

DISEÑO DE SISTEMAS

Matias Camino

TP diseño de sistemas

Continuación del trabajo de Franco De Longhi

Contenido

Nuevos Requerimientos no funcionales	2
Nuevos Requerimientos funcionales	2
Diagrama de casos de uso:.....	3
Descripción de casos de uso:.....	4
Diagrama de clases actualizado con la nueva clase alerta.....	7
DER con la nueva entidad Alerta	8
Prototipo	9
Nuevos endpoints	10

Para la aplicación de pasantías:

Agregar alertas basadas en fechas. Por ejemplo, recordatorio de fin de pasantía. Otras alertas pueden ser en momentos claves, por ejemplo al aceptar una aplicación se notifica al estudiante y empresa. Las notificaciones deben ser enviadas al usuario por el canal que elija. En principio serán: email, directamente en la app y SMS.

Implementar manejo de estados (leída/no leída, enviada/fallida) y auditoría de envíos.

Nuevos Requerimientos no funcionales

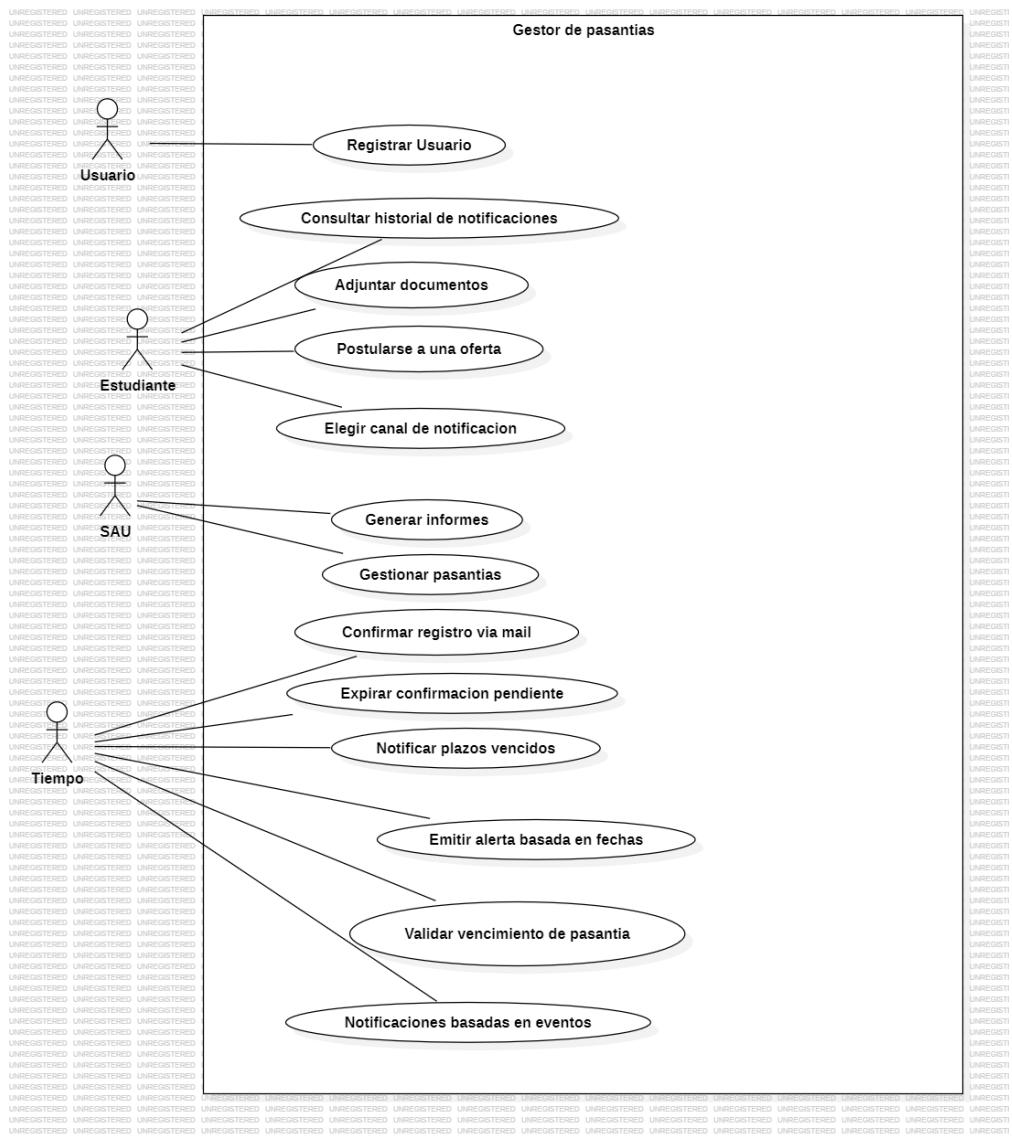
RNF-01 El sistema debe integrarse con SendGrid para el envío de notificaciones, alertas y validaciones a través del canal email.

RNF-02 El sistema debe integrarse con Twilio para el envío de notificaciones, alertas y validaciones que se realizaran a través del canal SMS.

Nuevos Requerimientos funcionales

- El sistema debe emitir alertas 7, 3 y 1 día antes de la fecha de inicio y finalización de una pasantía.
- El sistema debe permitir al usuario elegir sus canales de notificación preferidos.
- El sistema debe validar los datos de contacto ingresados (email válido, número de teléfono activo).
- El sistema debe calcular automáticamente las fechas de alerta al registrar una pasantía.

Diagrama de casos de uso:



Descripción de casos de uso:

Caso de Uso: Emitir alertas basadas en fechas	
ID: 1	Fecha:
Descripción: Se ejecuta una alerta de un determinado evento en base a una fecha determinada.	
Actores Principales: Tiempo	Actores Secundarios:
Flujo principal	Flujo Alternativo
1. El caso de uso comienza cuando se registra una pasantía, incluyendo su fecha de inicio y fecha de finalización.	
2. El sistema calcula cuando emitirá cada alerta en base a la fecha de inicio y finalización registrada en la pasantía. (Se emite una alerta 7, 3 y 1 día antes de dichas fechas)	
3. Verificación diaria: El sistema ejecuta diariamente un proceso que revisa todas las pasantías activas. Compara la fecha actual con las fechas programadas de alerta.	
4. Generación de mensaje: Si la fecha actual coincide con alguna fecha de alerta, el sistema genera un mensaje personalizado para el usuario correspondiente.	
5. Se notifica al estudiante/empresa a través de los canales que selecciono para ser notificado	
6. La alerta se envía con éxito y queda registrada como enviada.	6.1 La alerta no se logra enviar y queda registrada como fallida. 6.2 Fin CU
7. Fin CU	

Eventos claves que disparan notificaciones:

- Nueva publicación de oferta
- Confirmación de postulación
- Convocatoria a entrevista
- Cambio de estado de postulación

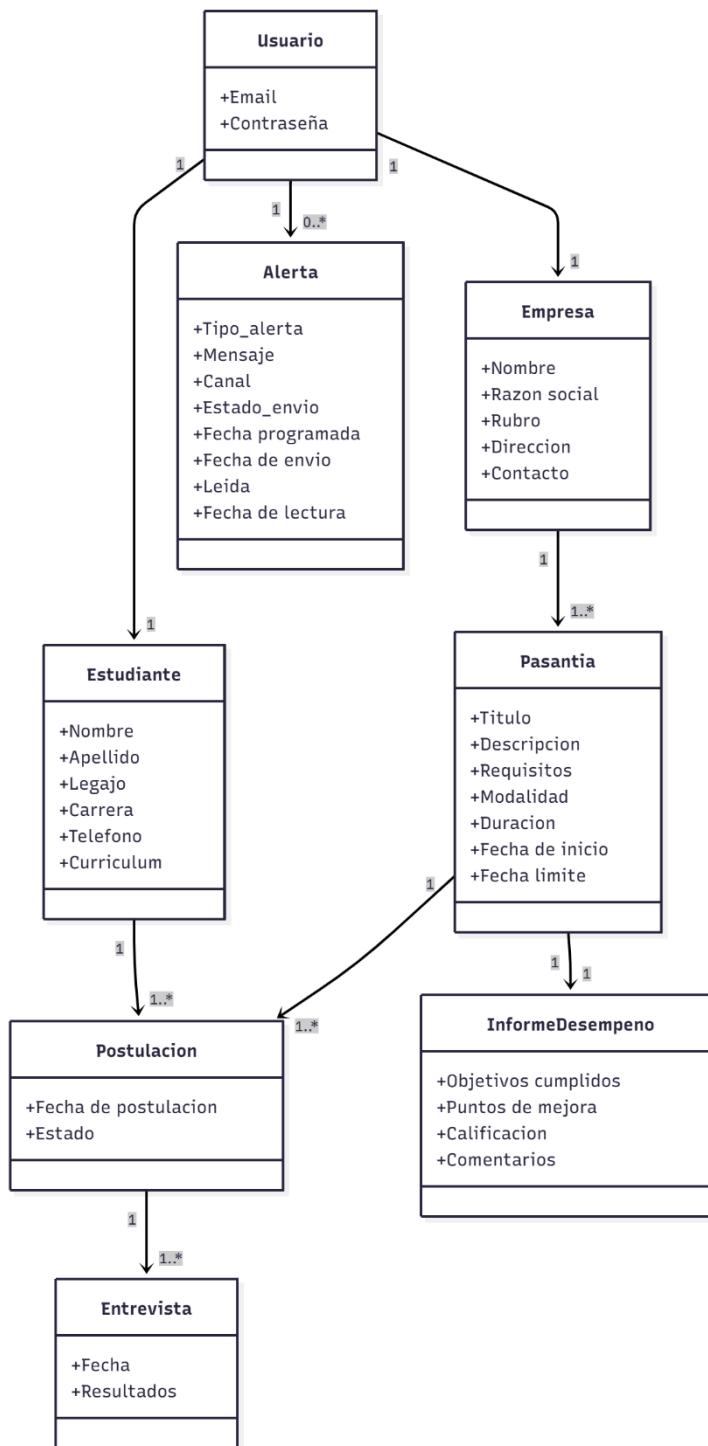
Caso de Uso: Emitir notificaciones basadas en eventos	
ID: 1	Fecha:
Descripción: Notificar automáticamente a los usuarios involucrados cuando ocurre un evento clave.	
Actores Principales: Tiempo	Actores Secundarios:
Flujo principal	Flujo Alternativo
1. El caso de uso comienza cuando el sistema monitorea continuamente la publicación de nuevas ofertas de pasantía y los cambios en el estado de las postulaciones de los estudiantes.	
2. El sistema detecta la ocurrencia de un evento clave.	
3. Genera la alerta correspondiente a la ocurrencia de ese evento detectado, incluyendo un mensaje personalizado para el destinatario.	
4. Se lo notifica al destinatario estudiante/empresa a través de los canales que selecciono para ser notificado	
5. La alerta se envía con éxito y queda registrada como enviada.	5.1 La alerta no se logra enviar y queda registrada como fallida.
6. Fin de CU.	

Caso de Uso: Elegir canal de notificación	
ID: 1	Fecha:
Descripción: Permitir al usuario configurar sus preferencias sobre cómo desea recibir las notificaciones del sistema (email, notificación en la app, SMS).	
Actores Principales: Usuario (estudiante/empresa)	Actores Secundarios:
Flujo principal	Flujo Alternativo
1. El caso de uso comienza cuando el usuario accede a la sección de notificaciones .	

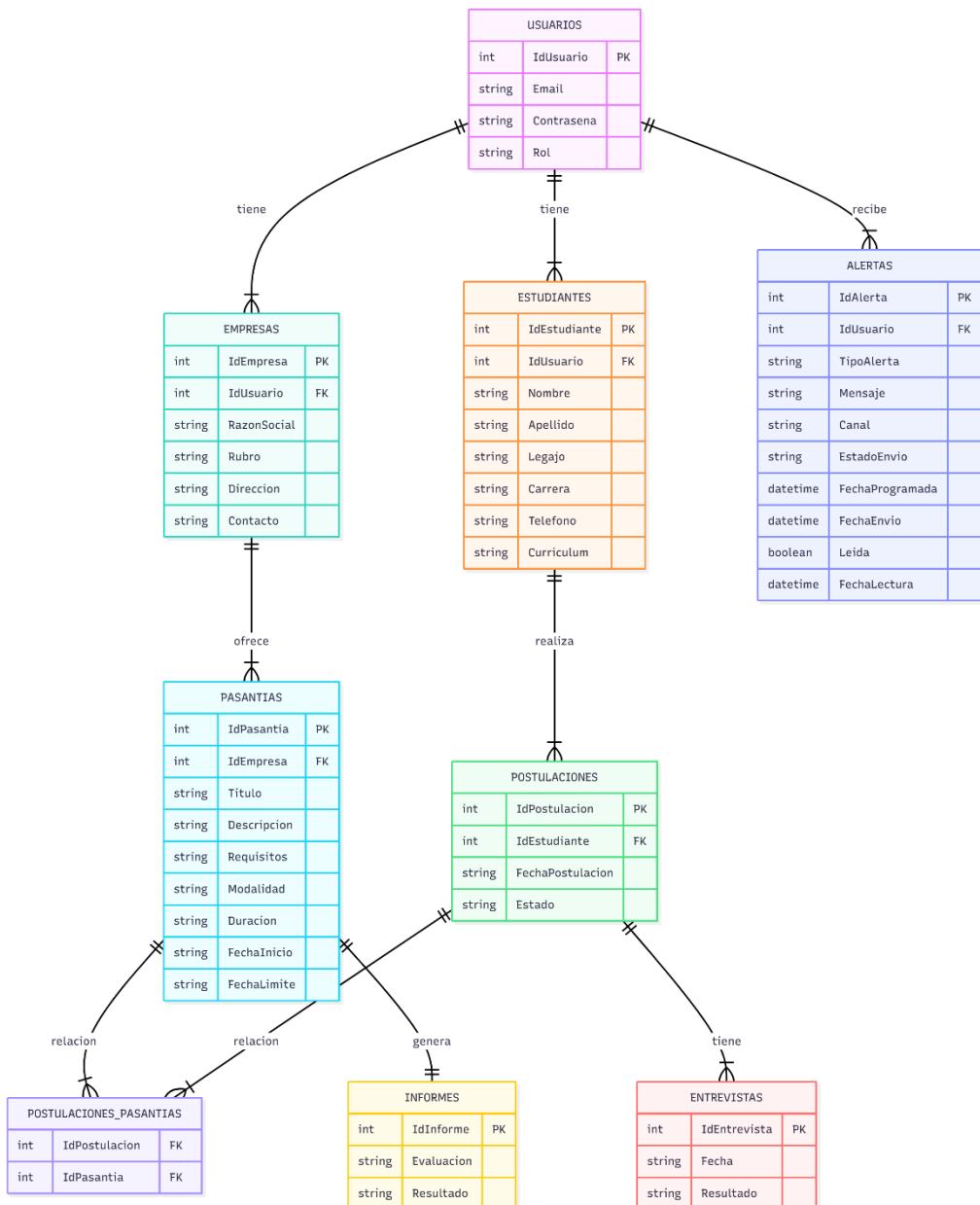
2. Selecciona la opción “preferencias de notificación”	
3. El sistema muestra los canales disponibles email, app, SMS.	
4. El usuario elige uno o mas canales como preferidos.	
5. El sistema valida los datos ingresados por ejemplo, email válido, número de teléfono activo.	
6. El sistema guarda la configuración en la base de datos.	
7. A partir de ese momento, todas las alertas se enviaran por los canales seleccionados.	

Caso de Uso: Consultar historial de notificaciones	
ID: 1	Fecha:
Descripción:	
Actores Principales: Usuario (estudiante/empresa)	Actores Secundarios:
Flujo principal	Flujo Alternativo
1. El caso de uso comienza cuando el usuario accede al perfil.	
2. El usuario accede a la sección de notificaciones	
3. EL usuario visualiza cada notificación con su estado (leída/no leído)	
4. Si el usuario entra a estas notificaciones queda marcada como leída.	

Diagrama de clases actualizado con la nueva clase alerta



DER con la nueva entidad Alerta



Se incorporó la entidad alerta en la base de datos

Los dichos atributos pueden tener los siguientes estados

Canal: canal utilizado (email,app,sms)

Estado envío: enviada, fallido, pendiente

Prototipo



Preferencias de notificación ▾

ⓘ Finalizacion pasantia
10:30hs 10/10/2025
Marcar como leido

ⓘ Finalizacion pasantia en tres dias
10:30hs 7/10/2025
Marcar como leido

ⓘ Finalizacion pasantia en 7 dias
10:30hs 3/10/2025
Leido

ⓘ Pasantia aprobada
10:30hs 3/10/2024
Leido



Preferencias de notificación ▾

Checkbox and Label

Email

SMS

NUmero de telefono Validar

APP

ⓘ Finalizacion pasantia
10:30hs 10/10/2025
Marcar como leido

ⓘ Finalizacion pasantia en tres dias
10:30hs 7/10/2025
Marcar como leido

ⓘ Finalizacion pasantia en 7 dias
10:30hs 3/10/2025
Leido

ⓘ Pasantia aprobada
10:30hs 3/10/2024
Leido

Nuevos endpoints

Método	Endpoint	Descripción
POST	/api/alertas/procesar	Ejecuta el proceso automático diario que revisa fechas programadas (inicio y fin de pasantías) y genera alertas.
POST	/api/eventos	Registra un evento del sistema (por ejemplo, nueva oferta, confirmación de postulación, convocatoria a entrevista, cambio de estado) y dispara la notificación correspondiente.
PUT	/api/notificaciones/preferencias/{usuarioId}	Permite al usuario configurar sus canales de notificación preferidos (email, app, SMS).
GET	/api/notificaciones/preferencias/{usuarioId}	Obtiene las preferencias de notificación del usuario.
GET	/api/notificaciones/historial/{usuarioId}	Muestra el historial completo de notificaciones recibidas, con su estado (leída/no leída).
PUT	/api/notificaciones/{notificacionId}/leida	Marca una notificación como leída cuando el usuario la abre.