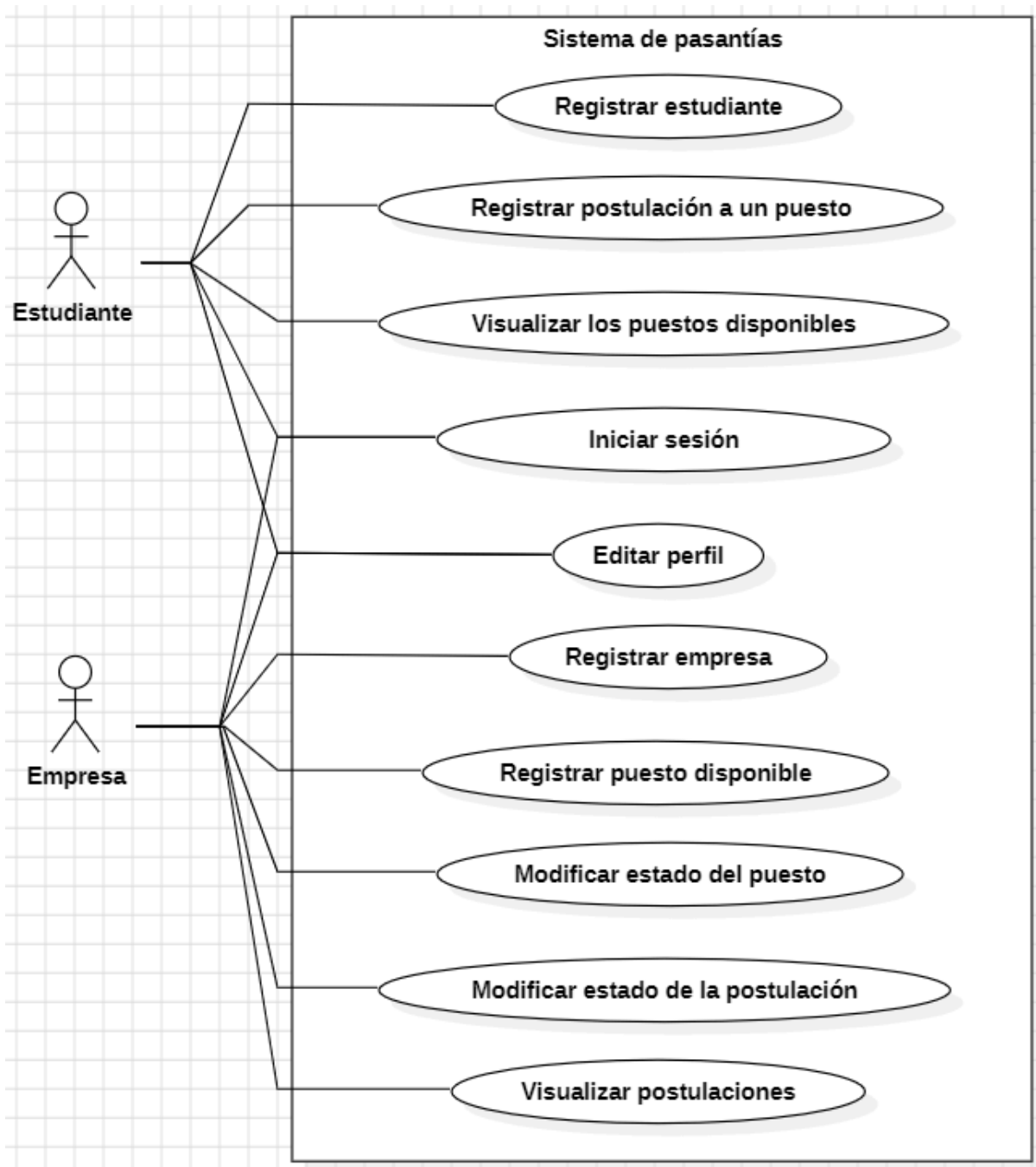


Proyecto: Sistema de Pasantías - Iteración 1 y 2

Procesos

1. Registrar alumno: permite a un estudiante crear su perfil en el sistema.
2. Registrar empresa: permite a una empresa registrarse para publicar puestos.
3. Iniciar sesión: permite el acceso autenticado para estudiantes y empresas.
4. Registrar nuevo puesto disponible: permite a la empresa crear un nuevo puesto disponible.
5. Visualizar puestos: permite a estudiantes ver las oportunidades disponibles.
6. Postularse a un puesto: permite a un estudiante postularse a un puesto disponible mediante la carga de su cv.
7. Modificar el estado de un puesto: permite a la empresa modificar si un puesto ya no se encuentra disponible.
8. Visualizar postulaciones: permite a la empresa visualizar quienes se han postulado al puesto y su cv.
9. Modificar el estado de una postulación: permite a la empresa modificar el estado de la postulación de un estudiante.
10. Editar perfil: permite a estudiantes y empresas modificar sus datos ingresados.



Requerimientos

Requerimientos funcionales:

- RF01 - Registro de alumnos: El sistema debe permitir a los estudiantes crear un perfil proporcionando datos personales y académicos.
- RF02 - Registro de empresas: El sistema debe permitir a las empresas crear una cuenta y registrar información relevante de la organización.

- RF03 - Inicio de sesión: El sistema debe permitir a estudiantes y empresas iniciar sesión con credenciales válidas.
- RF04 - Publicación de puestos: El sistema debe permitir a las empresas registrar nuevos puestos de pasantías con título, descripción, requisitos y fecha de cierre.
- RF05 - Visualización de puestos: El sistema debe permitir a los estudiantes ver los puestos disponibles publicados por las empresas.
- RF06 - Postulación a un puesto: El sistema debe permitir a los estudiantes postularse a un puesto mediante la carga de su CV o usando el que ya tienen en su perfil.
- RF07 - Modificación del estado de un puesto: El sistema debe permitir a las empresas cambiar el estado de un puesto a "cerrado", "ocupado" u otro estado definido.
- RF08 - Visualización de postulaciones: El sistema debe permitir a las empresas ver los estudiantes que se postularon a un puesto, incluyendo su información y CV.
- RF09 - Modificación del estado de una postulación: El sistema debe permitir a las empresas actualizar el estado de una postulación (por ejemplo: "en revisión", "rechazado", "seleccionado").
- RF10 - Edición de perfil: El sistema debe permitir a estudiantes y empresas modificar la información de su perfil.

Requerimientos no funcionales:

- RNF01 - Usabilidad: La interfaz del sistema debe ser intuitiva y fácil de usar tanto para estudiantes como para empresas.
- RNF02 - Rendimiento: El sistema debe poder responder a cada acción del usuario en un tiempo inferior a 3 segundos.
- RNF03 - Seguridad: Las contraseñas deben ser almacenadas de forma segura.
- RNF04 - Accesibilidad: El sistema debe ser accesible desde diferentes dispositivos (PC, tablet, celular) y navegadores modernos.
- RNF05 - Disponibilidad: El sistema debe estar disponible para los usuarios 24/7, salvo en casos de mantenimiento programado.
- RNF06 - Privacidad: La información personal de los usuarios (CVs, datos personales, etc.) no debe ser visible para otros usuarios no autorizados.

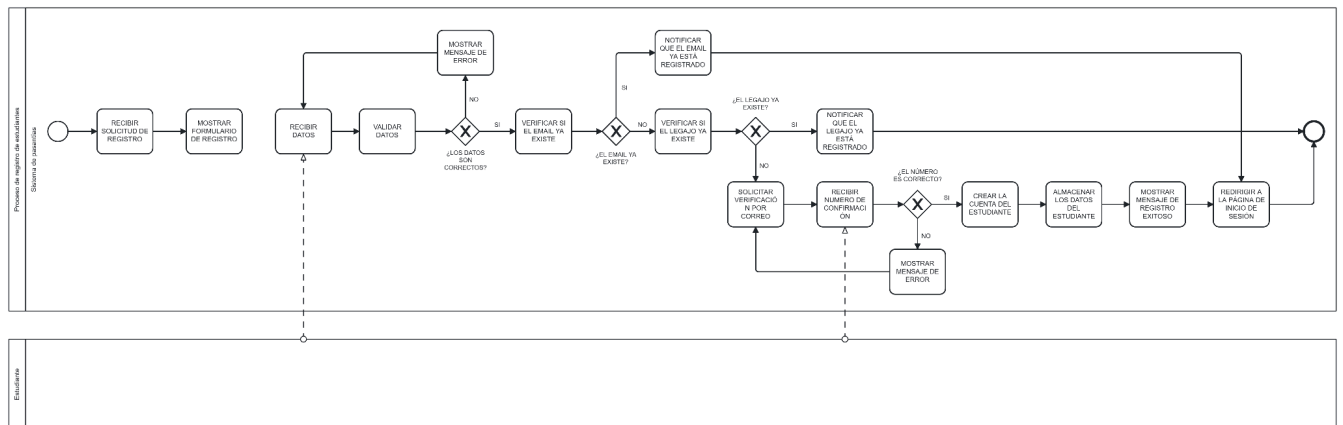
Selección de proceso:

Registrar alumno.

Descripción de caso de uso:

Caso de Uso: Registrar estudiante	
ID: 1	Fecha: 06/05/2025
Descripción: El caso de uso debe permitir registrar a un estudiante al sitio de pasantías.	
Actores Principales: Estudiante	Actores Secundarios: -
Observaciones: -	
Precondiciones: El estudiante no posee una cuenta activa en el sistema.	
Post- Condiciones	Éxito: El estudiante es registrado con éxito.
	Fracaso: No es posible registrar al estudiante.
<u>FLUJO PRINCIPAL</u>	FLUJO ALTERNATIVO
1. El caso de uso comienza cuando el estudiante selecciona la opción "Registrarse como estudiante".	
2. El sistema presenta el formulario de registro solicitando: Nombre, Apellido, Correo electrónico, Fecha de nacimiento, Legajo, Carrera, Año de cursada, Contraseña.	
3. El estudiante ingresa la información solicitada y confirma la operación.	
4. El sistema valida los datos ingresados.	4.1 El sistema resalta los campos con errores y muestra mensajes descriptivos. 4.2 El caso de uso vuelve al paso 3.
5. El sistema verifica que el correo electrónico no exista previamente para otro usuario.	5.1 El correo ya se encuentra registrado. 5.2 El Sistema informa al estudiante que el email ya se encuentra registrado. 5.3 El sistema permite al usuario modificar el Email.
6. El sistema verifica que el número de legajo no exista previamente para otro estudiante.	5.1 El legajo ya se encuentra registrado. 5.2 El Sistema informa al estudiante que el legajo ya se encuentra registrado. 5.3 El sistema permite al usuario modificar el legajo.
7. El Sistema genera un código de verificación único y establece su tiempo de validez.	
8. El Sistema, utilizando un Servicio de Email, envía el código de verificación al email proporcionado.	8.1 El sistema no puede enviar el email. 8.2 El sistema informa al estudiante y ofrece la opción de "Reenviar código". 8.3 El caso de uso vuelve al paso 7.
9. El estudiante ingresa el código de verificación en el formulario del Sistema y confirma.	
10. El sistema valida el código ingresado y es correcto.	10.1 El código ingresado es incorrecto. 10.2 El sistema le ofrece la opción de "Reenviar código". 10.3 El caso de uso vuelve al paso 7.
11. El sistema crea la cuenta del estudiante, asociando los datos ingresados.	
12. El sistema almacena los datos del nuevo estudiante.	
13. El sistema muestra un mensaje de "Registro exitoso".	
14. El sistema redirige al estudiante a la página de inicio de sesión o al panel principal del estudiante.	

BPMN



Prototipo:

PASANTÍAS UNIVERSITARIAS - UTN FRSCO

Registro de Estudiante

Nombre

Apellido

Email

Contraseña

Confirmar Contraseña

Legajo

Fecha de Nacimiento

Carrera ▼

Año de Cursada ▼

Conocimientos y Habilidades

Describe tus habilidades técnicas, idiomas, certificaciones, etc.

Crear cuenta

¿Ya tienes una cuenta? [Iniciar sesión](#)

Implementación:

<https://sistema-pasantias.vercel.app/>

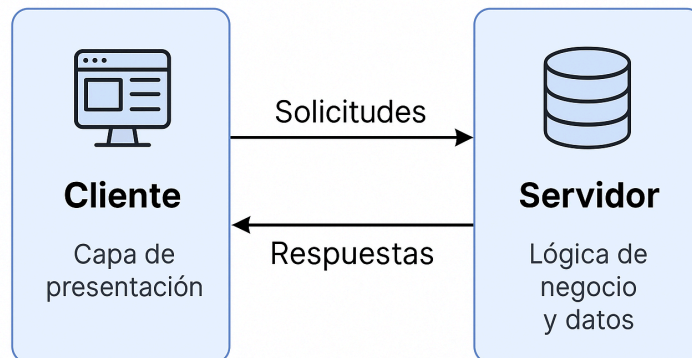
Testing:

https://docs.google.com/spreadsheets/d/1zgctHJC5_DexuF-NQFI37t5TC_oPaVN7/edit?usp=sharing&ouid=109314850177132573853&rtpof=true&sd=true

Arquitectura del sistema

Modelo elegido: Arquitectura Cliente/Servidor.

El sistema se basa en una arquitectura cliente/servidor tradicional. La lógica de negocio y la persistencia de datos se ejecutan del lado del servidor, mientras que el cliente se encarga de la interacción con el usuario.



Cliente (Frontend - navegador del usuario)

Responsabilidades:

- Mostrar la interfaz gráfica del sistema (pantallas de registro, login, perfil, búsqueda de puestos)
- Capturar entradas del usuario mediante formularios.
- Realizar validaciones básicas (formato de email, campos obligatorios).
- Enviar solicitudes HTTP al servidor usando formularios.
- Mostrar respuestas del servidor (errores, confirmaciones, listados).

Ejemplos de tareas:

- Enviar formulario de registro de alumno/empresa.
- Mostrar puestos disponibles.
- Cargar un CV para postularse.
- Mostrar el estado de una postulación.

Servidor (Backend)

Responsabilidades:

- Validar credenciales de acceso y permisos.
- Procesar la lógica del sistema (registro, postulación, cierre de puestos, etc.).
- Acceder a la base de datos para leer, crear o actualizar información.
- Encriptar contraseñas y proteger datos sensibles.
- Devolver respuestas estructuradas (por ejemplo, en formato JSON) al cliente.

Ejemplos de tareas:

- Verificar si un usuario está registrado.
- Almacenar nuevos puestos o postulaciones.
- Consultar y devolver listas de datos (puestos, postulaciones, perfiles).
- Cambiar el estado de un puesto o una postulación.

Listado de Endpoints

Autenticación

Método	Endpoint	Descripción
POST	/api/login	Inicia sesión un usuario (empresa o estudiante).
POST	/api/logout	Cierra sesión del usuario actual.
POST	/api/recuperar	Envía un correo con enlace para recuperar contraseña.
POST	/api/reestablecer	Reestablece la contraseña con un token válido.

Estudiantes

Método	Endpoint	Descripción
POST	/api/estudiantes	Registra un nuevo estudiante.
GET	/api/estudiantes/:id	Devuelve los datos de un estudiante por su ID.
PUT	/api/estudiantes/:id	Edita el perfil de un estudiante. (Solo estudiante)

Empresas

Método	Endpoint	Descripción
POST	/api/empresas	Registra una nueva empresa.
GET	/api/empresas/:id	Devuelve los datos de una empresa por su ID.
PUT	/api/empresas/:id	Edita el perfil de una empresa. (Solo empresa)

Puestos de pasantía

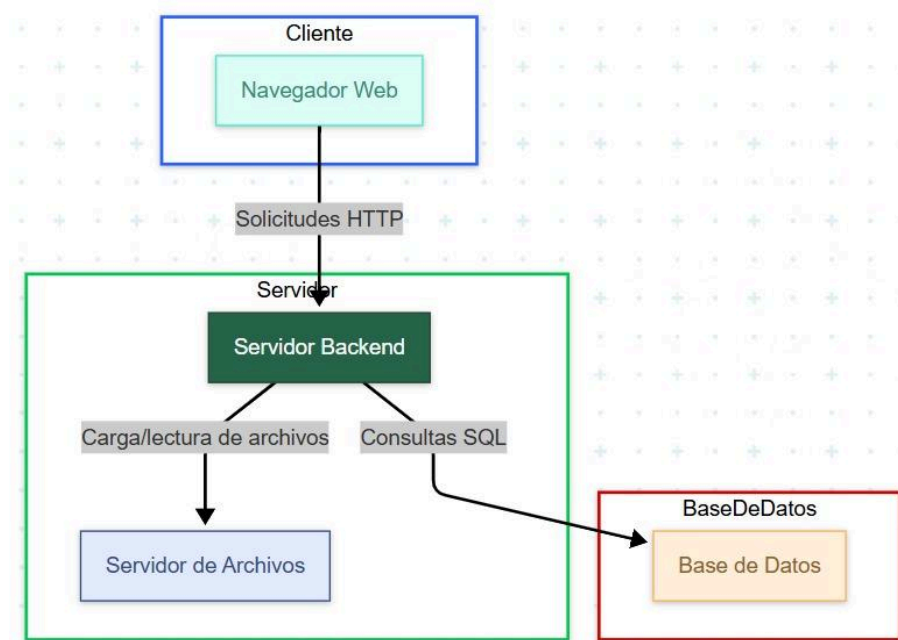
Método	Endpoint	Descripción
POST	/api/puestos	Crea un nuevo puesto disponible. (Solo empresa)

GET	/api/puestos	Devuelve una lista de puestos disponibles.
GET	/api/puestos/:id	Devuelve el detalle de un puesto específico.
PUT	/api/puestos/:id	Modifica el puesto (por ejemplo, cambiar estado). (Solo empresa)

Postulaciones

Método	Endpoint	Descripción
POST	/api/postulaciones	Permite a un estudiante postularse a un puesto. Se adjunta el CV como archivo, el cual es almacenado en el servidor, y se guarda su ruta en la base de datos. (Solo estudiante)
GET	/api/puestos/:id/postulaciones	Devuelve postulaciones de un puesto (Solo empresa dueña).
PUT	/api/postulaciones/:id	Cambia el estado de una postulación.

Diagrama de Despliegue



En el diagrama se muestran las direcciones principales de comunicación entre componentes. La interacción cliente-servidor es bidireccional: por cada solicitud enviada por el cliente, el servidor responde con datos o mensajes correspondientes

Componentes Principales

- Cliente (Frontend): navegador web usado por estudiantes y empresas.
- Servidor (Backend): recibe solicitudes, valida datos, maneja autenticación, gestiona lógica.
- Base de Datos: guarda perfiles, puestos, postulaciones.
- Servidor de archivos: guarda los CV cuando los estudiantes se postulan.

Flujo Principal

Los usuarios interactúan con la aplicación web (cliente), la cual se comunica con el backend mediante peticiones HTTP/HTTPS utilizando una arquitectura RESTful.

El backend se encarga de procesar las solicitudes, acceder a la base de datos y/o gestionar archivos (como los CVs), y devolver respuestas en formato JSON al cliente.