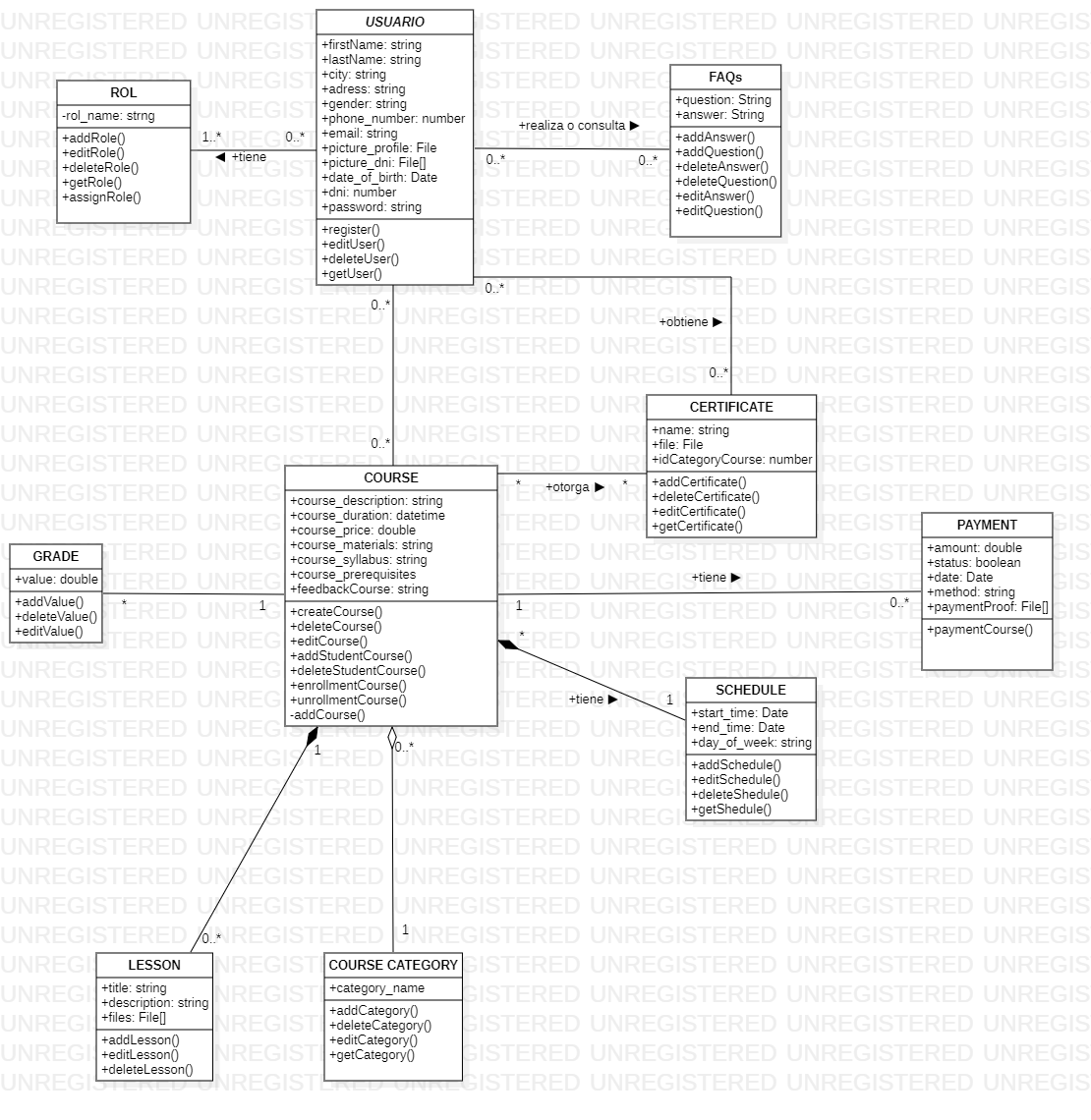
## 

**DSD** feedbackCourse() 

**DSD** grantCertificate()



## Diagrama de Clase



## 

## Diagrama Entidad Relación

## Modelo de Datos

**Links de Referencia/Documentación utilizada:**

- <https://next-auth.js.org/getting-started/example>

- <https://authjs.dev/reference/adapter/prisma>

- <https://react-hot-toast.com/docs>

- <https://docs.pmnd.rs/zustand/getting-started/introduction>

- <https://dev.to/franklin030601/usando-zustand-con-react-js-33le>

.- <https://react-hook-form.com/get-started/>

- <https://axios-http.com/es/docs/intro>

- <https://mui.com/material-ui/getting-started/overview/>

- <https://blog.logrocket.com/creating-responsive-sidebar-react-mui/>