**“Año del Bicentenario, de la consolidación de nuestra Independencia, y de la conmemoración de las heroicas batallas de Junín y Ayacucho”**



**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas**

**PROYECTO DE UNIDAD I**

**“SISTEMA DE MENTORÍA ACADÉMICA - AMS”**

Curso: *Calidad y Pruebas de Software*

Docente: *Mag. Ing. Patrick Cuadros Quiroga*

Integrantes:

***Huanca Merma, Gregory Brandon (2022073898)***

***Medina Quispe, Joan Cristian (2022074255)***

***Lira Alvarez, Rodrigo Samael Adonai (2019063331)***

**Tacna – Perú**

***2025***

| CONTROL DE VERSIONES | | | | | |
| --- | --- | --- | --- | --- | --- |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | GBHM, JMQ | NQC | NQC | 19/04/2025 | 2.0 |

**Proyecto AMS**

**“SISTEMA DE MENTORÍA ACADÉMICA”**

**Documento de Arquitectura de Software**

**Versión *2.0***

**ÍNDICE GENERAL**

[**ÍNDICE GENERAL 3**](#_heading=h.euuptpskndal)

[**1. Introducción 5**](#_heading=h.b370isyhcxm)

[**1.1. Propósito 5**](#_heading=h.5mq4y3nous2f)

[**1.2. Alcance 5**](#_heading=h.d8kmss2dpg35)

[**1.3. Definición, siglas y abreviaturas 6**](#_heading=h.v0c7sr4ya1wv)

[**1.4. Referencias 6**](#_heading=h.784zgx8tz8iz)

[**1.5. Visión General 6**](#_heading=h.8kigymo540ix)

[**2. Representación Arquitectónica 7**](#_heading=h.gkvy84346py3)

[**2.1. Escenarios 7**](#_heading=h.idiperqzqzkg)

[2.1.1. Escenario de Funcionalidad 7](#_heading=h.6ex49vedtyvw)

[**2.2. Vista Lógica 9**](#_heading=h.sqmzqu1zbedh)

[**2.3. Vista del Proceso 9**](#_heading=h.e7aujio00muw)

[**2.4. Vista del desarrollo 9**](#_heading=h.a43dqh5mq36y)

[**2.5. Vista Física 10**](#_heading=h.fc86fk1mhja3)

[**3.1. Disponibilidad 10**](#_heading=h.pvl1jdc8q4vw)

[**3.2. Seguridad 10**](#_heading=h.cqclicwoqwcf)

[**3.3. Adaptabilidad 10**](#_heading=h.a0r33nnalu1j)

[**3.4. Rendimiento 11**](#_heading=h.eyzs1xaqbb5z)

[**4.1. Requerimientos funcionales 11**](#_heading=h.a5tuc9hi2p4b)

[**4.2. Requerimientos no funcionales 11**](#_heading=h.i7lt09rpnrfk)

[5.1. Diagrama de Casos de Uso del Sistema Web AMS 12](#_heading=h.rz33a5vqjhz8)

[Comentario: Tenemos el diagrama de Casos de Uso del Sistema Escritorio AMS. 12](#_heading=h.i6c4ql3uhzdu)

[5.2. Diagrama de Casos de Uso del Sistema Escritorio AMS 13](#_heading=h.ddqz6gfu0x07)

[**6.1. Diagrama de Secuencia 14**](#_heading=h.pydnah1u6ptj)

[6.1.1. SD-UC-001 Autenticar usuarios 14](#_heading=h.tpimdsihv2f7)

[6.1.2. SD-UC-002 Registrar usuarios 14](#_heading=h.yot9a9kvfprt)

[6.1.3. SD-UC-003 Gestionar perfiles 15](#_heading=h.uhn1ings7z0q)

[6.1.4. SD-UC-004 Gestionar clases 16](#_heading=h.7ua4jhkavk0v)

[6.1.5. SD-UC-008 Gestionar calificaciones 17](#_heading=h.eid5126galn9)

[**6.2. Diagrama de Clases 18**](#_heading=h.b03hozabywn4)

[**6.3. Diagrama Contextual 19**](#_heading=h.1x4qwi7drcf)

[**7.1. Diagrama de Proceso Actual 19**](#_heading=h.gi4llqf11vbg)

[**7.2. Diagrama de Proceso Propuesto 20**](#_heading=h.vwuay0fvbzbw)

[**8.1. Diagrama de Contenedor 22**](#_heading=h.jbytjcxv01ih)

[**9.1. Diagrama de Paquetes 23**](#_heading=h.5volhsu2mfu4)

[9.1.1. Diagrama de Paquetes de Sistema Web AMS 23](#_heading=h.mssef5dzsr4t)

[Comentario: Tenemos el diagrama de paquetes del Sistema Web AMS. 23](#_heading=h.3x0kaa90hyax)

[9.1.2. Diagrama de Paquetes de Sistema Escritorio AMS 24](#_heading=h.pntkf0fcucb4)

[Comentario: Tenemos el diagrama de paquetes del Sistema Escritorio AMS. 24](#_heading=h.vtv8cscrbm38)

[**9.2. Diagrama de Componentes 25**](#_heading=h.25zegr5tcdwi)

[**Figura 11: Diagrama de Componentes. Fuente de Origen: Propia. 25**](#_heading=h.vmu7kpk2fnvn)

[**10.1. Diagrama Entidad Relación 26**](#_heading=h.f1r8vpwmiv66)

[**Figura 9: Diagrama Entidad Relación. Fuente de Origen: Propia. 26**](#_heading=h.ef8sh5elnwgs)

[**11.1. Escenario de Seguridad 27**](#_heading=h.snjszyklk2fd)

[**11.2. Escenario de Usabilidad 27**](#_heading=h.sz4eblqowxq1)

[**11.3. Escenario de Adaptabilidad 27**](#_heading=h.qds53nntems4)

[**11.4. Escenario de Disponibilidad 27**](#_heading=h.ilgk969r7qbv)

[**11.5. Otro Escenario 28**](#_heading=h.f8j4wbo955hr)

**PROYECTO DE   
SISTEMA DE MENTORÍA ACADÉMICA**

1. **Introducción**

## Propósito

Este documento de arquitectura tiene como propósito ofrecer una descripción detallada de la estructura del **Sistema de Mentoría Académica (AMS)**, diseñado para mejorar el rendimiento académico y reducir la tasa de deserción en la Universidad Privada de Tacna. Al documentar la arquitectura del sistema, se proporciona una base sólida que facilita la comprensión de su diseño y sus componentes principales. Además, este documento sirve como una guía para desarrolladores, arquitectos de software y administradores, permitiéndoles conocer los aspectos técnicos, las dependencias, y las interacciones clave entre los módulos del sistema. El propósito general es asegurar que el AMS sea escalable, seguro y adaptable a los cambios que puedan surgir en los requerimientos académicos y tecnológicos de la institución.

## Alcance

El AMS estará inicialmente disponible para la Facultad de Ingeniería de la Universidad, enfocándose en los estudiantes de la Escuela de Ingeniería de Sistemas que requieren apoyo académico adicional. Este documento cubre todos los aspectos de la arquitectura que permiten la operación del AMS: su diseño, representación lógica y física, vistas de procesos, y especificaciones de implementación. También se consideran los requisitos de disponibilidad, seguridad, adaptabilidad y rendimiento, necesarios para garantizar una experiencia de usuario fluida y eficaz. El sistema ha sido diseñado para ser modular y escalable, permitiendo su implementación futura en otras facultades e incluso en otras instituciones.

## Definición, siglas y abreviaturas

**AMS**: Sistema de Mentoría Académica, la plataforma que permite automatizar y gestionar mentorías en el contexto universitario.

**MVC**: Modelo Vista Controlador, un patrón arquitectónico que permite organizar la aplicación en tres capas independientes.

**CRUD**: Crear, Leer, Actualizar y Eliminar, las operaciones básicas de base de datos implementadas para gestionar datos en el AMS.

**JSP**: Java Server Pages, una tecnología utilizada en el AMS para crear páginas web dinámicas en el servidor.

## Referencias

Este documento se apoya en varios documentos técnicos y administrativos desarrollados previamente:

* Informe de Visión del Sistema de Mentoría Académica, Universidad Privada de Tacna (2025), que detalla el objetivo, alcance y visión general del sistema.
* Especificación de Requerimientos de Software (SRS) del AMS, que define los requisitos funcionales y no funcionales necesarios para la implementación.
* Informe de Factibilidad del AMS, que explora la viabilidad económica, técnica, operativa y social del proyecto.
* Informe Final de Proyecto del AMS, que resume los resultados y logros obtenidos durante el desarrollo inicial del sistema.

## 

## Visión General

El AMS es una plataforma web construida sobre la arquitectura MVC, que permite gestionar y automatizar el sistema de mentorías académicas en la universidad. Está diseñado para facilitar el proceso de asignación de mentores, programar sesiones de tutoría y realizar un seguimiento efectivo del progreso de los estudiantes. Además de ayudar a reducir la tasa de deserción, el sistema fomenta el desarrollo de una cultura de apoyo entre pares y liderazgo entre los estudiantes avanzados que actúan como mentores. El AMS también genera informes y métricas para ayudar a los administradores a tomar decisiones informadas sobre el progreso académico.

1. **Representación Arquitectónica**

## Escenarios

### Escenario de Funcionalidad

Necesidad: Se busca que los usuarios sean capaces de entrar en la página web en cualquier momento que ellos quieran o deseen.

Solución: Mantener la página web activa durante el tiempo las 24 horas, siendo posible su uso en ese intervalo de tiempo, en caso de no poder acceder también se puede llamar a soporte para acceder.

Justificación: Al ser una página web, debe estar disponible en todo momento.

* + 1. **Escenario de Usabilidad**

Necesidad: Se precisa que los usuarios puedan entender rápida e intuitivamente el uso de la página web.

Solución: Implementación de una interfaz sencilla y atractiva al ojo para que los usuarios puedan realizar su navegación desde el primer minuto.

Justificación: Al hacer que una interfaz sea fácil de manipular, provoca que los usuarios se sientan relajados de usarla y no alejarlos de la empresa.

* + 1. **Escenario de confiabilidad**

Necesidad: Se requiere que los usuarios tengan la confianza de realizar o usar cualquier acción o función dentro de la página y que posean una página web segura de usar o navegar para generar confianza en la misma.

Solución: Implementación de creación de cuentas de usuarios para acceder al sitio web que se encargará de proteger toda la información o datos de los clientes.

Justificación: El uso de las cuentas de usuarios proporciona cierto nivel de seguridad a cualquier sistema o página que necesite resguardarse de cualquier ataque o circunstancia, además de ser cómodo para los clientes de usar.

* + 1. **Escenario de rendimiento**

Necesidad: Se requiere que el sistema corra con normalidad de forma rápida y eficiente, cumpliendo con la carga de los procesos que requieran los clientes sin demora.

Solución: Correcta y buena implementación de los requerimientos establecidos sin ningún fallo, generando tranquilidad en las operaciones de los usuarios.

Justificación: Al implementarse de buena manera las funciones del sistema web, se evitan posibles errores que retrasen y ralentizan los procesos realizados dentro del sistema.

* + 1. **Escenario de mantenibilidad**

Necesidad: Se requiere que el sistema web sea capaz de mantenerse dentro de lo posible de manera sencilla en caso de cualquier problema y se adapte a las adversidades.

Solución: Revisar constantemente el funcionamiento de la página web en busca de posibles fallos que comprometan el sistema para los usuarios.

Justificación: Tratándose el proyecto sobre la creación de un nuevo sitio web y las mejoras e integraciones de funciones, se quiere dar resultados a la altura de lo establecido.

## Vista Lógica

La vista lógica del AMS se basa en la arquitectura MVC, que divide la aplicación en tres componentes principales: **Modelo**, **Vista** y **Controlador**. El **Modelo** representa los datos de la aplicación, donde se gestionan las entidades como usuarios, clases y reportes. La **Vista** corresponde a la interfaz de usuario desarrollada en JSP, que permite a los estudiantes, mentores y administradores interactuar con el sistema de manera intuitiva. El **Controlador** contiene la lógica de negocio y las reglas que regulan cómo los datos fluyen entre el modelo y la vista, asegurando que las operaciones de negocio se ejecuten de manera correcta y que los usuarios reciban respuestas rápidas y precisas.

## Vista del Proceso

El sistema automatiza diversos procesos, como la asignación de mentorías, la generación de reportes de seguimiento y el envío de notificaciones. Estos procesos son esenciales para mejorar la eficiencia y reducir el tiempo de respuesta en la asignación de clases de refuerzo. Por ejemplo, la asignación de mentorías sigue una secuencia lógica que evalúa la disponibilidad de mentores y empareja a los estudiantes en función de sus necesidades académicas. De esta forma, el sistema asegura que cada estudiante reciba el apoyo necesario sin intervención manual de los administradores.

## Vista del desarrollo

La elección de tecnologías de código abierto como **Java**, **JSP** y **MySQL** permite un desarrollo flexible y de bajo costo. Usando **NetBeans** como entorno de desarrollo (IDE) y **Apache Tomcat** como servidor de aplicaciones, el equipo de desarrollo cuenta con herramientas que facilitan el control de versiones, depuración y despliegue del AMS. La elección de un entorno basado en Java permite una integración fluida con sistemas empresariales y una comunidad de soporte activa, asegurando la sostenibilidad del desarrollo a largo plazo.

## Vista Física

El AMS se despliega en un servidor local de la universidad, con posibilidad de migración a una plataforma de alojamiento en la nube. Esta decisión asegura que el sistema sea accesible para estudiantes y mentores desde cualquier lugar con conexión a internet, optimizando la disponibilidad y redundancia del servicio. La arquitectura en la nube permite futuras mejoras y expansiones, tales como el aumento de la capacidad de procesamiento y almacenamiento.

1. **Objetivos y limitaciones arquitectónicas**

## Disponibilidad

El AMS está diseñado para operar de forma continua, permitiendo que los estudiantes y mentores accedan al sistema en cualquier momento. La arquitectura incluye un sistema de alertas y monitoreo que detecta fallas o interrupciones y permite recuperar el servicio rápidamente. Esto es crítico en un contexto académico, ya que asegura que los estudiantes no pierdan oportunidades de mentoría por problemas técnicos.

## Seguridad

La seguridad de los datos es una prioridad en el AMS, ya que maneja información sensible de los usuarios. Se implementan protocolos de autenticación segura, como el cifrado de contraseñas y control de acceso basado en roles, para proteger la integridad y privacidad de la información. Además, el sistema cumple con normativas de protección de datos, como la Ley de Protección de Datos Personales de Perú, lo que asegura la confidencialidad de la información en todo momento.

## Adaptabilidad

La arquitectura del AMS es modular, lo que permite realizar mejoras sin interrumpir el servicio. Esto es particularmente útil para futuras expansiones, como la adición de módulos de análisis predictivo de rendimiento, que identificarían a los estudiantes en riesgo antes de que su rendimiento baje significativamente. La adaptabilidad del sistema garantiza que pueda evolucionar junto con las necesidades académicas de la universidad.

## Rendimiento

El rendimiento del AMS es fundamental para ofrecer una experiencia de usuario satisfactoria. El sistema está optimizado para responder en menos de 2 segundos a las operaciones básicas de CRUD y soportar hasta 100 usuarios simultáneos sin afectar la velocidad. Esto se logra mediante consultas SQL eficientes y un uso adecuado de la memoria y los recursos del servidor.

1. **Análisis de Requerimientos**

## Requerimientos funcionales

**Autenticación y gestión de usuarios**: El sistema debe permitir el registro, autenticación y gestión de diferentes roles de usuario (estudiantes, mentores y administradores).

**Asignación y gestión de clases**: El AMS debe gestionar las clases de mentoría, asignando automáticamente mentores a estudiantes en función de criterios específicos y programando el uso de aulas disponibles.

**Sistema de notificaciones**: Notificaciones automáticas recordarán a los estudiantes y mentores sobre clases próximas, cancelaciones o cambios, mejorando la comunicación y asistencia.

**Generación de reportes**: El sistema debe generar informes de progreso y métricas académicas, facilitando el seguimiento del rendimiento estudiantil y la toma de decisiones por parte de los administradores.

## Requerimientos no funcionales

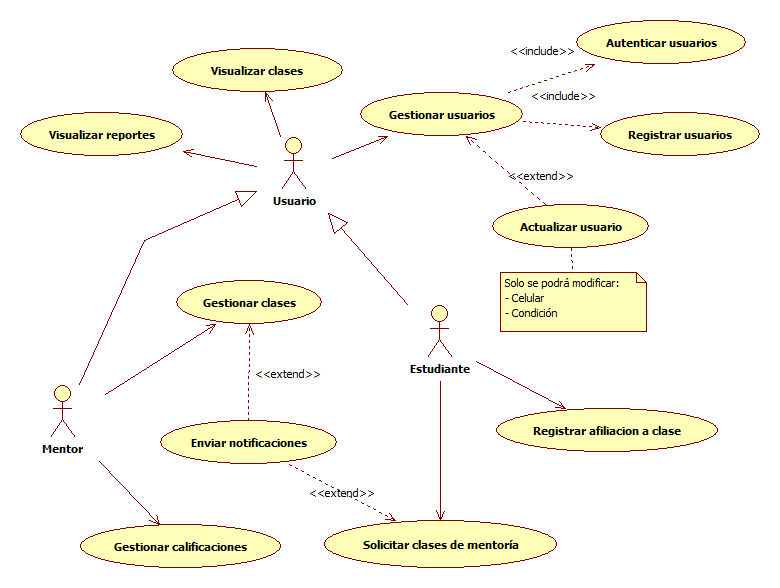
**Seguridad**: Utilización de encriptación avanzada y autenticación basada en roles para proteger la información personal y académica.

**Escalabilidad**: Estructura diseñada para soportar el crecimiento en el número de usuarios, funcionalidades y clases de mentoría sin afectar el rendimiento.

**Compatibilidad**: Soporte para dispositivos móviles y navegadores web modernos, permitiendo el acceso multiplataforma para los usuarios.

1. **Vistas de Caso de Uso**

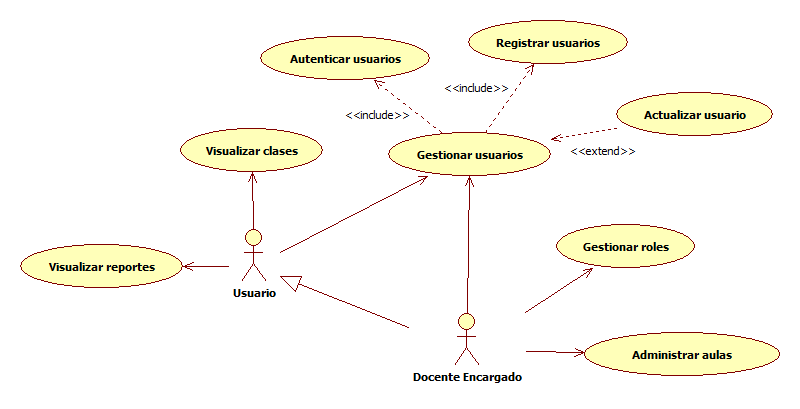
#### Diagrama de Casos de Uso del Sistema Web AMS

Figura 1: Diagrama de Casos de Uso del Sistema Web AMS. Fuente de origen: Propia.

### *Comentario:* Tenemos el diagrama de Casos de Uso del Sistema Escritorio AMS.

#### 

#### Diagrama de Casos de Uso del Sistema Escritorio AMS

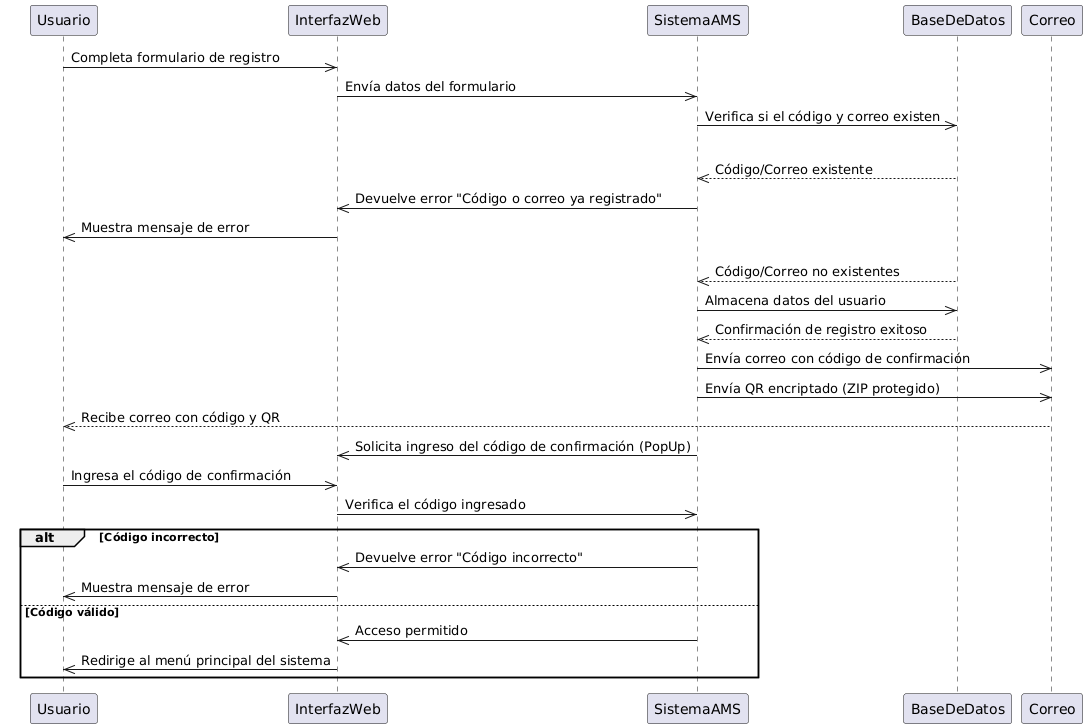
Figura 2: Diagrama de Casos de Uso del Sistema Escritorio AMS. Fuente de origen: Propia.

***Comentario:*** Tenemos el diagrama de Casos de Uso del Sistema Escritorio AMS.

1. **Vista Lógica**

## Diagrama de Secuencia

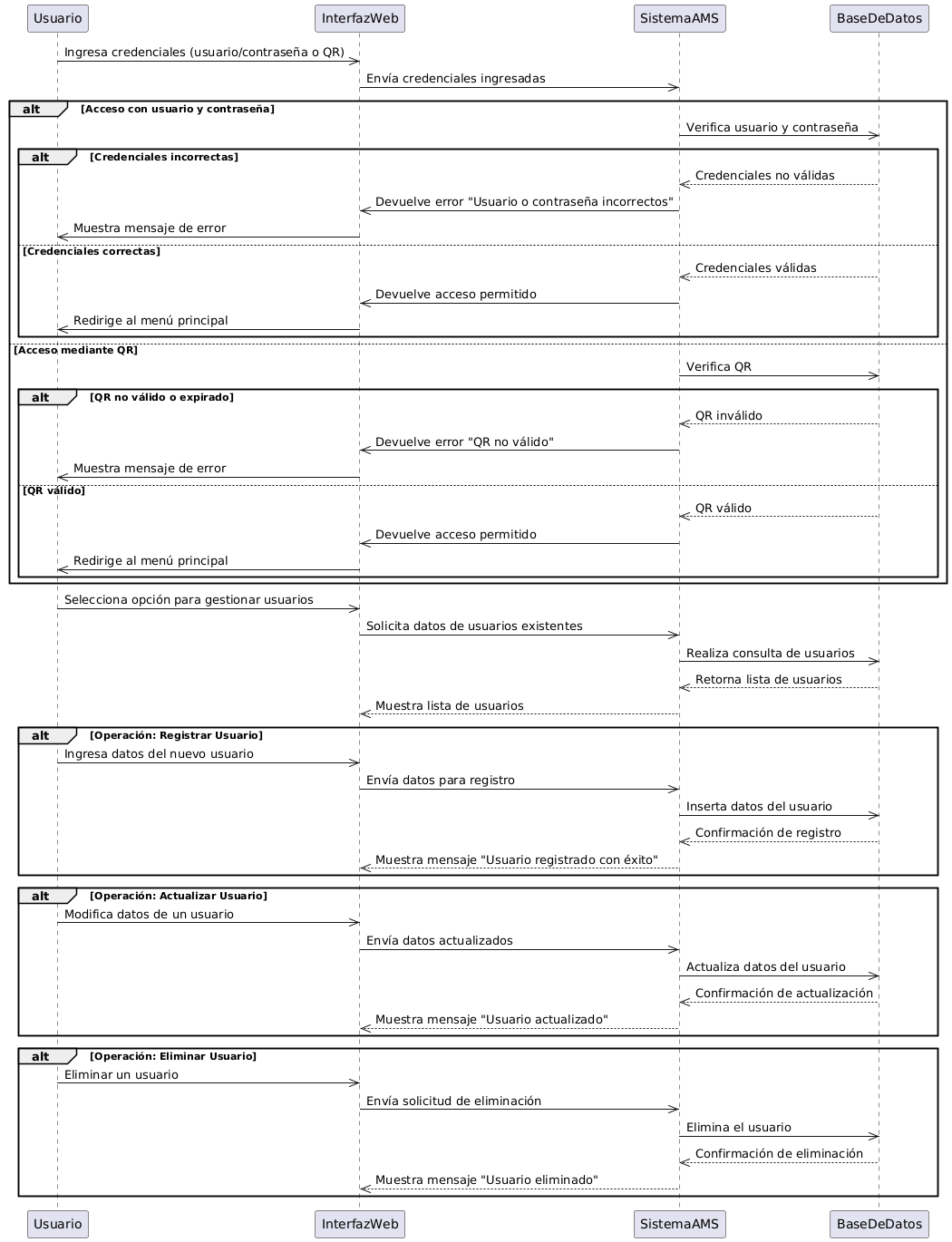
### SD-UC-002 Registrar usuarios

Figura 3: Diagrama de Secuencia: Registrar Usuarios. Fuente de origen: Propia.

### 

### 

### SD-UC-003 Gestionar usuarios

Figura 4: Diagrama de Secuencia: Gestionar usuarios. Fuente de origen: Propia.

### 

### SD-UC-004 Gestionar clases

Figura 5: Diagrama de Secuencia: Gestionar Clases. Fuente de origen: Propia.

### 

### SD-UC-008 Gestionar calificaciones

Figura 6: Diagrama de Secuencia: Gestionar Calificaciones. Fuente de origen: Propia.

## 

## 

## Diagrama de Clases

Figura 7: Diagrama de Clases Principal - Parte 1. Fuente: Elaboración Propia.

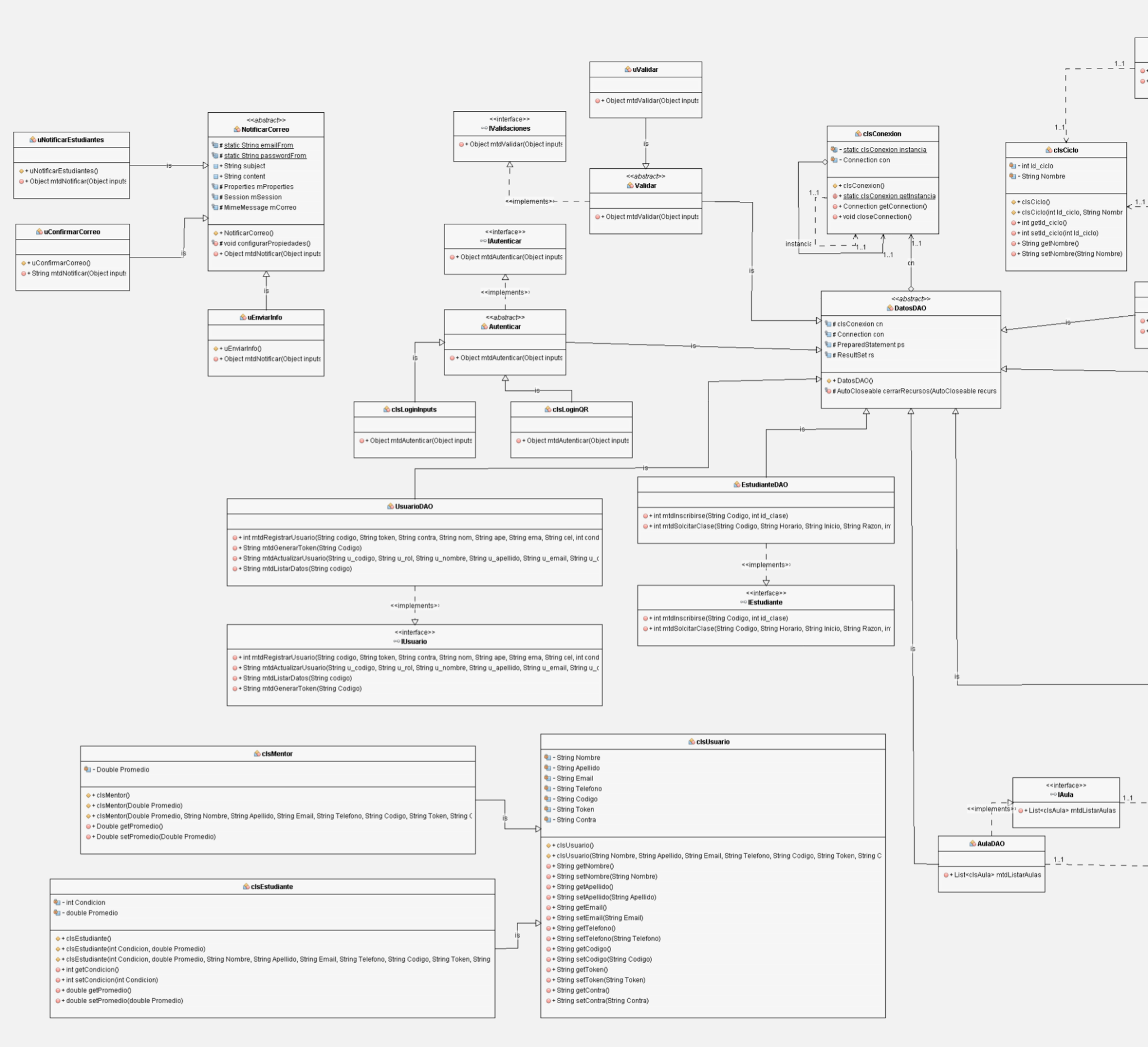


Figura 8: Diagrama de Clases Principal - Parte 2. Fuente: Elaboración Propia.

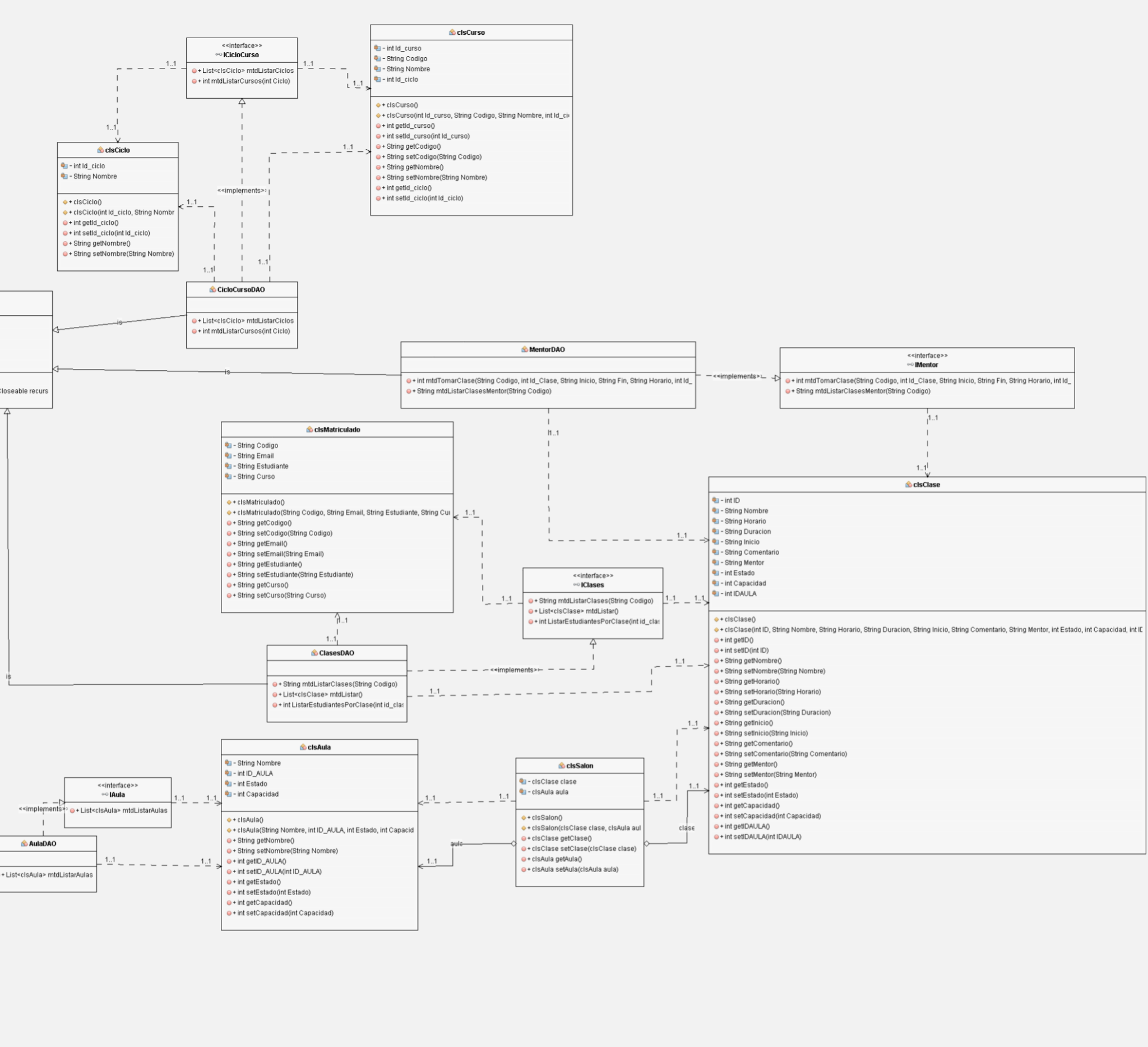
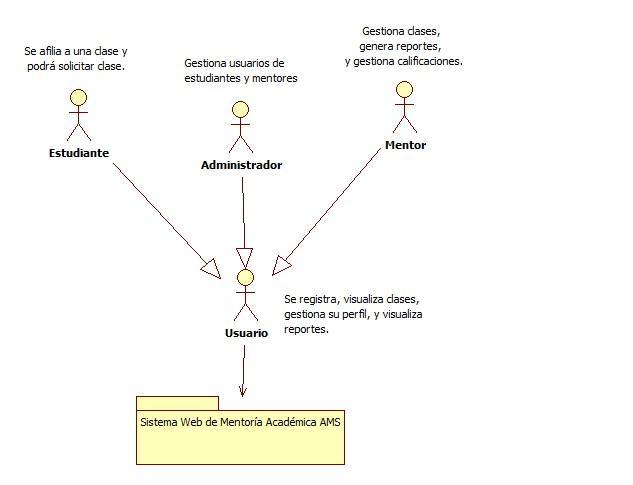


Figura 9: Diagrama de Clases Apartado - Redireccionar. Fuente: Elaboración Propia.

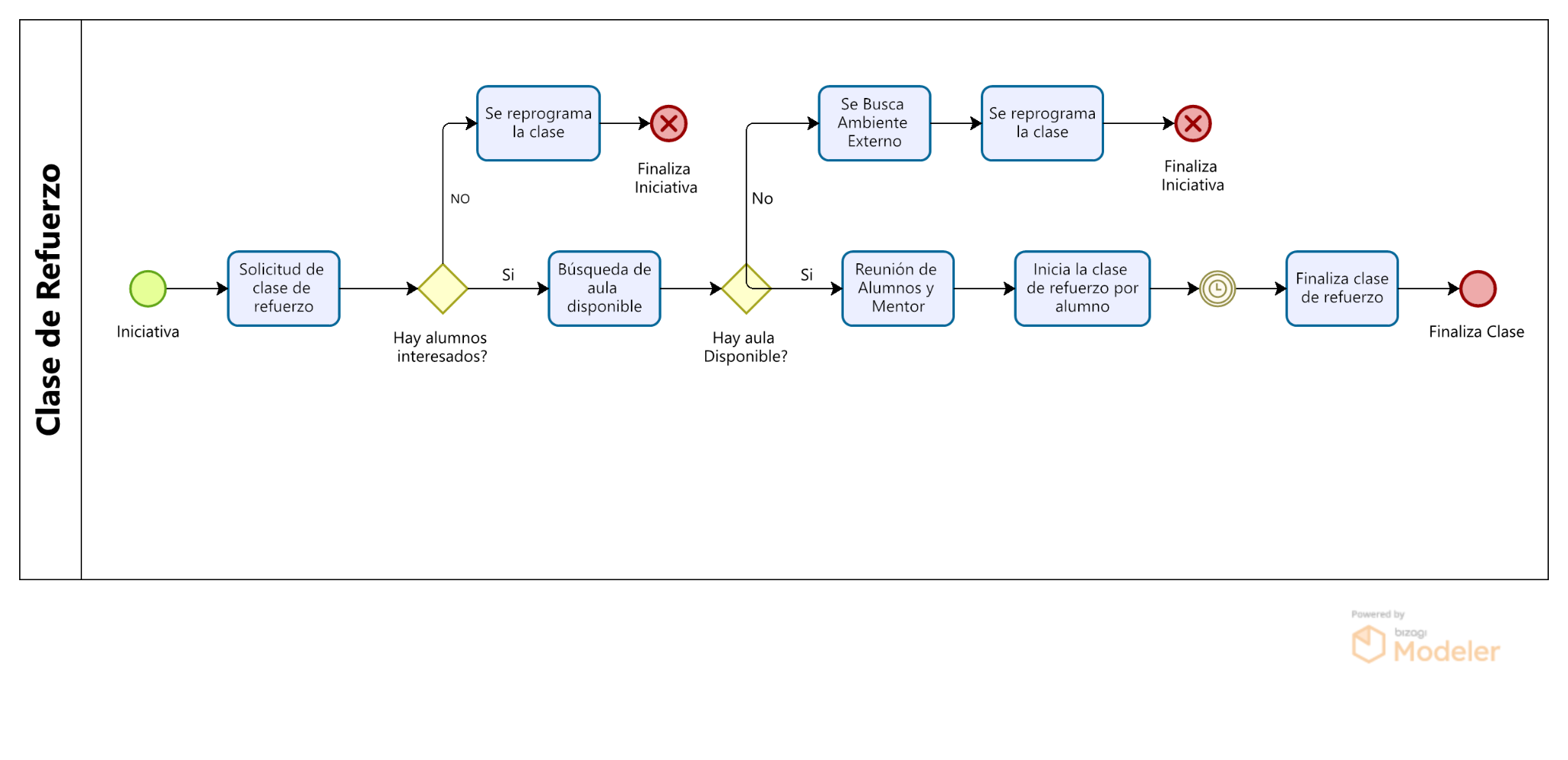
## 

## Diagrama Contextual

Figura 10: Diagrama Contextual. Fuente: Elaboración Propia.

1. **Vista de Procesos**

## Diagrama de Proceso Actual

Figura 11: Diagrama del Proceso Actual. Fuente de Origen: Propia.

Descripción: El diagrama de procesos actual adjunto representa a lo que se llamaría prototipo de mentoría en una situación real.

## 

## Diagrama de Proceso Propuesto

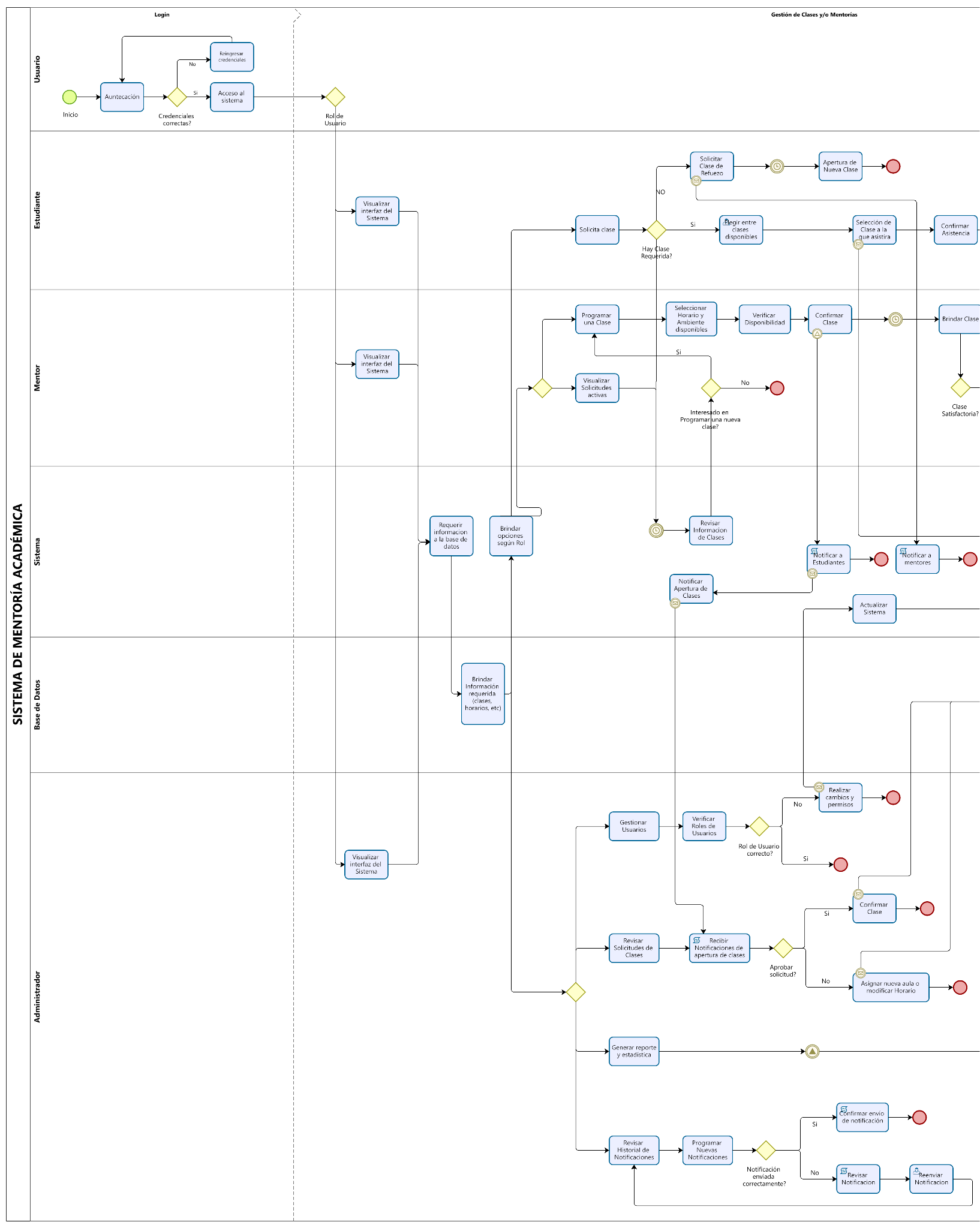
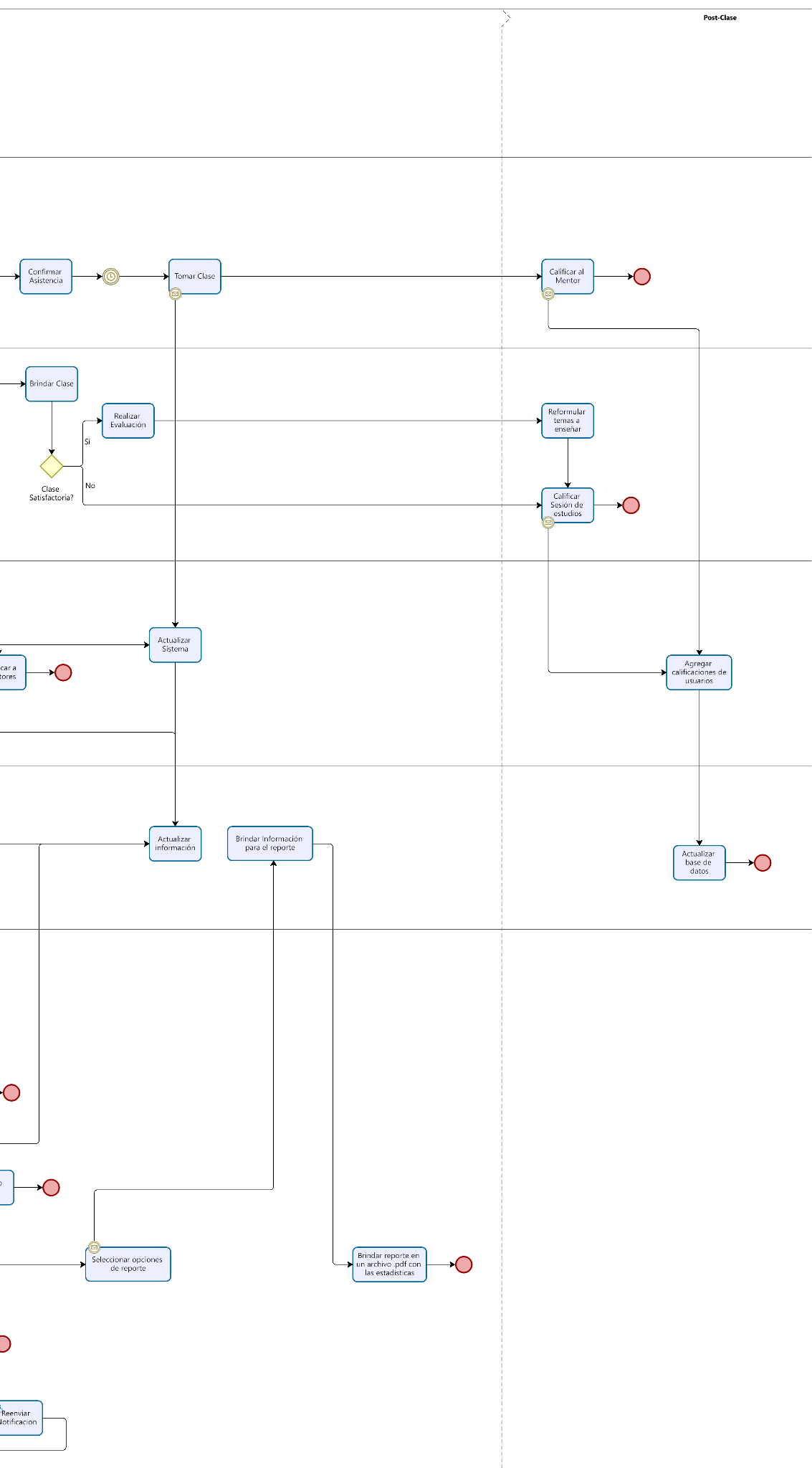
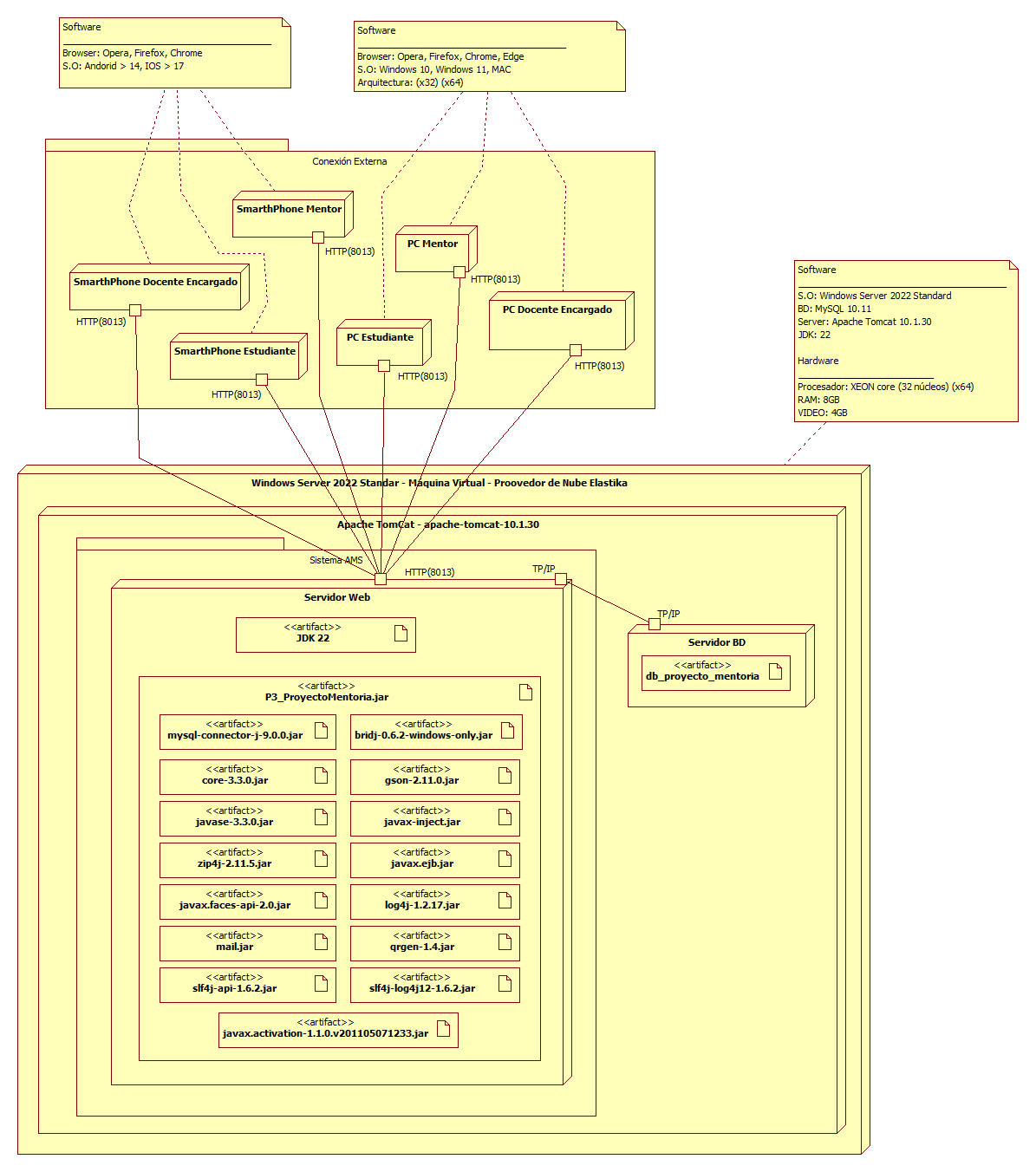
Figura 12: Diagrama del Proceso Propuesto - Parte 1. Fuente de Origen: Propia.

Figura 13: Diagrama del Proceso Propuesto - Parte 2. Fuente de Origen: Propia.

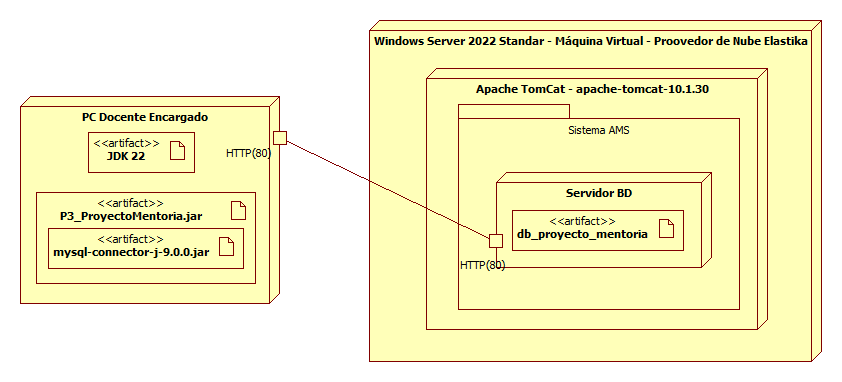
1. **Vista de Despliegue**

## Diagrama de Contenedor

### Diagrama de Contenedor del Sistema Web AMS

Figura 14: Diagrama del Contenedor. Fuente de Origen: Propia.

### Diagrama de Contenedor del Sistema Escritorio AMS

Figura 15: Diagrama del Contenedor. Fuente de Origen: Propia.

1. **Vista de Implementación**

## Diagrama de Paquetes

#### Diagrama de Paquetes de Sistema Web AMS

Figura 16: Diagrama de Paquetes de Sistema Web AMS. Fuente de origen: Propia.

### *Comentario:* Tenemos el diagrama de paquetes del Sistema Web AMS.

#### 

#### Diagrama de Paquetes de Sistema Escritorio AMS

Figura 17: Diagrama de Paquetes de Sistema Escritorio AMS. Fuente de origen: Propia.

### *Comentario:* Tenemos el diagrama de paquetes del Sistema Escritorio AMS.

## Diagrama de Componentes

## Figura 18: Diagrama de Componentes. Fuente de Origen: Propia.

1. **Vista de Datos**

## Diagrama Entidad Relación

## Figura 19: Diagrama Entidad Relación. Fuente de Origen: Propia.

1. **Calidad**

## Escenario de Seguridad

Para garantizar la seguridad de los datos, el AMS implementa encriptación de contraseñas y acceso basado en roles. Esto protege la información personal de los estudiantes y el progreso académico de accesos no autorizados. Además, se emplea autenticación de dos factores para los administradores, asegurando que solo usuarios autorizados tengan acceso a información sensible. Esta medida es esencial para cumplir con las normativas de protección de datos y brindar un entorno seguro para los usuarios.

## Escenario de Usabilidad

La usabilidad es un aspecto fundamental del AMS, diseñado para que los estudiantes puedan registrarse en el sistema, solicitar clases de mentoría y recibir notificaciones con la mínima cantidad de pasos. La interfaz es intuitiva y fácil de navegar, con un flujo de usuario que permite la realización de operaciones básicas en menos de tres pasos. Este enfoque garantiza que los estudiantes y mentores puedan acceder rápidamente a las funciones necesarias sin enfrentar complejidades innecesarias.

## Escenario de Adaptabilidad

El sistema AMS es adaptable y modular, lo que permite que el sistema crezca y se expanda para satisfacer nuevas necesidades. Por ejemplo, en el futuro se podría agregar un módulo de análisis predictivo que identifique estudiantes en riesgo académico. La arquitectura MVC facilita esta expansión sin comprometer la integridad o el rendimiento de los módulos existentes, permitiendo que el sistema se ajuste a los cambios en los requerimientos académicos o tecnológicos.

## Escenario de Disponibilidad

El AMS está diseñado para asegurar una disponibilidad continua de 24/7, lo que es crucial en un entorno educativo donde los estudiantes pueden necesitar apoyo fuera del horario normal de clases. La disponibilidad se garantiza mediante copias de seguridad regulares y un plan de recuperación ante fallos, que incluye restauración automática en caso de interrupciones imprevistas. Estas medidas aseguran que el sistema esté siempre accesible para los usuarios y minimizan el riesgo de pérdida de información crítica.

## Otro Escenario

El rendimiento es un aspecto clave en el diseño del AMS, optimizado para soportar al menos 100 usuarios simultáneos sin afectación en el tiempo de respuesta. Las operaciones comunes de consulta y actualización de datos deben completarse en menos de 2 segundos, proporcionando una experiencia fluida y ágil para los usuarios. Esto se logra mediante consultas SQL optimizadas y una gestión eficiente de los recursos del servidor, lo cual es fundamental para asegurar un rendimiento estable en entornos de alta demanda.