



Código MVC.

Karen juliana Castillo.

Gestor del conocimiento.

William Alexander Matallana Parra.

Universidad de Cundinamarca, extensión Chía.

Ingeniería de Sistemas y computación.

18 de febrero de 2026.



Contenido.

Código MVC.	1
Formato de Requerimientos del sistema	3
1. Información general del proyecto.....	3
2. Descripción general del sistema	3
3. Requerimientos Funcionales (RF).....	3
4. Requerimientos No Funcionales (RNF)	4
5. Relación Requerimiento – POO	4
Análisis.	5
Diagrama de flujo.	5
Diagrama de Clases	5
Diagrama de casos de uso.....	6
Diseño.	7
Referencias.	8



Formato de Requerimientos del sistema.

1. Información general del proyecto

Nombre del proyecto	Sistema de Control de asistencia
Integrantes	Karen Juliana Castillo Barreto
Programa académico	Ingeniería de Sistemas y Computación
Fecha	18/02/2026
Lenguaje de programación	Java
Tipo de aplicación	Consola

2. Descripción general del sistema

- El sistema permite crear estudiantes.
- El sistema permite registrar la asistencia de los estudiantes, sin conceder la realización de una doble asistencia el mismo día.
- El sistema permite eliminar a los estudiantes de la lista de asistencia.
- La aplicación esta desarrollada en java utilizando los principios de programación orientada a objetos.

3. Requerimientos Funcionales (RF)

ID	Nombre	Descripción	Entrada(s)	Proceso	Salida
RF-01	Crear Estudiante.	Permite registrar un nuevo estudiante con su nombre	Nombre del estudiante	Crea un nuevo estudiante.	Estudiante creado y agregado a la lista de asistencia.
RF-02	Marcar asistencia.	Permite cambiar el estado de la asistencia.	Nombre del estudiante.	Cambia el estado de la asistencia.	El estado de la asistencia es actualizado.
RF-03	Consultar asistencia.	Muestra una lista con todos los estudiantes y	Lista de los estudiantes.	Recorre la lista de los estudiantes	Nombre de los estudiantes



		su estado de asistencia.			y estado de asistencia.
RF-04	Eliminar Estudiante	Permite eliminar un estudiante de la lista de asistencia.	Lista de estudiantes y nombre de estudiante.	Elimina un estudiante de la lista.	Estudiante eliminado de la lista.
RF-05	Asistió o no asistió.	Permite observar un texto donde explícitamente se encuentra asistido o no asistido.	Nombre del estudiante.	Dependiendo del estado de la asistencia, se observa un mensaje de asistencia o no asistencia.	Texto donde se observa su asistencia o no asistencia.

4. Requerimientos No Funcionales (RNF)

ID	Tipo	Descripción
RNF-01	Validación	El sistema no debe permitir que algún estudiante asista dos veces en el mismo día.
RNF-02	Estructura	El sistema debe estar organizado por paquetes: model, service y útil.
RNF-03	Calidad	El sistema debe aplicar principios de calidad de código como encapsulamiento, polimorfismo y el ciclo de vida del desarrollo del software.

5. Relación Requerimiento – POO

ID Requerimiento	Clase	Método	Tipo
RF-01	EstudianteService	CrearEstudiante()	Funcional
RF-02	EstudianteService	MarcarAsistencia()	Funcional
RF-03	EstudianteService	ConsultarAsistencia()	Funcional
RF-04	EstudianteService	EliminarEstudiantes()	Funcional
RF-05	EstudianteService	AsistioONo()	Funcional
RNF-01	EstudianteService / EstudianteIMP	MarcarAsistencia() + validación de fecha	Funcional



Análisis.

Diagrama de flujo.

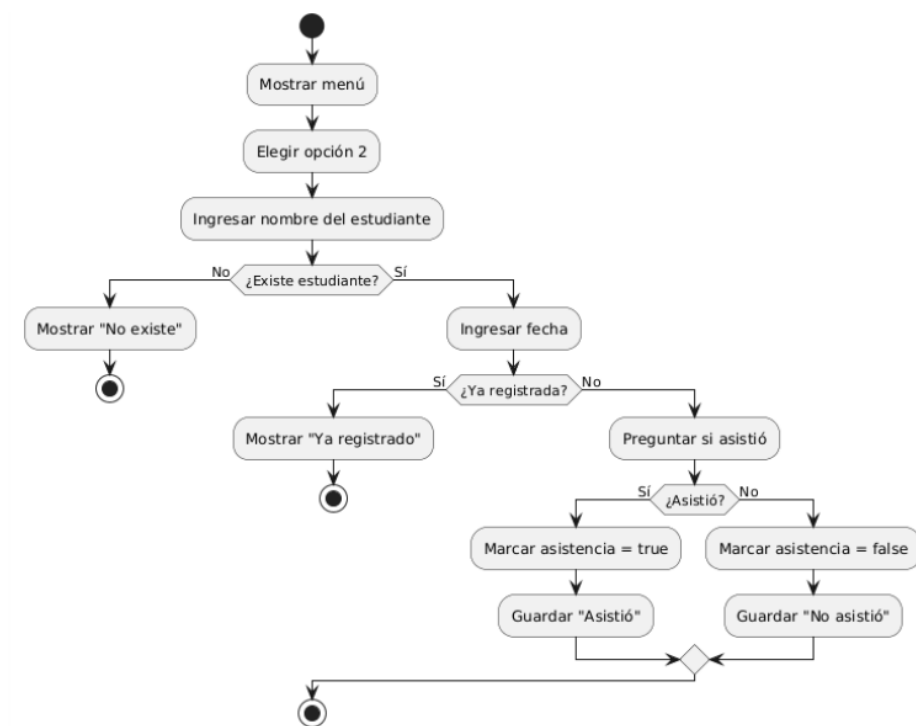


Diagrama de Clases.

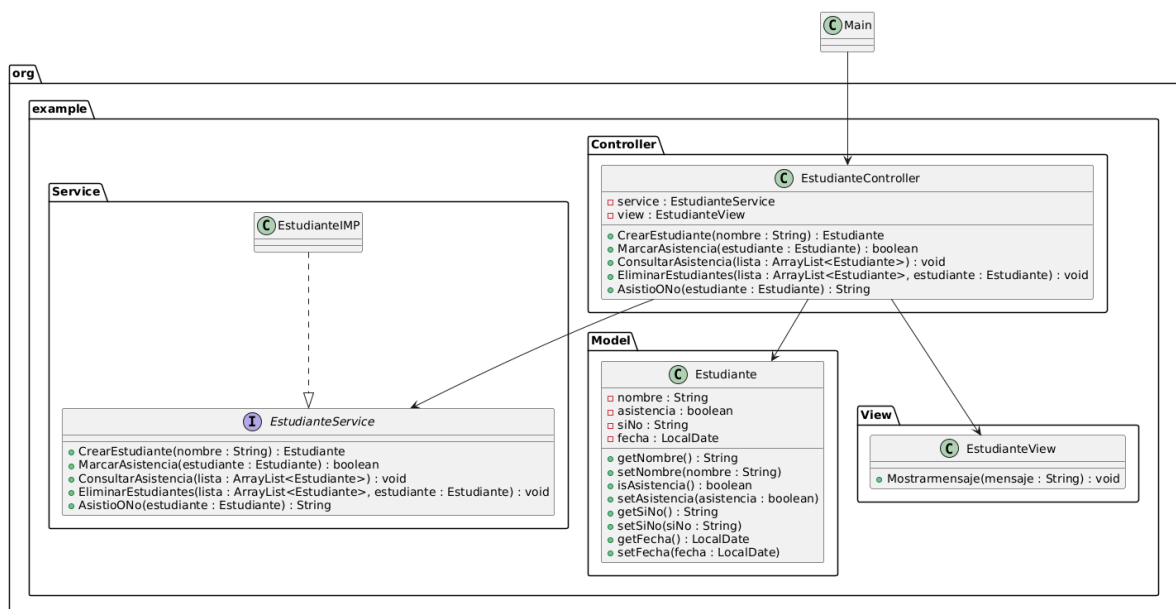




Diagrama de casos de uso.





Diseño.

Capa de presentación (controller/UI)

Main.java

EstudianteView.java

Capa de control (controller)

EstudianteController.java

Capa de Negocio (Services)

EstudianteService.java

EstudianteServiceIMP.java

Capa de Datos (Repositories)

Estudiante.java

Tecnologías.

JDK 23

IntelliJ

PlantUML



Referencias.

OpenAI. (2026). ChatGPT (modelo GPT-5-mini) [Asistencia para generación de diagramas UML y diagramas de flujo]. <https://openai.com/chatgpt>

PlantUML. (s. f.). PlantUML: Generador de diagramas UML en línea. https://www.plantuml.com/plantuml/uml/xLLD3n8n4BttLpIS8Ch-W8qX4CJ4X4KJ4kz3TrYhtPPC2sgK_-vsgMK_wiRerHFGzzxqpRnfM5i7vAwrOXSGdw52FZDKPVWLwel2h30bgXd_pZWN2gpbE-kkfGJji3taVCcrgK-4FET7Hr9NuHlij0wraD0W9sCKWWwGbGSpf9zHVFJCIH6WNeF3zlm5hz0TslhPHLVY8HkXWOD584YxYIQyFDg8-WwUyhjucZWsdCMMg3NKKVrndVbPQE-x2zmrBYNtzcTfGCv8jsam4Saiec1pY2m4qiWjBY_Q4i85E7nh7WUnT5S0II0EiF1oEZHnpFT6cslolN4gldeEzmGmRTzy-FLiyBAbxWPMGRzJibQwgvXEwr_oIUyTGgtVgHvruGr8OZXtbbYuzyMxt-RomU9NzBln2CnZS9OHq0rQWkVS1u-uytHzDGAD3DsP9H2Ib_unECJjM4fy16Y1CRDkym7jl-lqB-ia0_maKq1KZFMMoIUPJzMYJIIClwNVq6D8Vu1Ztc7yMn8PclKfV-B-0a0