

Especificación de Requisitos de Software (SRS) - Proyecto de Gestión Académica

1. Introducción

1.1 Propósito

Este documento describe los requisitos funcionales y no funcionales del sistema de gestión académica desarrollado en Spring Boot con interfaz visual en angular, cuyo propósito es facilitar la gestión de estudiantes y programas académicos en una institución educativa.

1.2 Alcance

El sistema permitirá a los administradores gestionar la información de estudiantes y programas académicos, incluyendo la creación, edición, eliminación y consulta de registros. Los usuarios clave del sistema son el personal administrativo y académico.

1.3 Definiciones, acrónimos y abreviaturas

- SRS : Especificación de requisitos de software
- API : Interfaz de Programación de Aplicaciones
- REST : Transferencia Representacional de Estado
- Spring Boot : Marco de desarrollo rápido para aplicaciones Java

1.4 Referencias

Documentación oficial de Spring Boot : [Click](#).

1.5 Visión general

Este documento incluye los requisitos del sistema, las restricciones y los casos de uso principales para el sistema de gestión académica.

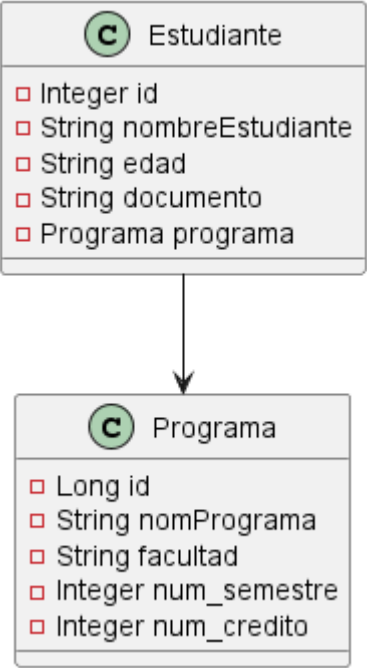
2. Descripción general del sistema

2.1 Perspectiva del producto

El sistema es una aplicación web basada en el modelo cliente-servidor. Utiliza Spring Boot como marco para construir y desplegar API RESTful, añadiendo una interfaz visual mediante el marco de trajo de angular, para gestionar datos a través de una base de datos MySQL. Los usuarios interactúan con la aplicación mediante solicitudes HTTP.

2.2 Funcionalidades del producto

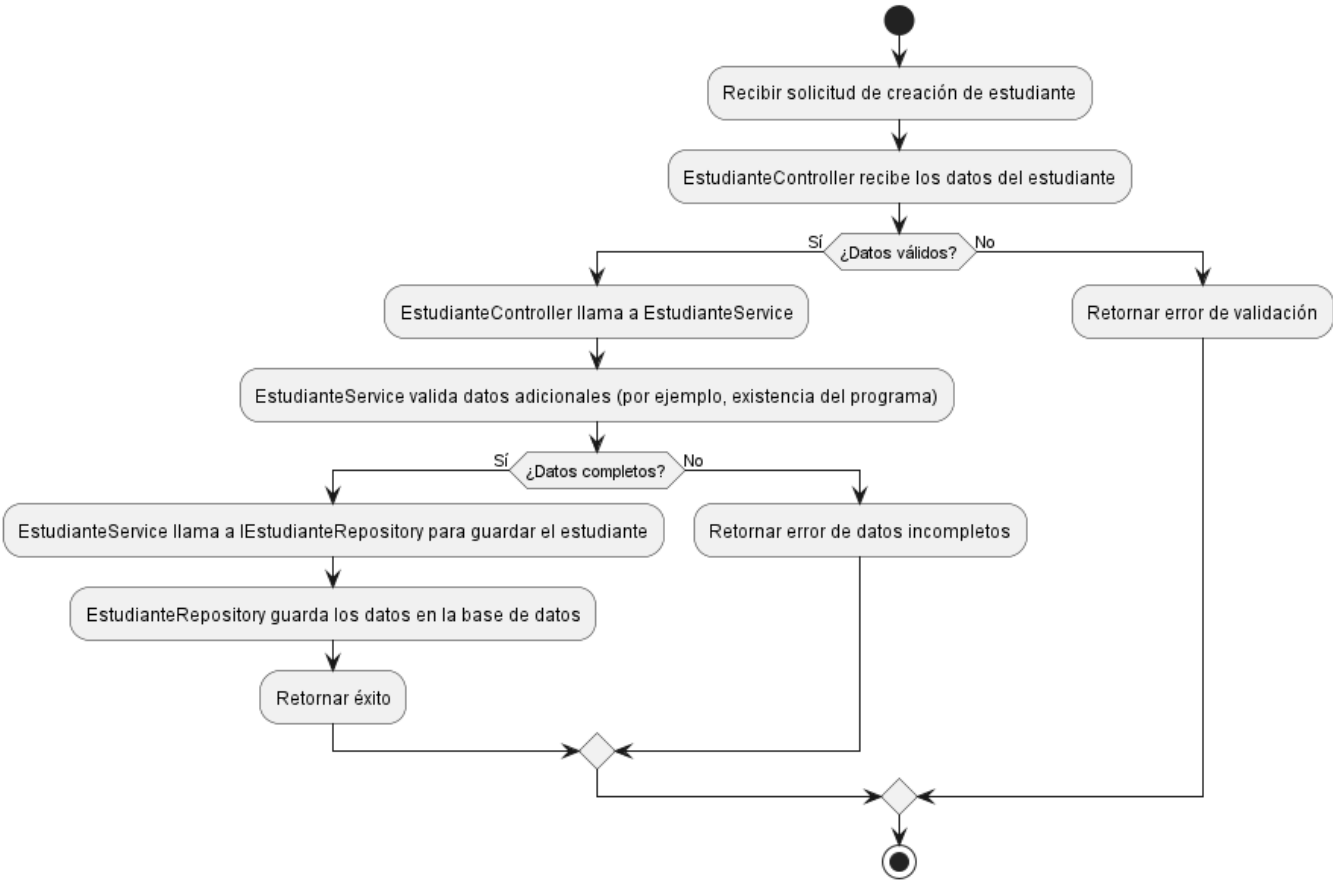
Diagrama de Clases:



Gestión de Estudiantes :

- Crear, consultar, actualizar y eliminar estudiantes.
- Buscar estudiantes por identificación o por documento de identidad.
- Obtenga la lista de estudiantes asociados a un programa académico específico.

Diagrama de Actividad:



Gestión de Programas :

- Crear, consultar, actualizar y eliminar programas académicos.
- Obtener información detallada de cada programa.

Diagrama de Actividad:

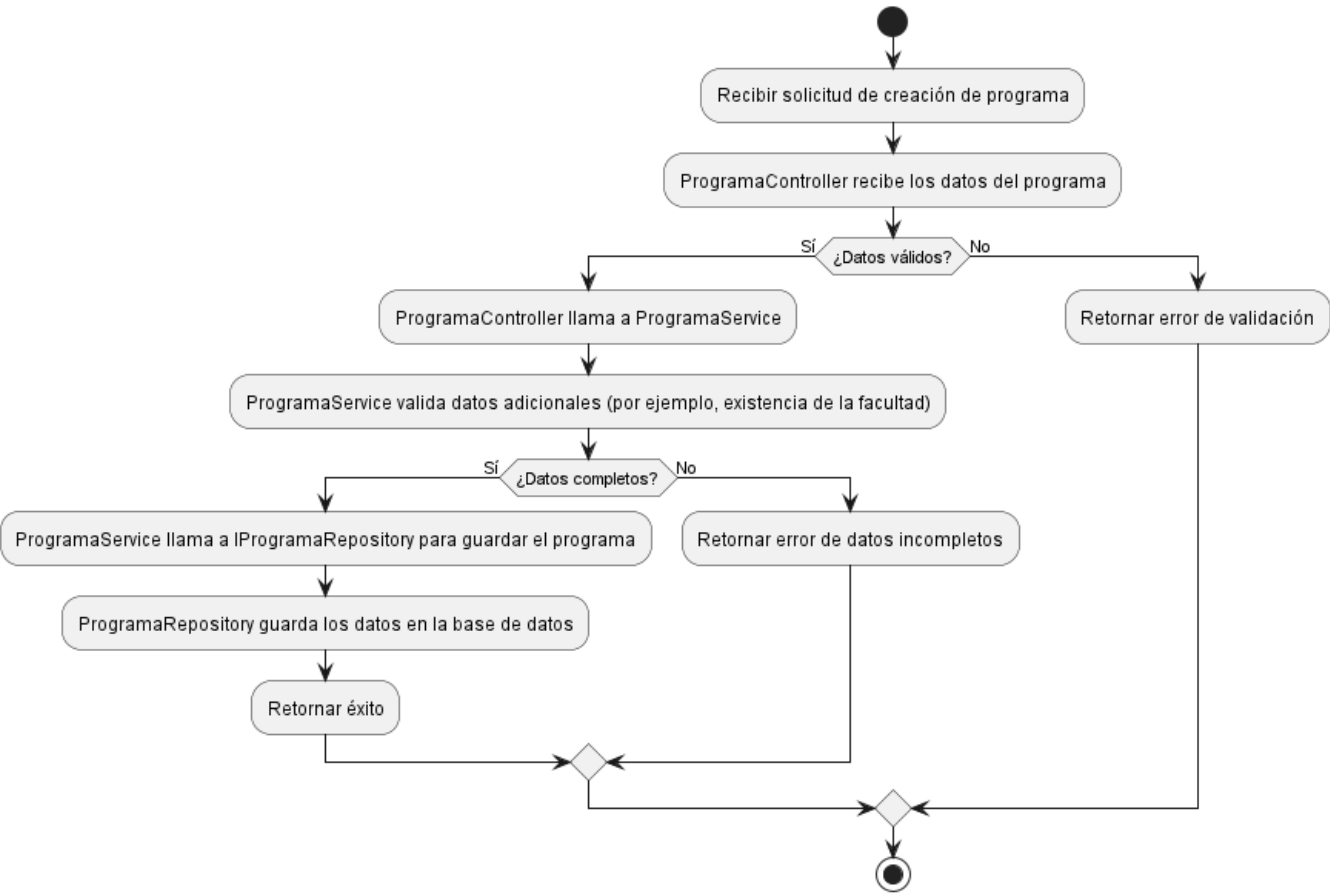
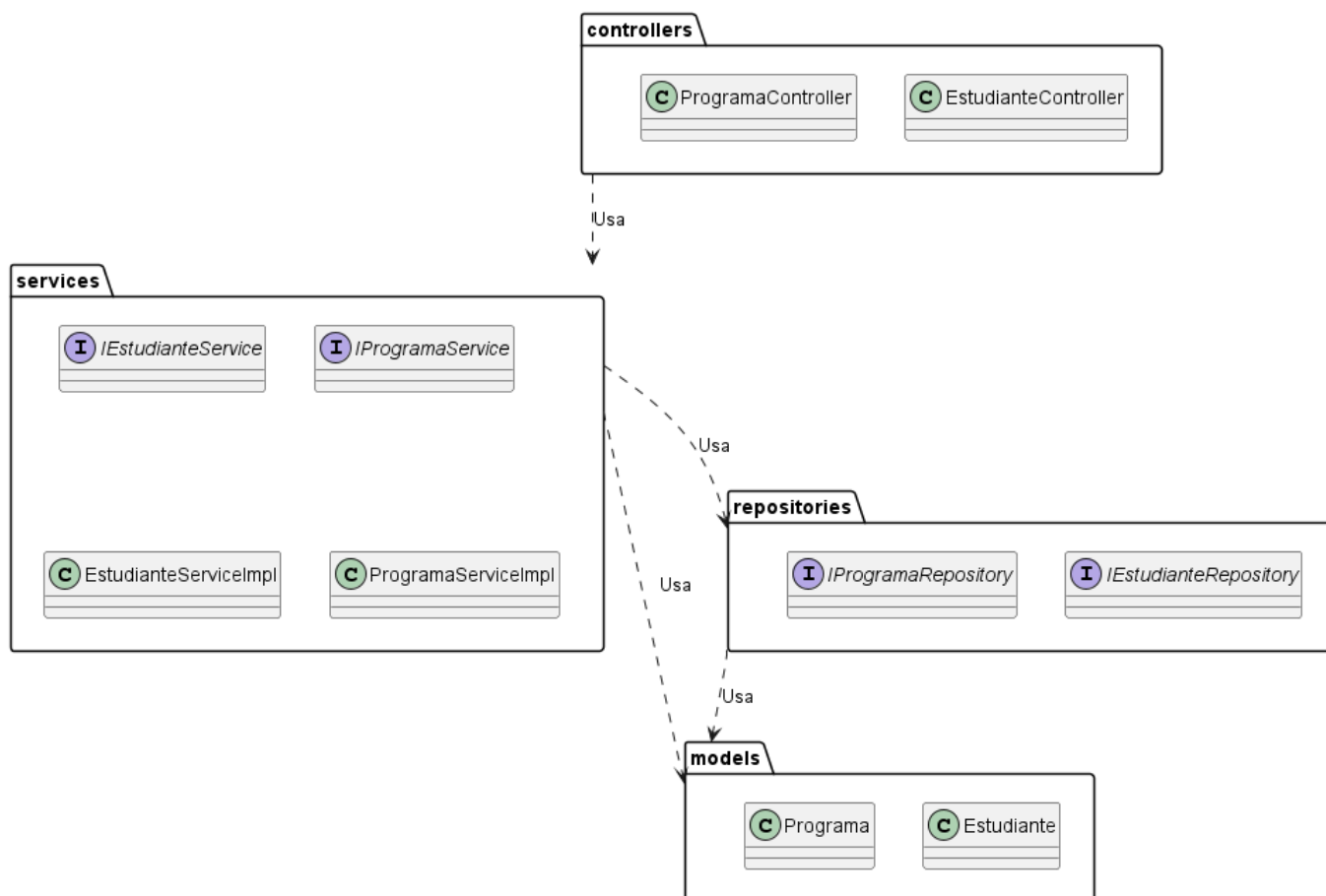


Diagrama de Paquetes :



Integración con MySQL :

- Persistencia de datos en una base de datos relacional MySQL.
- Uso de JPA para manejar operaciones con la base de datos.

2.3 Usuarios y características del sistema

Los principales usuarios del sistema son:

- Administradores : Encargados de gestionar toda la información de estudiantes y programas académicos.

2.4 Restricciones del sistema

- Base de datos : El sistema debe conectarse a una base de datos MySQL, lo que limita la capacidad de interacción con otras bases de datos si no se configuran adecuadamente los controladores.
- Navegadores : La interfaz de usuario debe ser compatible con navegadores modernos que soporten HTML5 y JavaScript.
- Rendimiento : El sistema debe responder en menos de 3 segundos para la mayoría de las solicitudes, incluso con una base de datos de gran tamaño.

2.5 Suposiciones y dependencias

Se supone que los usuarios tendrán acceso a internet y utilizarán la API desde dispositivos de escritorio con navegadores modernos. El sistema depende del entorno de ejecución Java (versión 17 o superior) y de un servidor MySQL.

3. Requisitos específicos

3.1 Requisitos funcionales

- RF-001 : El sistema debe permitir al usuario administrador crear, editar y eliminar estudiantes.
- RF-002 : El sistema debe permitir al usuario administrador consultar la lista de todos los estudiantes registrados.
- RF-003 : El sistema debe permitir consultar los detalles de un estudiante por su identificación o documento de identidad.
- RF-004 : El sistema debe permitir gestionar los programas académicos, incluyendo la creación, edición, consulta y eliminación.
- RF-005 : El sistema debe proporcionar una API para obtener los estudiantes asociados a un programa académico.

3.2 Requisitos no funcionales

- RNF-001 : El sistema debe tener un tiempo de respuesta inferior a 3 segundos para las solicitudes más comunes.
- RNF-002 : El sistema debe asegurar que solo los usuarios autenticados puedan realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar).
- RNF-003 : El sistema debe estar disponible el 99.5% del tiempo, excepto durante periodos planificados de mantenimiento.

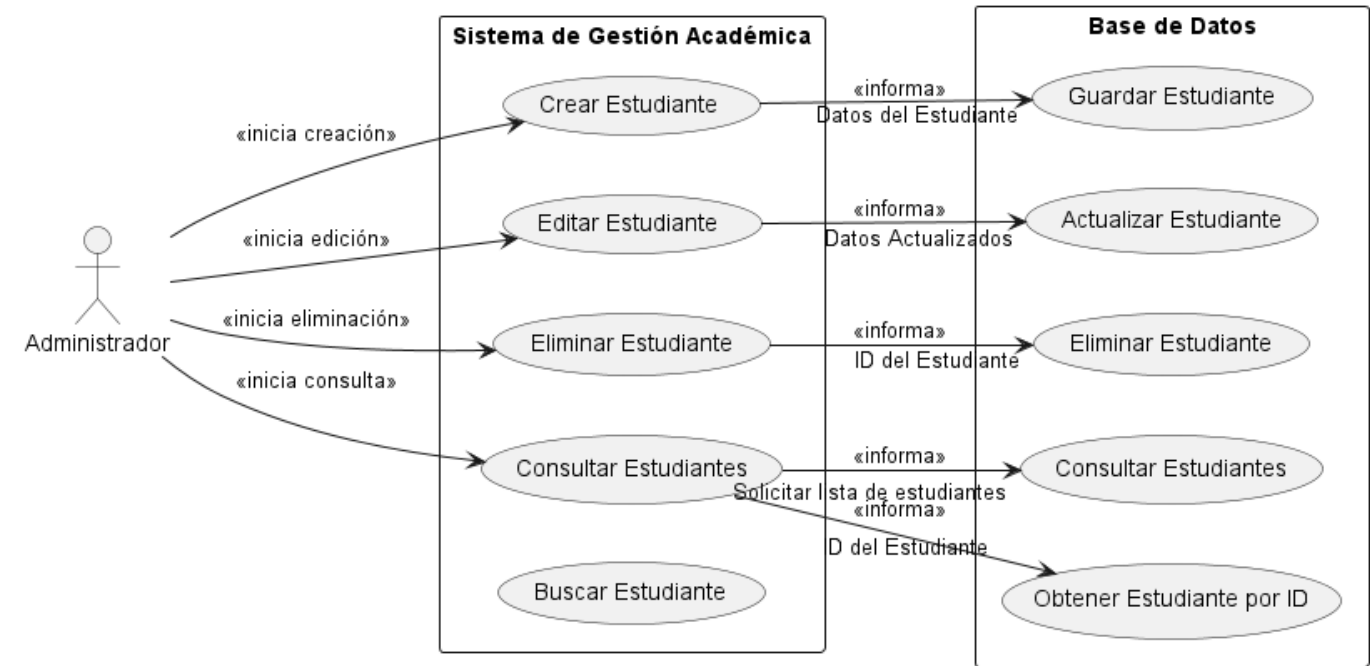
3.3 Restricciones

- R-001 : El sistema restringido estará a la integración con MySQL en la fase inicial. Soporte para otras bases de datos se podría implementar en futuras versiones.
-

4. Casos de uso

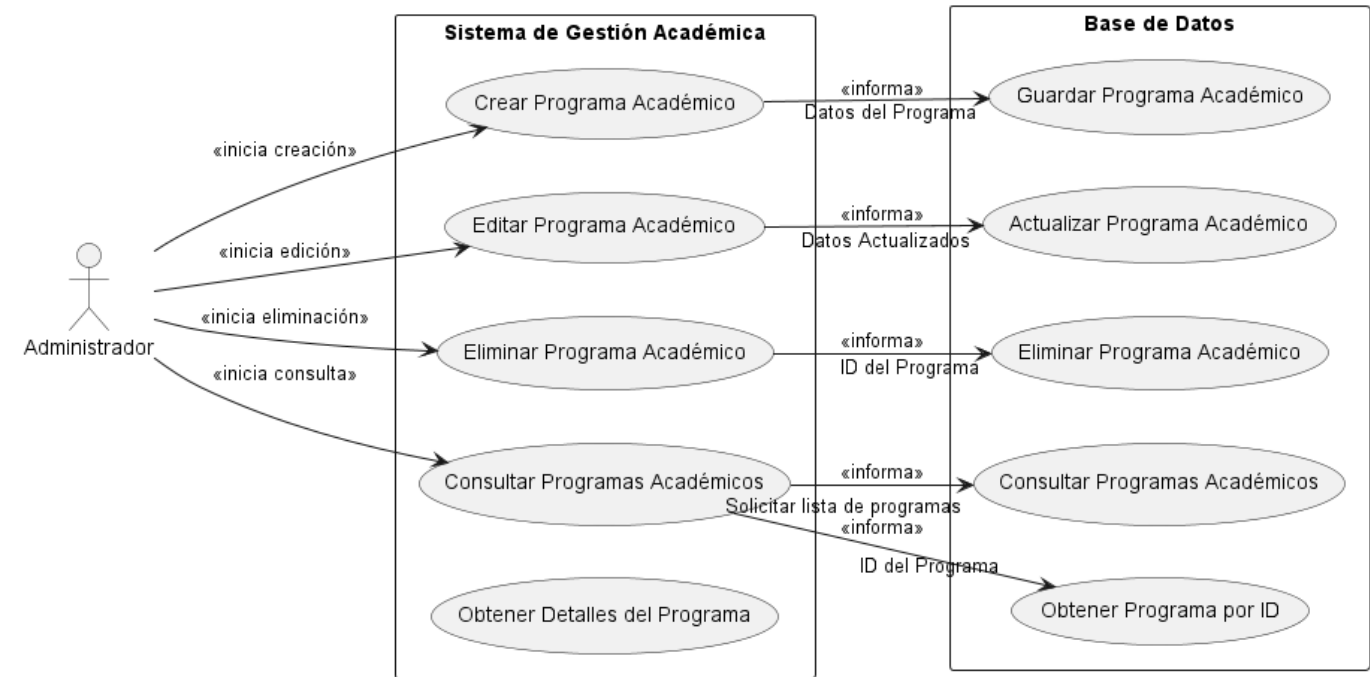
4.1 Caso de uso 1: Gestión de estudiantes

- Actores : Administrador.
- Descripción : El administrador puede crear, editar, eliminar y consultar a los estudiantes registrados en la base de datos.



4.2 Caso de uso 2: Gestión de programas académicos

- Actores : Administrador.
- Descripción : El administrador puede gestionar los programas académicos, incluyendo la creación, edición y eliminación de programas.



5. MoSCow

Must Have (Obligatorios)

- Gestión de estudiantes: El sistema debe permitir a los administradores realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los estudiantes (RF-001, RF-002, RF-003).

- Gestión de programas académicos: El sistema debe incluir la creación, edición, consulta y eliminación de programas académicos (RF-004).
- Autenticación y permisos de acceso: Solo los usuarios autenticados pueden realizar operaciones CRUD (RNF-002).
- Compatibilidad con MySQL: El sistema debe integrarse con una base de datos MySQL para la persistencia de datos (Restricción R-001).
- Tiempo de respuesta: El sistema debe responder en menos de 3 segundos para la mayoría de las solicitudes (RNF-001).

Should Have (Recomendados)

- Búsqueda avanzada de estudiantes: La funcionalidad para buscar estudiantes por identificación o documento de identidad (RF-003).
- Disponibilidad: El sistema debe garantizar un 99.5% de disponibilidad (RNF-003).
- Compatibilidad con navegadores modernos: La interfaz debe ser compatible con navegadores que soporten HTML5 y JavaScript (Restricción Navegadores).

Could Have (Opcionales)

- Soporte para otras bases de datos: Implementar soporte para bases de datos distintas de MySQL en futuras versiones (Restricción R-001).
- Listar estudiantes asociados a programas: La API podría proporcionar esta funcionalidad como un beneficio adicional para usuarios (RF-005).

Won't Have (No incluidos en esta versión)

- Compatibilidad con dispositivos móviles: No se menciona compatibilidad específica con dispositivos móviles, lo cual podría ser una extensión en versiones futuras.
- Soporte para entornos sin conexión a internet: Se asume que los usuarios tendrán acceso a internet, por lo que no se abordarán entornos sin conexión (Suposiciones y dependencias).

6. Figma

[Figma.](#)