



Universidad Austral de Chile

Facultad de Ciencias de la Ingeniería
Escuela de Ingeniería Civil en Informática

DESARROLLO DE MÓDULO DE NOTIFICACIONES E IMPLEMENTACIÓN DE MESA DE AYUDA PARA LA PLATAFORMA TRAC

Proyecto para optar al título de
Ingeniero Civil en Informática

PROFESOR PATROCINANTE:
VALERIA HENRIQUEZ NORAMBUENA
INGENIERA CIVIL EN INFORMÁTICA
MAGISTER EN DIRECCIÓN DE MARKETING
DIRECTO Y DIGITAL
DOCTORA EN SOFTWARE, SISTEMAS Y
COMPUTACIÓN

PROFESOR CO-PATROCINANTE
ELIANA SCHEIHING GARCÍA
INGENIERO CIVIL MATEMÁTICO
D.E.A. EN MATHEMATIQUE
DOCTEUR EN STATISTIQUE

PROFESOR INFORMANTE
NOMBRE DEL INFORMANTE
TÍTULOS Y GRADOS DEL INFORMANTE

JONATHAN NEFTALÍ GONZALEZ BORQUEZ
VALDIVIA – CHILE
2022

AGRADECIMENTOS

Quiero dedicar este trabajo a todas las personas que lo hicieron posible y agradecerles por su apoyo incondicional, no solo durante el desarrollo de este proyecto, sino que además durante toda mi etapa Universitaria.

En primer lugar, quiero agradecer a mi familia, a mi abuela Lucerina quién cada semana me llenaba de ánimos para continuar. A mi mamá Sofía, quién siempre ha estado presente para mí y jamás ha dejado de apoyarme, a mi hermana Lucerina por preocuparse por mi salud mental y darme consejos forjados con su experiencia, a mi hermano Benjamín por darme un segundo aire y motivarme a seguir intentándolo. A mis tíos y primos, no estaría aquí si no fuera por ustedes.

Quiero agradecer también a todos los docentes de la Universidad Austral de Chile, quienes no solo me enseñaron contenido, sino que además compartieron sus valores y motivaciones conmigo. Particularmente agradezco a mi profesora patrocinante Valeria Henríquez, quién siempre estuvo ahí para guiarme y mantenerme enfocado en los objetivos, gracias porque su apoyo no solo fue a nivel técnico, sino que además a nivel humano y emocional, doy gracias por su preocupación y su tiempo, gracias porque todo lo que aprendí junto a ella me hizo crecer a nivel personal y profesional, gracias por confiar en mí. Agradezco también a la profesora Eliana Scheihing por guiarnos y enseñarnos a abordar este arduo proceso, por su paciencia y comprensión.

Agradezco a los funcionarios y personas que trabajan en la Universidad, a la secretaria de escuela Yanett por siempre responder a nuestras dudas de la mejor manera y con la mejor disposición, a don Carlos por estar siempre atento a nuestras necesidades, a la tía del edificio 8 mil por su comida y su preocupación.

Finalmente agradezco a todas las personas que conocí, con las que compartí, aprendí de todos y cada uno, a mis amigos más cercanos, Manuel, Mauricio, Mónica, Edgar, Roberto, Diego, Sebastián, Sergio, Chester y Claudia les doy las gracias por siempre tenderme una mano y acompañarme en esta travesía.

INDICE

INDICE	I
INDICE DE TABLAS	III
INDICE DE FIGURAS.....	IV
RESUMEN.....	VI
ABSTRACT.....	VII
1 INTRODUCCIÓN	1
1.1 Motivación.....	2
1.2 Impacto	3
1.3 Objetivos	3
1.3.1 Objetivo general	3
1.3.2 Objetivos específicos	3
2 SITUACIÓN ACTUAL	4
2.1 TrAC-FID.....	4
2.1.1 Servicio TrAC-FID	5
2.1.2 Servidor de base de datos.....	8
2.1.3 Tecnologías de despliegue	8
2.2 Notificaciones.....	9
2.3 Mesa de ayuda	14
3 ANALISIS Y DISEÑO	16
3.1 Módulo de notificaciones	18
3.1.1 Requerimientos funcionales.....	18
3.1.2 Requerimientos no funcionales.....	19
3.1.3 Diseño de interfaz	19
3.1.4 Formato correo electrónico de notificación	21
3.1.5 Diseño de software.....	21
3.2 Mesa de ayuda	23
3.2.1 Requisitos funcionales	23
3.2.2 Otros requisitos	23
3.2.3 Tabla comparativa.....	24
3.2.4 UVdesk.....	27
3.2.5 Diseño de software.....	28
4 IMPLEMENTACIÓN Y HABILITACIÓN	30
4.1 Metodología de trabajo.....	30
4.2 Notificaciones.....	33
4.2.1 Análisis de tecnologías y lógica de TrAC-FID.....	33
4.2.2 Desarrollo.....	40
4.3 Mesa de ayuda	46
4.3.1 Manuales	46
4.3.2 Identificación de tecnologías.....	47
4.3.3 Implementación de mesa de ayuda UVdesk cómo servicio web	48

4.3.4	Adecuaciones	49
4.3.5	Configuración de la mesa de ayuda	50
4.3.6	Comunicación de TrAC-FID con mesa de ayuda UVdesk	60
5	VALIDACIÓN.....	66
5.1	Pruebas de usabilidad	66
5.1.1	Mesa de ayuda.....	70
5.1.2	Notificaciones	71
5.2	Mesa de ayuda: Pruebas de carga	72
5.3	Comentario de usuario.....	76
6	CONCLUSIONES	77
6.1	Cumplimiento de objetivos	77
6.2	Trabajos futuros.....	79
7	REFERENCIAS	81
8	ANEXOS	84
8.1	Anexo A: Revisión sistemática	84
8.2	Anexo B: Toma de requisitos	98

INDICE DE TABLAS

Tabla	Página
Tabla 1: Tipos de usuarios de casos de uso.....	17
Tabla 2: Requerimientos funcionales módulo de notificaciones	18
Tabla 3 : Requerimiento no funcionales de diseño	19
Tabla 4: Tabla de requerimientos funcionales mesa de ayuda.....	23
Tabla 5: Otros requisitos funcionales.....	24
Tabla 6: Requerimientos no funcionales de la mesa de ayuda	24
Tabla 7: Criterios de comparación de mesas de ayuda	25
Tabla 8: Criterios analizados versus plataformas	27
Tabla 9: Tipos de archivos y fechas aproximadas de carga	37
Tabla 10: Lista de actividades de prueba de usabilidad.....	68
Tabla 11: Resultados pruebas de usabilidad de pasillo mesa de ayuda	70
Tabla 12: Resultados pruebas de usabilidad de pasillo módulo de notificaciones	71

INDICE DE FIGURAS

Figura	Página
Figura 1: Vista general de la arquitectura de TrAC-FID	5
Figura 2: Dashboard TrAC-FID y sus componentes	6
Figura 3: Diagrama bloques fundamentales API	7
Figura 4: Ratio de apertura de correos por tipo (SendGrid Team, 2014)	10
Figura 5: Ejemplo de Tablero creado Trello	11
Figura 6: Correo notificación enviado por Trello	12
Figura 7: Ejemplo 1 - Correo notificación enviado por correoautomatico@uach.cl	12
Figura 8: Ejemplo 2 - Correo notificación enviado por correoautomatico@uach.cl	13
Figura 9: Ejemplo correo notificación enviado por sistemaacademico@uach.cl	13
Figura 10 : Ciclo de Trabajo del proceso Venture Design (Cowan, 2022)	17
Figura 11: Ventana Panel de administrador, pestaña Users.	20
Figura 12: Ventana Panel de administrador, pestaña Programs.	20
Figura 13: Prototipo de Interfaz Módulo de Notificaciones	21
Figura 14: Diagrama de secuencia usuario y sistema de notificaciones.	22
Figura 15: Diagrama simple de arquitectura de la mesa de ayuda.	28
Figura 16: Diagrama de secuencia de la interacción del usuario con la mesa de ayuda.	29
Figura 17: Ramas GitHub TrAC-FID	32
Figura 18: Tecnologías TrAC - TrAC v2 <i>Documentation</i>	33
Figura 19: Carpeta api proyecto TrAC-FID.	34
Figura 20: Carpeta <i>client</i> proyecto TrAC-FID.	35
Figura 21: Diagrama de secuencia de envío de correos de activación.	36
Figura 22: Estructura Tabla Parameters	37
Figura 23: Componente Parameters Interfaz TrAC-FID.	38
Figura 24: Estructura Tabla Risk_notification.	39
Figura 25: Panel lateral Listas Interfaz TrAC-FID.	39
Figura 26: Pestaña <i>Notifications</i> , panel de administrador.	40
Figura 27: Formato correo electrónico de notificación TrAC-FID.	42
Figura 28: Diagrama de secuencia del envío automático de notificaciones.	43
Figura 29: Diagrama de secuencia Verificar_estado_notificaciones().	44
Figura 30: Estructura Tabla Notifications Data	45
Figura 31: Diagrama de secuencia Analizar_respuesta().	45
Figura 32: Principales tecnologías del proyecto UVdesk	47
Figura 33: Arquitectura inicial mesa de ayuda	48
Figura 34: Cambios visuales a nivel de código en mesa de ayuda	49
Figura 35: Interfaz gráfica UVdesk con barra de desarrollador.	50
Figura 36: Interfaz gráfica UVdesk sin barra de desarrollador.	50
Figura 37: Paleta de color aplicada a Mesa de ayuda UVdesk	51
Figura 38: Configuración buzón UVdesk	52

Figura 39: Artículos de ayuda UVdesk TrAC-FID.....	53
Figura 40: Ejemplo de artículo - Ver información de estudiantes en panel de listas.....	54
Figura 41: Ejemplo Interfaz de configuración de Flujo de trabajo – Ticket Creado	55
Figura 42: Diagrama de actividad de flujo de trabajo de la creación de ticket.....	56
Figura 43: Diagrama de actividad de flujo de trabajo de Soporte responde a ticket.	56
Figura 44: Diagrama de actividad de flujo de trabajo de Director responde a ticket.....	57
Figura 45: Ejemplo respuesta guardada - Desbloqueo de cuenta	58
Figura 46: Tipos de tickets.....	59
Figura 47: Arquitectura con volúmenes Docker	60
Figura 48: Acceso a la mesa de ayuda sin sesión iniciada en TrAC-FID	60
Figura 49: Acceso a la mesa de ayuda con sesión iniciada en TrAC-FID.....	62
Figura 50: Diagrama de secuencia de acceso de usuario con sesión iniciada.....	63
Figura 51: Interfaz usuario Director.....	64
Figura 52: Interfaz usuario Soporte o administrador.	64
Figura 53: Arquitectura y comunicación TrAC-FID y UVdesk	65
Figura 54: Ejemplo afirmación en Formulario de Google	69
Figura 55: Representación de resultado SUS.....	70
Figura 56: Interfaz de usuario mesa de ayuda - Artículos de ayuda	71
Figura 57: Cantidad de carreras registradas en la base de datos de TrAC-FID.	72
Figura 58: Configuración de hilos JMeter	73
Figura 59: Configuración Data set JMeter.....	74
Figura 60: Configuración petición http JMeter.....	74
Figura 61: Resultado consulta SQL para verificar que se crearon los usuarios.....	75
Figura 62: Resultado resumen prueba de carga JMeter	75

RESUMEN

TrAC-FID es una plataforma web que emplea datos relacionados con la desempeño académico y laboral de los estudiantes para generar representaciones visuales accesibles y fáciles de entender. Con el fin de mejorar la experiencia de los usuarios, se han integrado dos mejoras en la plataforma, un sistema de alertas y un centro de asistencia o mesa de ayuda. Ambas herramientas tienen como objetivo maximizar la utilización de TrAC-FID por parte de los directores de escuela de la Universidad Austral de Chile.

El proceso de mejora se llevó a cabo utilizando la metodología ágil Scrum. En primer lugar, se implementó el sistema de alertas, que permite detectar situaciones de riesgo académico y notificar a los directores de manera automática. Durante la implementación, se prestó especial atención a los aspectos tecnológicos y lógicos, y se aprovechó la reutilización de componentes existentes.

Una vez concluido el módulo de alertas, se procedió a la implementación de la mesa de ayuda que se integró utilizando UVdesk como solución elegida después de evaluar diversas opciones del mercado. Esta plataforma ofrece un manejo de solicitudes mediante tickets y un gran abanico de funcionalidades de automatización de tareas como respuestas guardadas y flujos de trabajo que permiten facilitar el trabajo de los encargados de prestar el soporte. El proceso de integración presentó varios desafíos, pero gracias a un análisis exhaustivo y la implementación de soluciones tecnológicas, se logró una sinergia entre ambas plataformas.

Por último, se realizaron actividades de validación de los sistemas integrados, incluyendo pruebas de usabilidad para evaluar la satisfacción de los usuarios con las interfaces gráficas, pruebas de carga para evaluar la fiabilidad del sistema y recopilación de comentarios de los usuarios. Con estas actividades, se logró asegurar la eficacia de la mesa de ayuda y el módulo de notificaciones.

ABSTRACT

TrAC-FID is an advanced web platform that uses data related to student academic and employment performance to generate accessible and easy-to-understand visual representations. To improve the user experience, two improvements have been integrated into the platform, an alert system and an assistance center or helpdesk. Both tools aim to optimize the usefulness of TrAC-FID for school principals at the Universidad Austral de Chile.

The improvement process was carried out using the agile Scrum methodology. First, the alert system was implemented, which makes it possible to detect situations of academic risk and automatically notify directors. During the implementation, special attention was paid to technological and logical aspects, and the reuse of existing components was taken advantage of.

Once the alerts module was completed, the helpdesk system was implemented, which was integrated using UVdesk as the chosen solution after evaluating various market options. This platform offers request handling through tickets and a wide range of task automation functionalities such as saved responses and workflows that facilitate the work of those in charge of providing support. The integration process presented several challenges, but thanks to a thorough analysis and the implementation of technological solutions, an optimal synergy between both platforms was achieved.

Finally, validation activities of the integrated systems were carried out, including usability tests to evaluate user satisfaction with the graphical interfaces, load tests to evaluate the reliability of the system, and the collection of user comments. With these activities, it was possible to ensure the effectiveness and efficiency of the help desk and the notifications module.

1 INTRODUCCIÓN

El sistema de seguimiento a la progresión de Estudiantes de Carreras de Formación Inicial Docente (TrAC-FID), tal cómo se puede inferir de su nombre, tiene por objetivo presentar la progresión de cada alumno en las diferentes carreras de pedagogía impartidas por la Universidad Austral de Chile. Esta información es obtenida de distintas fuentes de datos, procesada y presentada mediante un *dashboard* o panel de control visual a los y las directores de escuela de la universidad (en adelante directores). Aunque este sistema logra centralizar en una única visualización la información que actualmente está disponible en múltiples sistemas y en consecuencia representa un aporte en término del uso del tiempo de los directores, la detección de situaciones riesgosas que requieren atención sigue siendo un desafío. Más concretamente, para analizar la situación académica de un estudiante y de una carrera, es necesario que el director revise periódicamente esta información directamente en el sistema. Por lo tanto, representa una sobrecarga de trabajo y acciones reactivas. Se espera que este proyecto de título logre resolver al menos en parte, problemas de este tipo. Para ello, se ha creado un módulo de notificaciones personalizadas que permite tomar medidas de manera preventiva y no reactiva cómo se realiza normalmente.

Además, TrAC-FID es un sistema que recibe solicitudes de cambios de manera continua, integración de nuevos componentes y mejoras en componentes ya existentes. Esta característica refleja dos puntos importantes, a) los desarrolladores se encuentran trabajando de manera iterativa cumpliendo con requisitos adaptados a las necesidades de los usuarios, y b) los usuarios deben adaptarse a los cambios en el sistema. Estos dos puntos acentúan la importancia de la comunicación que debe existir entre los desarrolladores y los usuarios o directores, esta comunicación debe permitir a los desarrolladores conocer las necesidades de los directores, y a los directores reportar sus necesidades. Actualmente esta comunicación se lleva a cabo por diferentes medios, cómo correo electrónico, reuniones, llamadas, entre otras. Esta falta de formalización de los procesos de contacto entre ambas partes provoca que las solicitudes no reciban la prioridad que merecen. Por ejemplo, una solicitud realizada mediante correo electrónico podría perderse en la bandeja de entrada del desarrollador. Por esta razón fue necesaria la habilitación de una mesa de ayuda virtual, en donde los directores puedan realizar sus solicitudes y los encargados de soporte puedan responder a estas, otorgándoles la importancia que se merecen.

1.1 Motivación

En 2020 la Universidad Austral de Chile se encontraba ejecutando el Plan de Implementación (PI) de la Formación Inicial Docente (FID). Dentro de sus hitos se encontraba la implementación de un sistema de seguimiento a la progresión de estudiantes de carreras de pedagogía, FID. Necesidad que llegó a cubrir, al menos en parte, TrAC, una plataforma que nació en el proyecto *Learning Analytics Latin America* o LALA Project y cuyo objetivo más general se alineaba de manera precisa con el del plan de implementación, fortalecer la calidad de la formación y entrega de la educación. En este punto nace TrAC-FID, una evolución de TrAC, ajustado a los requerimientos solicitados mediante un documento de términos de referencia, que contempló entre sus objetivos la creación de un sistema de alertas para los usuarios y la puesta en marcha de una mesa de ayuda. Este proyecto de implementación fue entregado a la empresa Puentes Digitales SPA.

TrAC-FID unifica los objetivos de ambos proyectos, tanto del PI FID como de TrAC y ofrece una solución moderna, creativa y enormemente beneficiosa para la universidad, en los aspectos relacionados con el seguimiento de la trayectoria académica de los estudiantes, apuntando a ofrecer una actualización que mejorará procesos que son parte de los nudos críticos identificados en el Programa de Implementación FID. TrAC-FID permite afrontar de manera más eficaz los desafíos normativos y contribuir al mejoramiento continuo de la educación en el país.

Entonces, la motivación de este proyecto está relacionada directamente con el inmenso potencial de la plataforma TrAC-FID y el gran aporte que podría ser en cuanto a el desempeño de los directores de carrera de formación inicial docente y de manera colateral un aporte para los estudiantes, docentes y a la universidad en general. La motivación de este proyecto es impactar positivamente, facilitando la absorción, relación o el encuentro entre la plataforma y las personas relacionadas, incluyendo además de los usuarios a los encargados de prestar el soporte.

El sistema de alertas o módulo de notificaciones favorece la velocidad de reacción frente a situaciones académicas irregulares de los alumnos, lo que se puede traducir en la detección de situaciones no tan solo de índole académico, sino que además situaciones propias de la vida los alumnos y que podrían estar causando problemas en su desempeño. Esto permite a la Universidad prestar ayuda, en caso de ser posible, demostrando el interés en el bienestar de los alumnos, propio del compromiso intelectual y ético con la preservación, generación y transmisión del conocimiento de la institución. La mesa ayuda acerca las necesidades de los usuarios con quienes estén encargados de responder a estas, haciendo uso de tecnologías modernas, facilitando y agilizando su comunicación.

1.2 Impacto

El impacto esperado de este proyecto es estrechar la relación que existe entre los usuarios y la plataforma TrAC-FID, mejorando o aumentando el valor que estos perciben a través de notificaciones en sus correos electrónicos con contenidos enriquecedores para el desempeño de sus labores, junto a una invitación a utilizar la plataforma, sumado a esto una mesa de ayuda virtual que permita atender las necesidades de los usuarios de manera ágil y amigable.

1.3 Objetivos

1.3.1 Objetivo general

Apoyar a los Directores de Escuela en la toma de decisiones oportunas respecto a situaciones irregulares de los estudiantes mediante el desarrollo de un módulo de notificaciones en TrAC-FID. Además, ofrecer una mesa de ayuda eficaz con un foco en los usuarios finales.

1.3.2 Objetivos específicos

- Conocer el estado actual de las tecnologías asociadas al desarrollo del sistema TrAC-FID.
- Identificar la lógica de los indicadores e información que es generada actualmente por el sistema y que será utilizada para generar las notificaciones.
- Desarrollar el módulo de notificaciones para alertar de situaciones potencialmente riesgosas.
- Determinar los requerimientos e identificar tecnologías compatibles con el sistema TrAC-FID que permitan el desarrollo de una mesa de ayuda.
- Desarrollar mesa de ayuda para facilitar la comunicación entre los usuarios y el equipo técnico que opera TrAC-FID.
- Validar el impacto real de las mejoras al sistema.

2 SITUACIÓN ACTUAL

Con el fin de cumplir con el primer y segundo objetivo, se realizó un análisis de la situación actual de TrAC-FID. En particular, se buscó información con respecto a las tecnologías y lógica asociadas a la plataforma y que fueron pilares fundamentales en la implementación y desarrollo del módulo de notificaciones. Además, en este capítulo se presenta un breve análisis de la situación actual de las notificaciones y alertas enviadas por diferentes aplicaciones populares de la industria, con el fin de plasmar y explicar el verdadero valor de estas funcionalidades, examinando desde aplicaciones como Trello hasta aquellas alertas enviadas por sistemas de la Universidad Austral de Chile. Esto último, fue analizado para replicar y mantener tanto el formato como la identidad de la institución en las notificaciones enviadas por el sistema implementando en TrAC-FID. Este proyecto incluyó la implementación de una mesa de ayuda, por lo que se hizo necesario un análisis de estas para determinar su alcance y funcionalidades, así como las consideraciones importantes para su implementación. Con esto se buscó ofrecer la solución más adecuada a los requisitos de los usuarios

2.1 TrAC-FID

TrAC-FID es un sistema que utiliza una gran cantidad de tecnologías, incluyendo lenguajes de programación, *frameworks* y servicios, estos, a su vez, poseen sus propias estructuras y lógicas de implementación. Estas tecnologías en conjunto colaboran para mantener a la plataforma en funcionamiento, y la labor de comprender cada una de ellas, desde la base de datos y los servicios que la “alimentan” hasta el despliegue visual de la aplicación, podría resultar complicado para alguien que no ha trabajado en su desarrollo o mantenimiento. Por ello, el análisis presentado a continuación se limita a aquellas tecnologías que permiten brindar una mejor comprensión del trabajo realizado en el desarrollo e implementación del módulo de notificaciones y la mesa de ayuda, omitiendo detalles de algunas tecnologías que no tienen una relación directa con los objetivos de este trabajo.

TrAC-FID se ejecuta en dos ambientes, uno de ellos es el de desarrollo, que permite realizar pruebas a nuevas funcionalidades y solo tienen acceso los encargados de mantener la plataforma. El segundo ambiente, es al cual acceden los usuarios finales de la plataforma, que en el caso de TrAC-FID son los directores de escuelas de pedagogía de la Universidad Austral de Chile y se conoce como ambiente de producción. Las descripciones presentadas a continuación están basadas en la versión de TrAC-FID que se encuentra en el ambiente de producción.

Para ilustrar de manera simple la labor de cada tecnología se ha creado un diagrama de los componentes que describe a grandes rasgos la arquitectura de TrAC-FID. La Figura 1

presenta una representación simplificada de cómo se relacionan y estructuran las partes que componen a TrAC-FID.

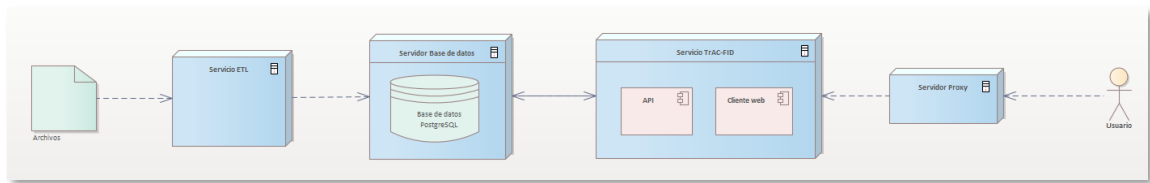


Figura 1: Vista general de la arquitectura de TrAC-FID

La arquitectura descrita en la Figura 1 consiste en varios servicios trabajando en conjunto. Analizándolo de derecha a izquierda, muestra que cuando el usuario accede a la URL o dominio de la plataforma web mediante su navegador, se genera una petición que es redirigida por el Servidor Proxy hacia el servicio de TrAC-FID éste se encarga de resolver y responder a las interacciones del usuario con las funcionalidades, estas interacciones requieren de información que es almacenada y consultada al Servidor de Base de Datos.

La base de datos es alimentada por un sistema ETL, cuyas siglas corresponde a *Extract* o Extraer, *Transform* o transformar y *Load* o Cargar, y cómo su nombre lo indica extrae, transforma y carga datos desde archivos obtenidos de diferentes fuentes.

2.1.1 Servicio TrAC-FID

El servicio TrAC-FID está compuesto por dos componentes, la aplicación o cliente web que corresponde al código asociado a los elementos visuales con los que interactúa el usuario y la API que resuelve y administra las consultas a la base de datos y ofrece otros servicios que permiten el funcionamiento de la plataforma. Ambos componentes, aplicación web y API, fueron desarrollados utilizando **TypeScript**, este es un lenguaje de programación libre y de código abierto desarrollado y mantenido por Microsoft, y es un superconjunto de **JavaScript**, uno de los lenguajes de programación más utilizados del momento para desarrollar aplicaciones web (Microsoft, s.f.).

La principal característica de **TypeScript** es que opera de manera fuertemente “tipada” y orientada a objetos, en otras palabras, permite y requiere la definición de los tipos de variables con los que se va a trabajar, por ejemplo, si una variable es un número, y opera como tal, esta puede definirse con el tipo *number*, o si es una palabra o texto puede definirse con el tipo *string*, estos son tipos simples de variable, pero pueden crearse estructuras más complejas. El definir una variable antes de usarla permite conocer de antemano las operaciones que se pueden realizar con esta. Además, por esta razón se conoce a TypeScript como un lenguaje de tipado estático, en donde el tipo de variable se conoce en tiempo de compilación, a diferencia de JavaScript en donde se conoce cuando

se ejecuta el programa o en tiempo de ejecución. TypeScript sacrifica flexibilidad por una mejor estructura, lo que se traduce en un lenguaje ideal para proyectos complejos y con una gran cantidad de variables involucradas, cómo es el caso de TrAC-FID.

2.1.1.1 Aplicación web

El cliente o aplicación web se encarga de renderizar la interfaz de TrAC-FID y posee una arquitectura basada en componentes. Este tipo de arquitectura divide en pequeñas piezas la lógica de la interfaz del usuario, en el caso de TrAC-FID cada elemento que se muestra en el *dashboard* corresponde a un componente, tal como se muestra en la Figura 2.

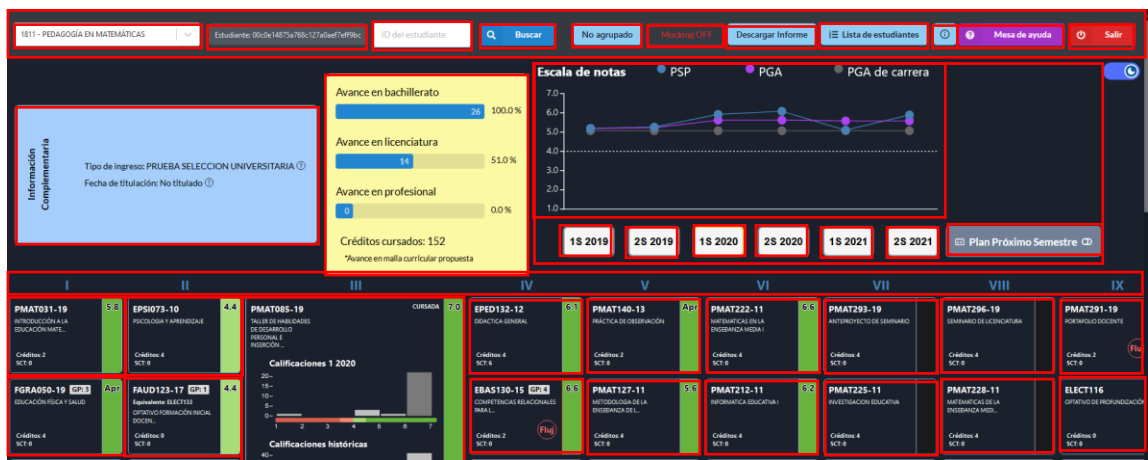


Figura 2: *Dashboard* TrAC-FID y sus componentes

Las líneas rojas en este caso delimitan cada componente, que a su vez se encuentra constituido por otros subcomponentes. Este método de encapsulación introduce la segunda tecnología a presentar, **React**.

React es una librería de JavaScript desarrollada por Facebook y disponible como código abierto (Facebook Open Source, s.f.). Su enfoque principal es la reutilización de componentes, lo que permite una mayor eficiencia en el desarrollo de interfaces de usuario. Además, React ofrece un sistema de control de estado mediante variables y funciones denominadas *hooks*, lo que permite una mayor flexibilidad en la gestión de la interfaz. Además, tiene un enfoque de rendimiento optimizado, recargando solo los componentes necesarios para mejorar la velocidad de despliegue de la interfaz de usuario.

2.1.1.2 API

Las siglas API corresponden a *Application Programming Interfaces*, se trata de un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos aplicaciones de software a través de un conjunto de reglas (Fernández, 2019).

La API de TrAC-FID se encarga de gestionar y responder a las solicitudes realizadas por el Cliente, en sus archivos se encuentra plasmada la lógica de negocios detrás de cada componente gráfico. El diagrama de bloques fundamentales de la Figura 3 corresponde a una representación detallada de los scripts o archivos de código involucrados en el funcionamiento de la API.

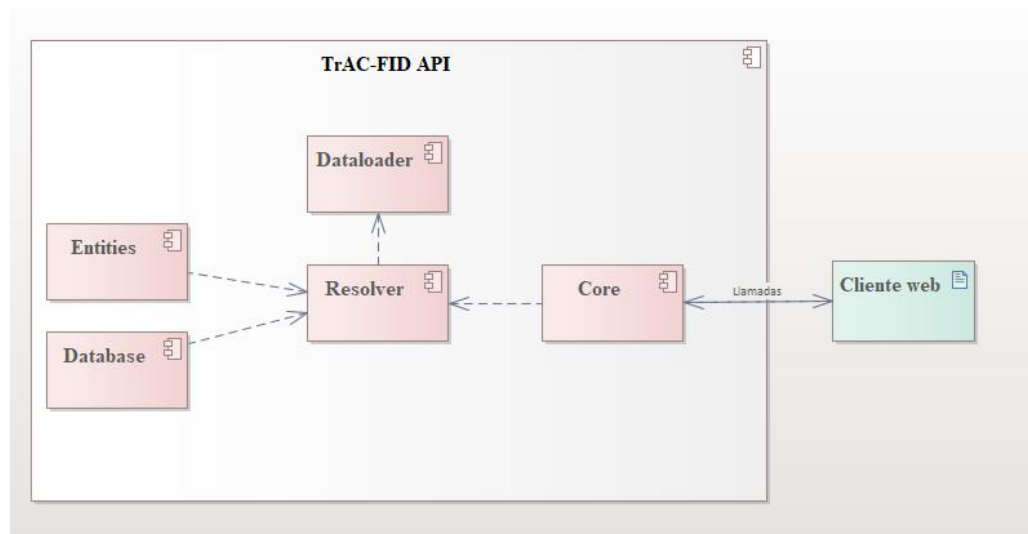


Figura 3: Diagrama de componentes fundamentales API

- **Bloque Principal o Core.** Orquesta las funcionalidades necesarias para el correcto funcionamiento de la API.
- **Entidades o *Entities*.** Corresponden a las definiciones de las clases y estructuras de los objetos con las cuales trabajará la API.
- **Base de datos o *Database*.** Corresponden a las definiciones de la estructura de las tablas de datos reales.
- **Resolutores o *Resolvers*.** Hacen el enlace entre las estructuras definidas en las entidades y base de datos, en otras palabras, satisface las estructuras definidas por las entidades con los datos obtenidos desde las tablas definidas en la base de datos.
- **Cargadores de información o *Dataloaders*.** Se encargan de almacenar la información consultada a la base de datos para su reutilización ayudando a optimizar la cantidad de llamadas realizadas.

Cómo se ha mencionado anteriormente TrAC-FID trabaja con grandes cantidades de datos, por lo que es necesario hacer las búsquedas lo más específicas posibles, evitando la descarga de información innecesaria, siendo este uno de los principales objetivos de **GraphQL**. Dicha tecnología es utilizada por la API para gestionar de manera más eficientes las llamadas a la base de datos.

GraphQL es un lenguaje de uso gratuito y fue desarrollado por Facebook, presenta una idea parecida a la de TypeScript con la definición de campos agrupados en tipos y funciones asociadas a cada uno de ellos. (The GraphQL Foundation., s.f.)

2.1.2 Servidor de base de datos

El servidor de base de datos funciona cómo un sistema independiente del servicio de TrAC-FID. Se encarga de almacenar la información que es extraída desde archivos de diferentes fuentes dentro de la institución, además de gestionar y responder a las solicitudes de datos realizadas por el servicio de TrAC-FID. Para la administración de los datos se utiliza **PostgreSQL**, un sistema de base de datos que combina SQL con otras funcionalidades que lo transforman en una herramienta poderosa a la hora de trabajar con grandes cantidades de datos.

PostgreSQL al igual que las anteriores tecnologías es de uso gratis y Open Source, fue desarrollada por *The PostgreSQL Global Development Group*. Es un sistema independiente de los lenguajes y tecnologías utilizadas en el desarrollo *FrontEnd* y posee una gran compatibilidad. (The PostgreSQL Global Development Group, s.f.)

2.1.3 Tecnologías de despliegue

De manera complementaria a las tecnologías con que está construido TrAC-FID, existen otras que permiten su funcionamiento y dentro las cuales se encuentran:

- **NodeJS: Entorno de ejecución**

Los entornos de ejecución son tecnologías que permiten la compilación y ejecución de una aplicación. Estos reúnen todas las utilidades necesarias para transformar el código en aplicaciones. En el caso de TrAC-FID el entorno de ejecución es **NodeJS**. Dicha tecnología, al igual que TypeScript se basa en JavaScript y es un recurso *open-source*, es decir que puede ser utilizado de manera gratuita. Dentro de sus características más notables destaca la ejecución de eventos de manera asíncrona, es decir que se pueden ejecutar diferentes porciones de código o funciones en un mismo instante, lo que impacta positivamente la velocidad de despliegue de interfaces y la velocidad de respuesta a las peticiones de datos (Acibeiro, 2022).

- **Nginx: Proxy Server**

Nginx es un conjunto de soluciones o productos ofrecida por f5.com, una empresa de soluciones tecnológicas, dentro de las cuales se encuentra su servidor web *open-source*, el cual es utilizado por la plataforma TrAC-FID. (Kinsta, 2021)

Corresponde al Servidor Proxy descrito en la Vista general de la arquitectura de TrAC-FID de la Figura 1 y tiene por objetivo redirigir las llamadas al dominio hacia la plataforma.

- **Docker: Virtualización**

Para evitar problemas como la incompatibilidad de librerías, diferencias de versiones entre los ambientes de producción y desarrollo, configuraciones de sistema operativo, entre otros, se utiliza la tecnología de virtualización **Docker**.

Docker permite crear una imagen independiente del equipo (computador) en la cual se ejecuta la aplicación. Mediante la creación de un contenedor que funciona como una máquina virtual, todos los parámetros asociados pueden ser ajustados por el usuario antes de iniciar el contenedor, como definir el sistema operativo con el cual se construirá, las versiones de las librerías que se instalarán y utilizarán. Además, permite la automatización de comandos, lo que hace más fácil el despliegue de la plataforma. (Red Hat, s.f.)

Para resumir este análisis, las tecnologías mencionadas apuntan a la eficiencia, velocidad de carga, características obligatorias para sistemas complejos como lo es TrAC-FID y que tienen por objetivo ofrecer a los usuarios las mejores de las experiencias.

2.2 Notificaciones

Noticias, alertas, eventos, recordatorios, y muchos otros son los motivos de las notificaciones que se reciben día a día los usuarios, ya sea en correos electrónicos, smartphones o computadores, estas funcionalidades se han convertido en parte vital de la relación que existe entre los sistemas de información y los usuarios. La importancia de las notificaciones queda perfectamente ilustrada en la siguiente cita:

“Los clientes se registran, usarán su producto por un tiempo y luego abandonarán. A veces, la rotación ocurre cuando su empresa simplemente se les escapa, se olvidaron de usted, otras veces sucede porque nunca experimentaron el momento "AH-HA", el momento en que su producto simplemente hace clic y funciona en su mente. Los correos electrónicos de notificación cierran esa brecha...”

(SendGrid Team, 2014)

El equipo de SendGrid en su artículo “*Why the Notification is the Most Important Email You Can Send for Growth*” analiza los motivos y la importancia de los correos de notificaciones. Indican que estos se ubican en segundo lugar en la categoría de mails abiertos, por sobre mensaje de recordatorios, sociales, boletas y promociones, cómo se puede ver en la Figura 4.

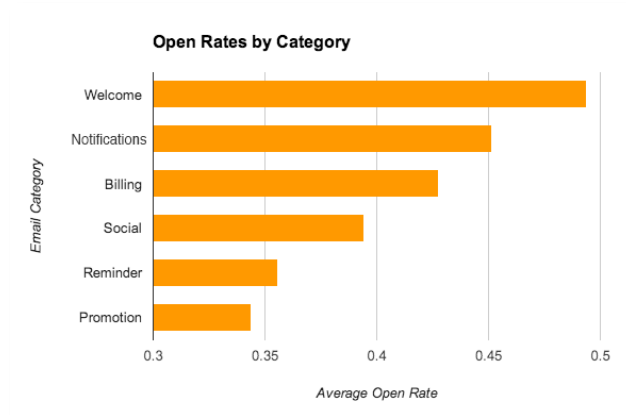


Figura 4: Ratio de apertura de correos por tipo (SendGrid Team, 2014)

Resulta natural la pregunta ¿A qué se debe este comportamiento? Las notificaciones corresponden a información en tiempo real y además tienen relación con la cuenta del usuario, en otras palabras, los usuarios muestran interés por información que los involucra directamente.

El objetivo es entonces identificar información que resulte de valor para el usuario. En dicho artículo se dan cómo ejemplo las notificaciones de nuevos seguidores utilizadas por Twitter, apelando a la vanidad del usuario.

“A la gente le gusta ver nuevos seguidores y comentarios, y quién ha visitado su perfil.”

Josh Elman (Khailee, s.f.)

Josh Elman, antiguo líder de crecimiento de Twitter se dio cuenta de que la retención de usuarios aumentaba considerablemente cuando estos conseguían 30 seguidores. Lo que hizo fue promover este comportamiento mediante correos electrónicos.

Una vez identificada la información que puede ser de valor para el usuario surge otra interrogante ¿Cómo se debe entregar esta información?, Brent Sleeper del Equipo de

MessageBird responde a esta pregunta e identifica ocho puntos clave que se deben cumplir en un buen email de notificación (Sleeper, 2018):

1. La línea de asunto es lo más importante.
2. Las buenas notificaciones transmiten una sola idea.
3. Las alertas deben ir directamente al grano con una llamada directa a la acción.
4. Las notificaciones por correo electrónico deben ser altamente personalizadas.
5. Las mejores notificaciones refuerzan la experiencia del usuario.
6. Proporcione una opción de suscripción o cancelación de suscripción cuando corresponda.
7. Envíelos de manera oportuna.
8. Incluir información de contacto.

A continuación, y a modo de ejemplo se analiza cómo se aplica lo descrito anteriormente en Trello y en sistemas de notificaciones de la Universidad Austral de Chile.

Trello es una herramienta que mediante tableros y tarjetas permite organizar y gestionar proyectos, un ejemplo de un Tablero creado en la plataforma se puede ver en la Figura 5.

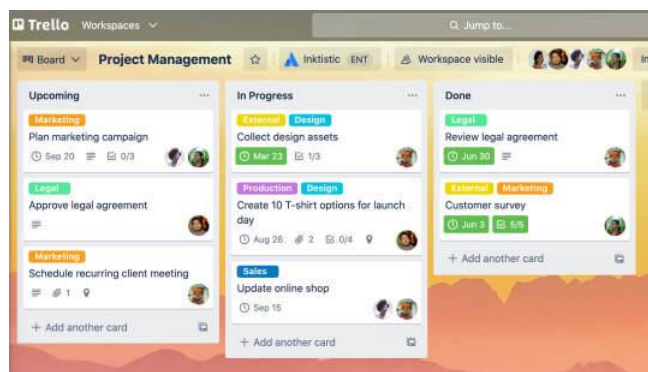


Figura 5: Ejemplo de Tablero creado Trello

Utiliza las notificaciones para indicar al usuario que se han realizado acciones en la plataforma que son de su interés, las que son enviadas dentro de la aplicación y mediante un correo electrónico al usuario. Con respecto al formato, el mensaje presenta en el asunto la acción realizada y detalles sobre esta, además aprovecha el cuerpo para agregar información que podría ser de interés para el usuario, agrega la opción de cambiar las preferencias de correo electrónico para detener o modificar la suscripción a las notificaciones, este análisis se puede ver gráficamente en Figura 6.



Figura 6: Correo notificación enviado por Trello

En el caso particular de la Universidad Austral de Chile esta posee un dominio propio al cual se encuentran asociados los remitentes de correos de alertas, entre los cuales existen: correoautomatico@uach.cl y sistemaacademico@uach.cl. El primero se utiliza para el envío de alertas, invitaciones, afiches y recordatorios de eventos en general, el segundo se encarga de informar a los alumnos sobre notas obtenidas en las asignaturas que se encuentran cursando.

El email correoautomatico@uach.cl no posee una estructura estrictamente definida, cómo se muestra en los ejemplos de la Figura 7 y la Figura 8.

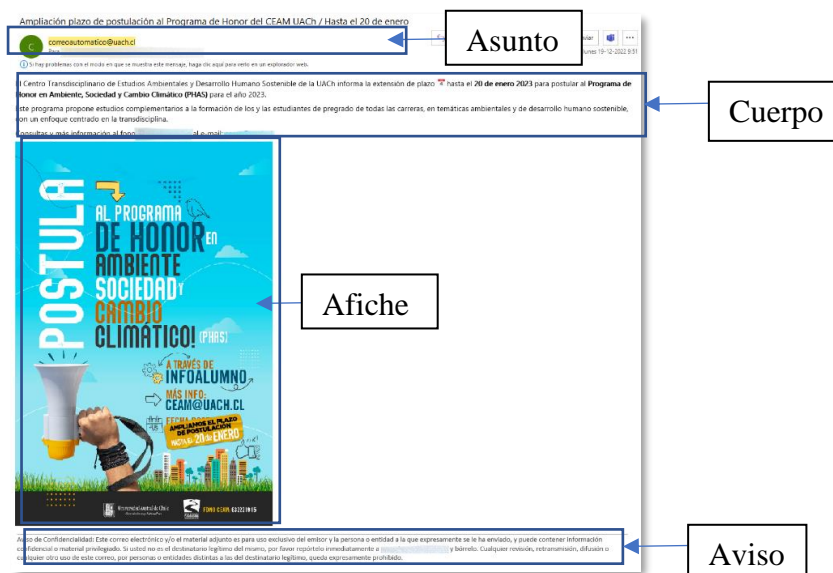


Figura 7: Ejemplo 1 - Correo notificación enviado por correoautomatico@uach.cl

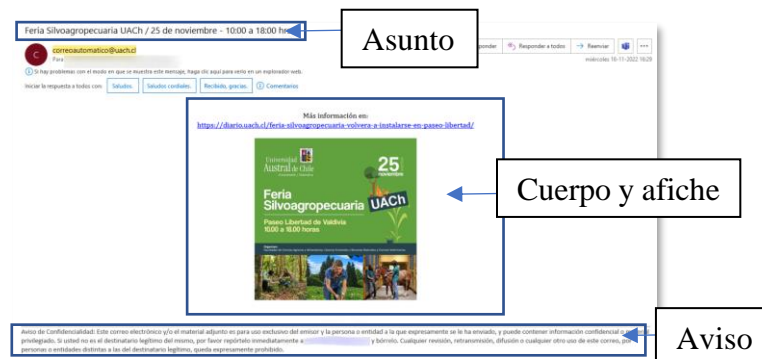


Figura 8: Ejemplo 2 - Correo notificación enviado por correoautomatico@uach.cl

Además del asunto, que presenta un breve resumen del motivo, el correo puede ser dividido en dos partes, el cuerpo del mensaje, que varía de un correo a otro, y el aviso de confidencialidad, que se mantiene constante.

El cuerpo concentra toda la información asociada al evento, cómo fechas, lugares, asociados, medios de contacto y otros. Además, en el caso de existir un afiche se muestra la imagen correspondiente. El aviso de confidencialidad indica la naturaleza del mensaje, así como una alerta que permite identificar a los usuarios que no fueron contemplados en el envío de la notificación.

Por su parte, los correos enviados por sistemaacademico@uach.cl poseen una estructura definida y rígida, cómo se muestra en la Figura 9.

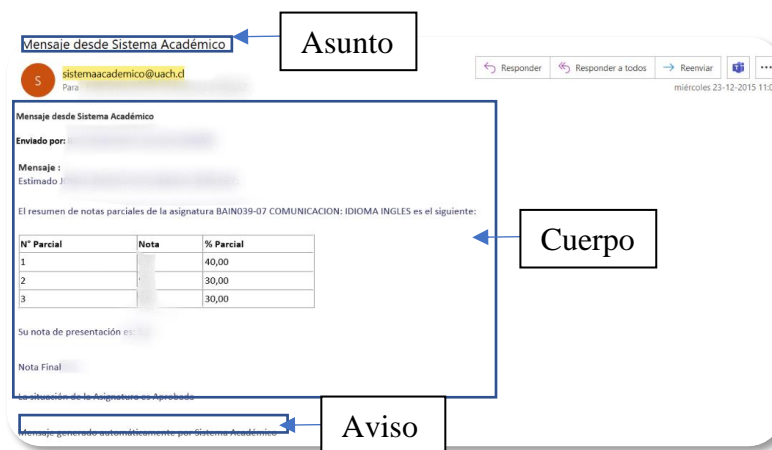


Figura 9: Ejemplo correo notificación enviado por sistemaacademico@uach.cl

El correo puede ser dividido en cuatro partes, el encabezado que indica el origen del mensaje, el emisor que corresponde al profesor que ha ingresado la nota, el mensaje que

incluye la o las notas obtenidas y la asignatura a la que pertenecen y finalmente la naturaleza del mensaje.

En conclusión, las notificaciones son una excelente manera de estrechar la relación con los usuarios, aumentar el valor que estos perciben, pero es necesario identificar la información que le resulta de interés y saber cómo entregarla. Además, se puede inferir que los eventos que desencadenan los envíos deben ser situaciones que están ocurriendo en tiempo real dentro del sistema.

2.3 Mesa de ayuda

Las mesas de ayuda son una herramienta esencial en la relación entre los usuarios y las plataformas a las que participan. Estos sistemas complementarios tienen como objetivo resolver problemas y dificultades que puedan surgir en esa relación. Las funcionalidades de una mesa de ayuda pueden variar ampliamente, desde un sencillo correo electrónico o número de teléfono, hasta sistemas más sofisticados con gestión de usuarios, solicitudes mediante tickets y eventos automatizados. El objetivo final es encontrar un sistema de mesa de ayuda que se adapte a las necesidades y demandas de los usuarios.

En el mercado actual existen una gran variedad de opciones, algunas de código abierto u *open-source* y otras de pago, algunas ofrecen soluciones mixtas con acceso a funcionalidades simples de manera gratis y mediante suscripciones pagadas se puede acceder a las funcionalidades más complejas, varias de estas alternativas son analizadas más adelante en el documento.

Resulta vital conocer las consideraciones que se debe tener a la hora de implementar una mesa de ayuda. Mediante una revisión sistemática realizada por el autor se concluyó que se deben considerar las siguientes:

- **FAQ o Panel de preguntas frecuentes:** Este apartado permite disminuir la cantidad de solicitudes provocadas por las mismas interrogantes.
- **Integración y análisis de datos:** Las estadísticas y los análisis permiten a los encargados conocer en métricas el comportamiento de las solicitudes.
- **Contexto, horarios:** Un punto importante es conocer a quienes está dirigida la mesa de ayuda y quienes la van a operar, así como los horarios de disponibilidad y necesidad.
- **Experiencia de usuarios:** Las mesas de ayuda no deben complejizar la relación entre los usuarios y las plataformas a las que pertenecen, sino qué hacerla más amena.
- **Conocer las debilidades y fortalezas de la mesa de ayuda, así como del personal involucrado:** Saber que funcionalidades debe ofrecer la mesa de

ayuda para que en conjunto con sus operarios puedan dar abasto a todo el espectro de solicitudes de los usuarios.

Para más detalles sobre la revisión sistemática realizada vea el Anexo A: Revisión sistemática.

Para finalizar, cabe destacar que en el caso particular de la Universidad Austral de Chile existe actualmente una mesa de ayuda, a la cual se puede acceder mediante un correo electrónico y un número de teléfono. Posee horarios establecidos y resuelve solicitudes realizadas por todos los usuarios de la UACH, académicos, administrativos e investigadores (Universidad Austral de Chile, 2016). TrAC-FID al no ser una herramienta institucionalizada, no puede ser gestionada por esta mesa de ayuda central y, en consecuencia, se requiere de una solución particular para ella.

3 ANALISIS Y DISEÑO

Los requerimientos de diseño para este proyecto fueron generados a partir del documento de solicitud realizado por el Plan de formación Inicial Docente descrito en el capítulo de introducción. En este documento se exponía la necesidad de implementar un sistema de alertas. A continuación, se presentan fragmentos del documento que destacaban esta necesidad.

- “... desarrollar un sistema de alertas y la generación de información.”
- “El sistema debe considerar la detección temprana de estudiantes en riesgo académico, estableciendo modalidades de alerta a las escuelas bajo parámetros consensuados con estas.”

Además, se menciona la necesidad de una mesa de ayuda, asistencia para los usuarios del sistema y manuales de usuario:

- “Operativizar sistema web, desarrollando manual de usuario y capacitaciones según nivel de usuario”.
- “Establecer mesa de ayuda y mantención de errores durante el periodo establecido para La utilización del sistema web”.
- “... Mesa de ayuda en horarios establecidos en forma semanal en mismo periodo de mantención de errores”.

A partir de estos extractos, fue posible inferir varios requisitos del sistema a implementar, pero aún no era del todo claro el alcance y necesidades específicas de los usuarios reales.

Utilizando la metodología de *Venture Design Process*, desarrollada por Alexander Cowan, se logró especificar con precisión las funcionalidades necesarias. Este proceso de diseño proporciona un marco de trabajo con varios pasos y enfoques específicos que permitieron centrar los esfuerzos en las funcionalidades que realmente importan para los usuarios. La Figura 10 describe el ciclo de trabajo del proceso Venture Design.

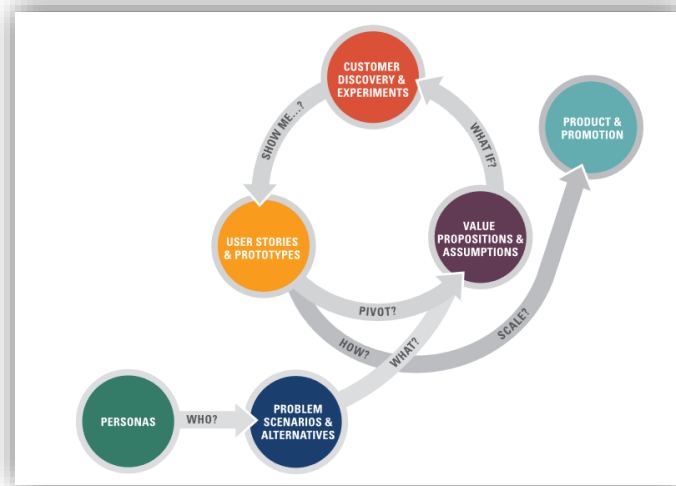


Figura 10 : Ciclo de Trabajo del proceso Venture Design (Cowan, 2022)

En un comienzo se describieron a las *Personas* o arquetipos, correspondientes a los tipos de usuario que interactúan con el sistema, tal como se muestra en la Tabla 1.

Tabla 1: Tipos de usuarios de casos de uso

Identificador	Descripción
Director(a)	Director de Escuela perteneciente al área de formación inicial docente de la Universidad Austral de Chile.
Soporte	Encargado de administrar TrAC-FID y responder a las necesidades de los Directores con respecto al sistema.

Partiendo de las inferencias generadas a partir del documento de solicitud FID, se generaron posibles escenarios y alternativas, terminando con hipótesis sobre los requisitos del sistema que fueron contrastadas con entrevistas realizadas a potenciales usuarios reales. Este ciclo de trabajo permitió asegurar la dirección del proyecto y evitar la generación de funcionalidades que no hubieran llegado a satisfacer las necesidades reales de los usuarios, además de revelar perspectivas no consideradas previo a las entrevistas. Para conocer más detalles sobre este proceso revise el Anexo B: Toma de requisitos.

Luego de que se contrastaron las hipótesis con las necesidades reales, se identificaron historias de usuario para describir cada uno de los requisitos funcionales. Estas historias siguen el formato "Cómo [Tipo de usuario] necesito [Funcionalidad] con la finalidad de [Motivación]", lo que permite identificar al usuario que participa, su objetivo o necesidad y la motivación detrás de ello en el contexto de uso real. Junto con esto, se sumó una

validación o criterio de aceptación por cada una para saber si se ha cumplido con la necesidad descrita.

Una vez generadas las historias de usuarios, correspondiente a los requisitos, se tomaron vías de solución diferentes para el desarrollo del sistema de alertas y la implementación de una mesa de ayuda. En el primero caso, se continuó con el diseño de interfaz y de software a partir de los requisitos y en el segundo se utilizaron los requisitos de usuarios junto a requisitos técnicos y no funcionales para generar un método de pruebas de aceptación orientado a analizar y valorar varios softwares *open-source* de mesas de ayuda existentes en el mercado. A continuación, se presentan los requisitos y diseños separados por el tema al que pertenecen: mesa de ayuda o módulo de notificaciones.

3.1 Módulo de notificaciones

3.1.1 Requerimientos funcionales

La Tabla 2 presenta las historias de usuarios identificadas en los requisitos del módulo de notificaciones.

Tabla 2: Requerimientos funcionales módulo de notificaciones

Cómo [Tipo de usuario]	Necesito [Funcionalidad]	Con la finalidad de [Motivación]	Criterios de aceptación
Director(a)	Ser alertado cuando se realicen cargas de nuevos datos.	No tener que revisar estas fechas periódicamente en el sistema.	El correo enviado informa las fechas de actualización y los tipos de datos actualizados.
			Las fechas se condicen con la información que se encuentra en la base de datos.
Director(a)	Ser alertado(a) con las situaciones académicas irregulares de mis alumnos	Poder tomar acciones preventivas.	El correo enviado informa las situaciones irregulares de los estudiantes.
			La información permite a los usuarios acceder rápidamente a la plataforma, además explica cómo visualizar las situaciones irregulares en el sistema.
Soporte	Enviar correos electrónicos a modo de notificaciones a los usuarios.	Informar a los Directores que los datos se han actualizado y además alertarlos sobre las situaciones de riesgo.	El sistema genera notificaciones personalizadas para cada director, considerando los programas o carreras pertenecientes a su escuela.
			El sistema muestra un registro de las notificaciones enviadas
			El sistema muestra un mensaje indicando si se han realizado o no el envío de los emails.

También, como se ha mencionado previamente en este documento, el módulo de notificaciones fue una extensión de TrAC-FID. Esto quiere decir que el desarrollo de esta parte del proyecto estaba sujeto a las tecnologías utilizadas para implementar la plataforma. La mayoría, si no es que todas las más relevantes, de estas tecnologías fueron presentadas en el análisis de la situación actual.

3.1.2 Requerimientos no funcionales

La Tabla 3 se refiere a los requisitos no funcionales de diseño y se fundamentan en la necesidad de mantener el mismo aspecto visual de las otras componentes y no generar un estilo o formato de diseño que se pueda ver poco orgánico.

Tabla 3 : Requerimiento no funcionales de diseño

Requisito	Descripción
Conservar Diseño	Utilizar el formato, diseño y paletas de colores de las componentes existente en TrAC-FID.

3.1.3 Diseño de interfaz

Puesto que el envío de notificaciones es una funcionalidad exclusiva de usuarios administradores se consideró que la mejor opción para localizar el nuevo módulo de notificaciones era en la interfaz de configuración o panel de administrador, para así mantener las operaciones en una misma pestaña y evitar crear nuevos flujos o ventanas de trabajo que puedan entorpecer sus labores.

El panel de administrador posee un selector de pestañas el cual permite navegar por distintas ventanas; *Users*, *Programs*, *BaseConfig*, *Tracking*, *Data* y *Feedback*, cómo se puede observar en la Figura 11 y la Figura 12 , donde en la primera se muestra el diseño de interfaz de la pestaña Usuarios y en el segundo el diseño de la pestaña de Programas.

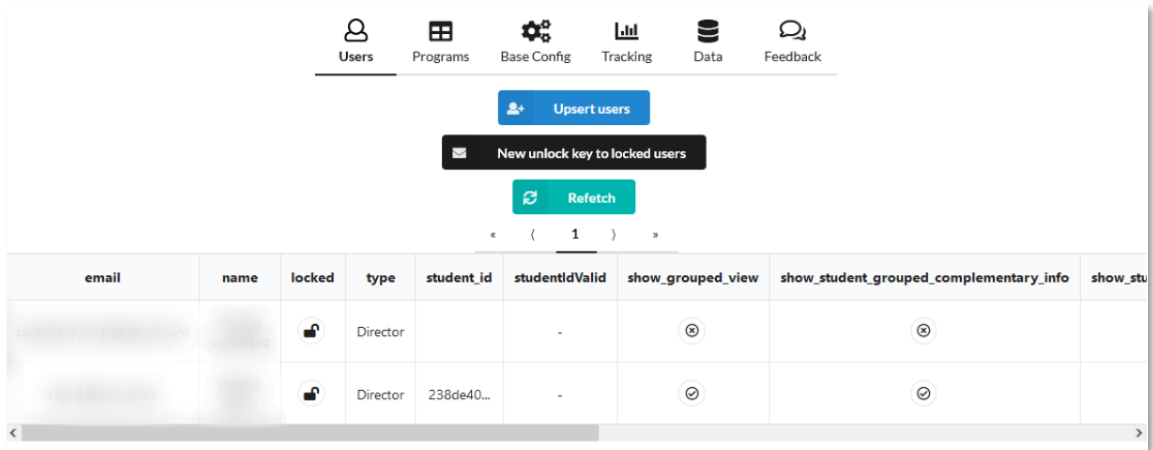


Figura 11: Ventana Panel de administrador, pestaña *Users*.

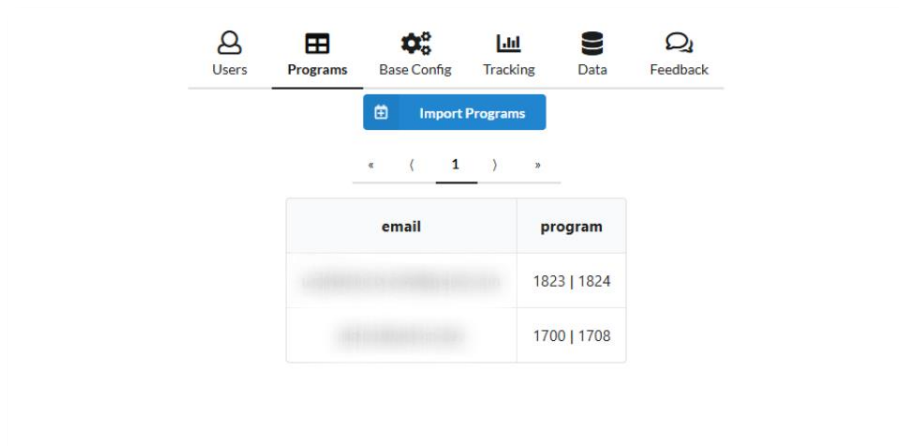


Figura 12: Ventana Panel de administrador, pestaña *Programs*.

Tomando como base el mismo formato de diseño de las demás pestañas se desarrolló una propuesta de diseño para la nueva pestaña de Notificaciones, en donde se consideró la necesidad de un botón para realizar el envío y una tabla en donde se muestre el registro de las notificaciones enviadas, ver Figura 13.

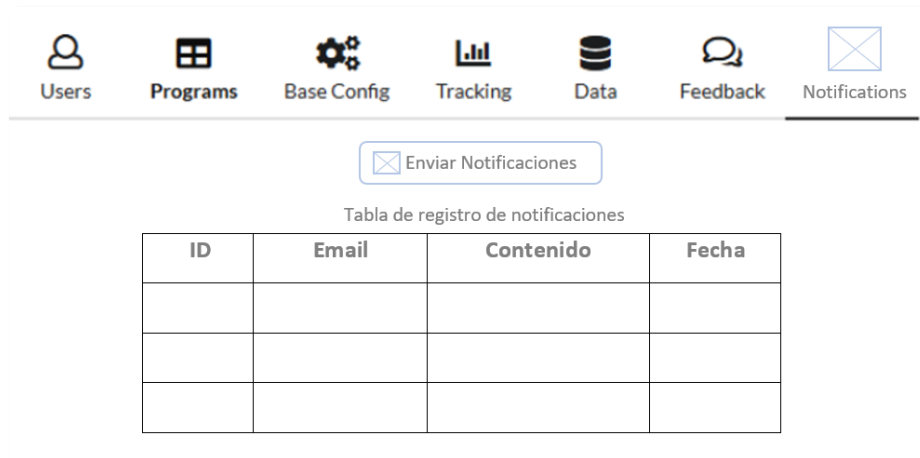


Figura 13: Prototipo de Interfaz Módulo de Notificaciones

Esta tabla debe contener un id identificando el registro, el email al cual fue enviado, su contenido y la fecha de envío.

3.1.4 Formato correo electrónico de notificación

Además, debieron tomarse decisiones con respecto a el formato y contenido del correo electrónico que sería enviado a los usuarios, y se definieron los siguientes puntos cómo necesarios en la estructura del correo:

- Asunto indicando que existen novedades en los datos de la plataforma.
- Encabezado o saludo.
- Informe de nuevos tipos de cargas de datos y sus respectivas fechas.
- Informe de situaciones de riesgo identificadas en la última carga de datos.
- Invitación a visitar la plataforma.
- Despedida.
- Pie de correo indicando que el correo fue generado automáticamente y que no debe ser respondido.

3.1.5 Diseño de software

Se definió un diseño simple de cómo debería comportarse el envío de notificaciones, que puede verse en la siguiente Figura 14.

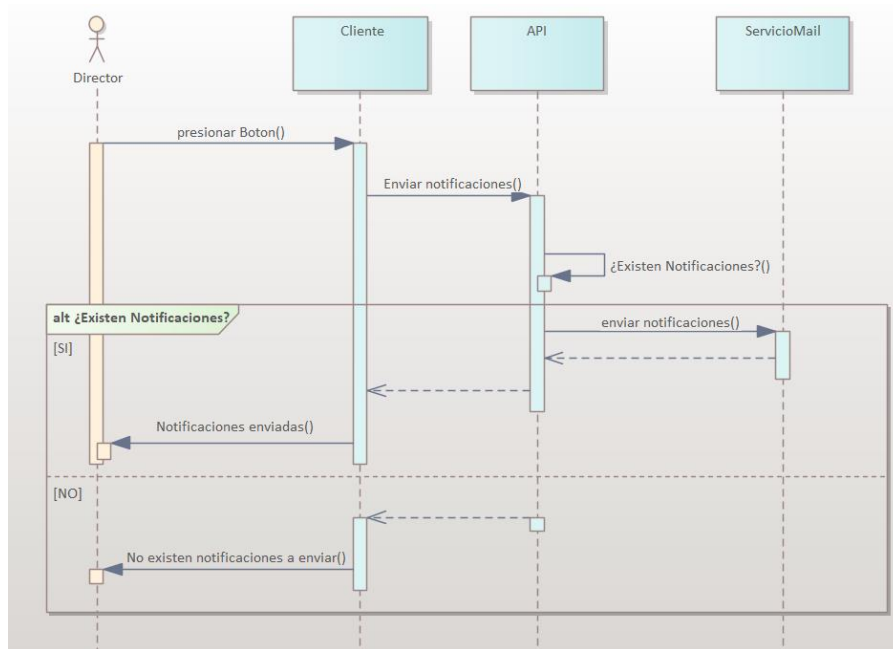


Figura 14: Diagrama de secuencia de interacción de usuario con sistema de notificaciones.

La secuencia de pasos es la siguiente, al presionar el botón el sistema debe verificar si existen usuarios a los cuales deba notificarse la actualización de datos o alertas sobre situaciones de riesgo, en caso de existir debe enviar correos electrónicos con la información pertinente, luego debe retornar un mensaje al usuario indicando el resultado de su solicitud.

El sistema debe considerar la lectura de todos los usuarios que cumplan con los requisitos necesarios para ser notificados. Además, debe ser capaz de notificar las actualizaciones de nuevas cargas de datos, agregando dinámicamente las situaciones de riesgo si están existen, es decir que podría haber mensaje que solo involucren la alerta de actualización de datos puesto que podrían no existir situaciones de riesgo.

3.2 Mesa de ayuda

3.2.1 Requisitos funcionales

La Tabla 4, contiene las historias de usuario identificada en los requisitos de la mesa de ayuda.

Tabla 4: Tabla de requerimientos funcionales mesa de ayuda

#	Cómo [Tipo de usuario]	Necesito [Funcionalidad]	Con la finalidad de [Motivación]	Criterios de aceptación
1	Director(a)	Compartir mis sugerencias	Poder aportar a crear un sistema más amigable.	Se pueden enviar comentarios en la plataforma.
2	Director(a)	Agregar imágenes a mis sugerencias o errores encontrados en el sistema	Poder explicar mejor el contexto y contenido de mi solicitud.	Los Directores pueden adjuntar imágenes en los comentarios/sugerencias/errores reportados. El Soporte puede ver las imágenes.
3	Director(a)	Compartir el error que encuentro en el sistema.	Aportar en la mejora del sistema.	Se pueden reportar errores en la plataforma.
4	Director(a)	Editar/eliminar mi comentario.	Modificar un comentario que envié o eliminarlo.	Se puede eliminar un comentario enviado o realizado. Se puede editar un comentario enviado o realizado.
5	Director(a)	Ver la sección de preguntas frecuentes	Evitar solicitar ayuda en temas que son comunes y acceder de manera más rápida a la información que necesito.	El director puede visualizar la sección de preguntas frecuentes.
6	Soporte	Conocer la opinión/solicitud de los Directores sobre el sistema.	Mejorar su experiencia en la plataforma.	El Soporte puede visualizar los comentarios realizados por los Directores.
7	Soporte	Ver las solicitudes de ayuda de los Directores.	Resolver sus dudas o prestar ayuda.	El Soporte puede visualizar las solicitudes de ayuda. El Soporte puede responder a las solicitudes de ayuda.
8	Soporte	Crear y editar Respuestas a preguntas frecuentes.	Evitar consultas repetitivas y enfocarme en las necesidades más específicas.	El Soporte puede crear una pregunta frecuente y agregar una respuesta. El Soporte puede editar o eliminar una pregunta frecuente.

Además, se consideró otro requerimiento funcional asociado a las formas de acceso a la mesa de ayuda Tabla 5.

Tabla 5: Otros requisitos funcionales

Tipo	Requisito
Acceso	<p>Deben existir dos vías de acceso a la mesa de ayuda para los usuarios:</p> <ul style="list-style-type: none"> • El usuario ha iniciado sesión en TrAC-FID y desea acceder a la mesa de ayuda. • El usuario no ha iniciado sesión en TrAC-FID.

3.2.2 Otros requisitos

La Tabla 6 se refiere a los requerimientos que se debían considerar a la hora de implementar el entorno de la mesa de ayuda.

Tabla 6: Requerimientos no funcionales de la mesa de ayuda

Tipo	Requisito
Despliegue	La mesa de ayuda debe ser una aplicación funcionando independiente de TrAC.
Implementación	Se espera que el sistema sea una implementación de alguna plataforma <i>open-source</i> y que no considere el desarrollo desde cero.
Concurrencia	El sistema debe ser capaz de soportar la cantidad máxima de directores de escuela de la universidad conectados de manera paralela.

3.2.3 Tabla comparativa

Considerando los requisitos antes descritos, se decidió optar por un sistema ya existente que cuente con las funcionalidades principales de mesa de ayuda. Las peticiones de ayuda, reportes y comentarios se simplificaron en el concepto de “*ticket*” donde el “tipo de *ticket*” indique la naturaleza de la solicitud. Además de la creación de *tickets* se consideró que era estrictamente necesario que el sistema permitiera la creación y manejo de preguntas frecuentes, apartado en el que además deberían incluirse los manuales de usuario.

Luego de analizar los requisitos y las consideraciones antes descrita se desarrolló una tabla de análisis, en donde se contrastaban las necesidades del sistema versus las plataformas analizadas. En la Tabla 7 se describen los criterios de análisis utilizados.

Tabla 7: Criterios de comparación de mesas de ayuda

Tipo	Criterio	Descripción
Detalles	Link	URL principal.
	Repositorio	URL del repositorio de código.
	Demo	URL de la demo o prueba.
Requisitos Técnicos	OpenSource	¿Es una plataforma gratuita?
	Tecnologías	Tecnologías principales, por ejemplo: Lenguaje de programación, frameworks, etc.
	Host	¿Es una plataforma web?, ¿Se puede levantar en un servidor propio?
	Usuarios	¿Permite la creación y manejo de tipos de usuarios?
Tickets	Apartado tickets	¿Posee un apartado de tickets, tanto como para los Directores cómo para el Soporte?
	Crear ticket	¿Permite la creación de ticket?
	Tipo ticket	¿Se pueden editar los tipos de tickets?
	Ver ticket	¿Directores pueden ver sus tickets? ¿Soportes pueden ver los tickets de los Directores?
	Editar/Borrar ticket	¿Se pueden editar o borrar los tickets?
	Responder ticket	¿Soportes pueden responder a tickets?
	Envío de Correo electrónico	¿Se envía un correo electrónico cuando ocurre algún evento relacionado a los tickets? (creación, respuesta, edición o eliminación)
	Estadísticas	¿Muestra estadísticas e indicadores sobre el flujo de tickets?
FAQ	Apartado FAQ	Posee un apartado de preguntas frecuentes.
	Creación de pregunta y respuesta	Permite la creación y edición de preguntas frecuentes.
Costos asociados		Posee algún otro costo asociado, cómo planes, pago por funcionalidades.

Se analizaron las siguientes plataformas:

- **Zammad Community**

Zammad es un sistema de atención al cliente/servicio de asistencia al cliente de código abierto basado en la web con muchas características para administrar la comunicación con el cliente a través de varios canales como teléfono, Facebook, Twitter, chat y correos electrónicos. Se distribuye bajo la versión 3 de la Licencia Pública General GNU

AFFERO (GNU AGPLv3). (Zammad, 2023). Zammad ofrece una buena solución de manejo de tickets, su código se encuentra disponible de manera gratuita lo que permite la implementación en un host propio. Lamentablemente no posee un apartado asociado a preguntas frecuentes.

- **Zoho**

Zoho Mail es una solución de correo electrónico empresarial confiable y segura que está personalizada según las necesidades de comunicación de su organización. Gracias a las funciones de colaboración mejorada, es mucho más que solo una bandeja de entrada. (Zoho Corporation, 2023). Zoho Mail es una plataforma web que dispone de un gran número de herramientas que favorecen la organización de eventos y la comunicación dentro de un grupo de trabajo, esta comunicación se realiza mediante correos electrónicos que se muestran en una interfaz llamativa y amigable. Lamentablemente requiere ser pagado luego del tiempo de prueba.

- **Google Workspace**

Al igual que en G Suite, todos los planes de Google Workspace proporcionan correo electrónico personalizado para empresas y ofrecen herramientas de colaboración, como Gmail, Calendario, Meet, Chat, Drive, Documentos, Hojas de cálculo, Presentaciones, Formularios, Sites y muchas otras más en Workspace (Google, LLC, 2023). Google Workspace, al igual que Zoho, ofrece una plataforma web con una amplia variedad de herramientas. Permite la integración con otras funcionalidades ofrecidas por Google. Sin embargo, se consideró que su implementación y uso podrían requerir ciertos conocimientos técnicos, y su característica modular podría resultar en un proceso engorroso para los usuarios.

- **UVdesk**

UVdesk ofrece una solución de mesa de ayuda basada en SaaS¹ y de código abierto para facilitar el proceso de soporte general para brindar el mejor servicio al cliente. (Webkul Software , 2023). Esta fue la opción seleccionada para implementar pues se consideró que satisfacía en gran medida todos los requisitos.

La Tabla 8 ofrece un análisis resumido del trabajo de comparación entre plataformas realizado.

¹ Software como un Servicio o SaaS es un modelo de distribución de software donde el soporte lógico y los respectivos datos que maneja se alojan en los servidores de un proveedor, cuyo acceso es a través de Internet.

Tabla 8: Criterios analizados versus plataformas

Plataforma	Zammad Community	Zoho	Google Workspace	UVdesk
Link	zammad.org	zoho.com	workspace.google.com	UVdesk.com
Repositorio	https://github.com/zammad/zammad	No posee	No posee	UVdesk.com/es/demo/
Demo	No posee	No posee	No posee	Sí
OpenSource	Sí	Sí	Sí	Sí
Tecnologías	Ruby	Desconocido	Desconocido	PHP
Host	Se puede ejecutar en servidor propio.	Plataforma web	Plataforma web	Se puede ejecutar en servidor propio.
Usuarios	Permite gestión.	Permite gestión.	Permite gestión.	Permite gestión.
Apartado tickets	Sí	Sí	Sí	Sí
Crear ticket	Sí	Sí	Sí	Sí
Tipo ticket	Sí	Sí	Sí	Sí
Ver ticket	Sí	Sí	Sí	Sí
Editar/Borrar ticket	Sí	Sí	Sí	Sí
Responder ticket	Sí	Sí	Sí	Sí
Envío de Correo electrónico	Sí	Sí	Sí	Sí
Estadísticas	No			Sí
Apartado FAQ	No	Sí	No	Sí
Creación de pregunta y respuesta	No	Sí	No	Sí

3.2.4 UVdesk

UVdesk es una herramienta de gestión de tickets de soporte y ayuda al cliente que también incluye una funcionalidad de creación de una "wiki" o repositorio de información para los usuarios. El código fuente está disponible gratuitamente en GitHub y cuenta con una página web con documentación para ayudar en la instalación y el uso de sus características.

UVdesk destacó por su gran facilidad de instalación, y su gran cantidad de funcionalidades. Aunque algunas de estas funcionalidades pueden parecer redundante o innecesarias al principio, su implementación práctica demuestra que son esenciales para facilitar el trabajo de los operadores y usuarios del sistema. Estas funcionalidades incluyen la posibilidad de guardar respuestas automáticas, un sistema interno que lee correos electrónicos y genera tickets automáticamente en la mesa de ayuda, así como la posibilidad de enviar un correo automático al usuario cuando se genera un ticket.

3.2.5 Diseño de software

La Figura 15 ilustra cómo se diseñó el sistema de mesa de ayuda para ser independiente del servidor de TrAC-FID. Se consideró que, debido a que la plataforma de mesa de ayuda tiene una vida útil menor que TrAC-FID, su nivel de madurez en aspectos como la seguridad y estabilidad es inferior, por lo que además las llamadas necesarias para comunicar las aplicaciones y satisfacer los requerimientos deberán ser solamente realizadas desde TrAC-FID a la mesa de ayuda y no en sentido contrario.

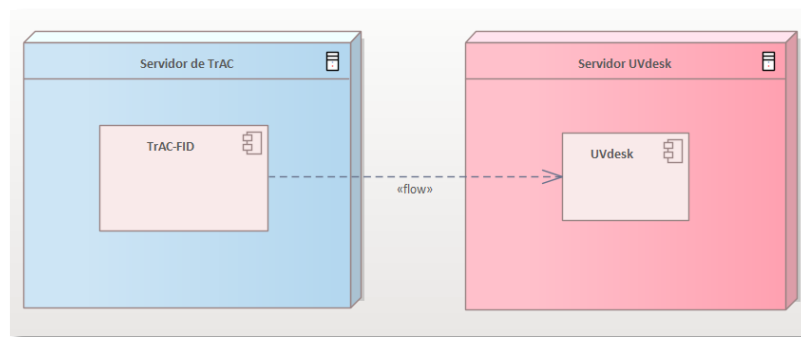


Figura 15: Diagrama simple de arquitectura de la mesa de ayuda

La Figura 16 retrata el comportamiento que deberán seguir las plataformas web (TrAC-FID y UVdesk) dentro del navegador. El usuario siempre comenzará el uso de la mesa de ayuda desde la plataforma TrAC-FID, esta se encargará indicar al navegador web que necesita abrir otra pestaña con la mesa de ayuda.

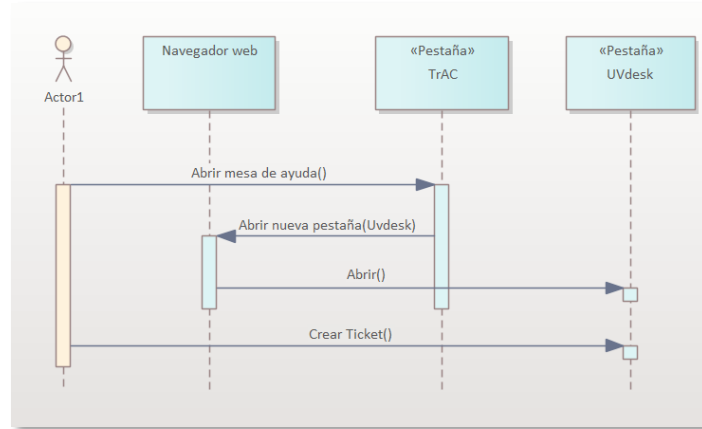


Figura 16: Diagrama de secuencia de la interacción del usuario con la mesa de ayuda

Además, se consideró que los usuarios deberán tener dos caminos de acceso a la mesa de ayuda, el primero cuando ya poseen una sesión iniciada en TrAC-FID y el segundo cuando no. Este último caso podría darse para usuarios que olvidaron su contraseña, o no la han creado y que desean crear un ticket para solicitar ayuda. En resumen, en el primer caso la nueva pestaña de mesa de ayuda se deberá mostrar en la interfaz con la sesión creada para el usuario, en donde podrá ver sus tickets y su información asociada, y en el segundo caso deberá mostrar una interfaz sin una sesión asociada.

4 IMPLEMENTACIÓN Y HABILITACIÓN

Esta sección está dedicada a describir las actividades llevadas a cabo durante el proceso de implementación y habilitación de ambos módulos. Se inicia con una breve descripción de la metodología de trabajo utilizada, luego se presenta el desarrollo del módulo de notificaciones y finalmente la implementación y puesta en marcha de la mesa de ayuda UVdesk.

4.1 Metodología de trabajo

Para la planificación y gestión del trabajo realizado se utilizó la metodología de desarrollo ágil Scrum, y se consideraron los siguientes elementos:

1. **Product Backlog.** Conjunto de requisitos contenidos en historias de usuarios, los cuales pueden ir cambiando durante el transcurso del proyecto.
2. **Sprint.** Iteración de duración establecida en el cual el equipo de Scrum trabaja en las tareas asignadas.
3. **Sprint Backlog.** Conjunto de tareas o historias de usuario que deben ser completadas dentro del sprint
4. **Sprint Planning Meeting.** Reunión de planeación del Sprint en donde se definen las tareas que serán agregadas al Sprint Backlog.

En Scrum se definen tres tipos de roles de trabajo:

1. **Scrum Master.** Líder del equipo Scrum cuyo objetivo es ayudar al equipo a entender y aplicar Scrum correctamente, y asegurar que el equipo tenga un ambiente de trabajo adecuado para completar el trabajo.
2. **Product Owner.** Se encarga de velar y representar los intereses de los clientes.
3. **Equipo.** Se encargan de llevar a cabo las tareas encomendadas, poseen los conocimientos técnicos necesarios para llevar sus responsabilidades.

El rol de Scrum Master y Product Owner fueron llevados a cabo por el profesor Henrique Chevreux y la profesora copatrocinante Valeria Henríquez respectivamente, quiénes se encargaron de verificar que las funcionalidades desarrolladas cumplan con las exigencias de los usuarios, además se encargaron de comunicar las dudas del equipo con los clientes y de transmitir las necesidades de los clientes con el equipo. El equipo de trabajo encargado del módulo de notificaciones y la mesa de ayuda estuvo compuesto por Jonathan Gonzalez, autor de este documento.

Para facilitar la gestión de los sprint y la comunicación del equipo se utilizó la plataforma Trello, en esta se creó un tablero en donde se definieron cinco columnas por las cuales se movían las tarjetas. Cada tarjeta correspondía a una historia de usuario.

1. Backlog. Corresponden a todas las historias de usuarios.
2. Sprint Backlog. Historias de usuario definidas para completar en el sprint.
3. In Progres. Tareas en las cuales el equipo se encuentra trabajando.
4. PO Review. Tareas que el equipo considera cómo completadas, pero que debe ser revisadas por el Product Owner, si este último considera que no se ha completado, la tarjeta es devuelta a la columna “In Progress” con los comentarios pertinentes.
5. Done. Tareas completadas y aceptadas por el Product Owner.

En la Figura 17 se puede ver una imagen del tablero utilizado.

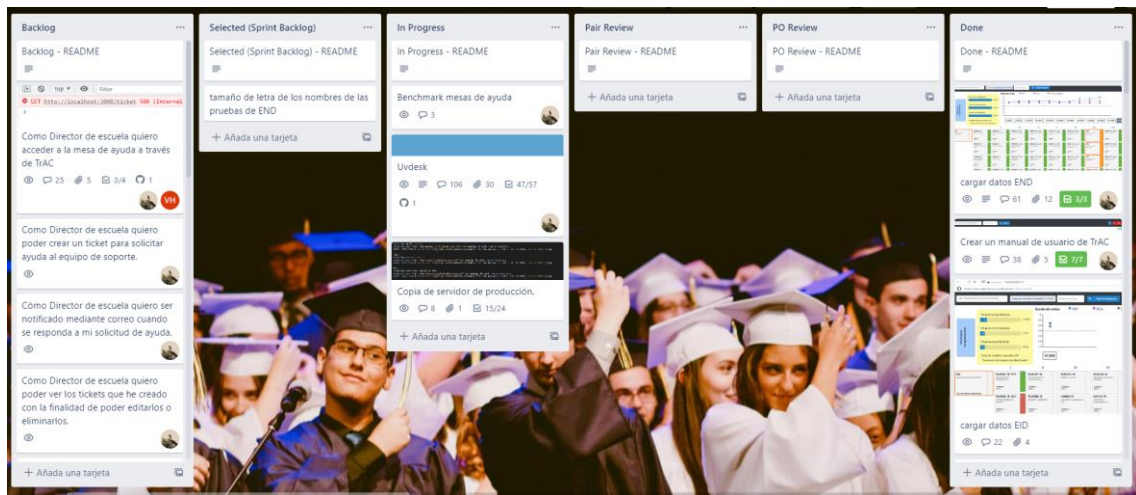


Figura 17: Organización tablero Trello desarrollo de mejoras TrAC-FID

Se definió la duración de cada Sprint en dos semanas y la Sprint Planning Meeting se llevó a cabo los jueves, en estas reuniones se revisaba el avance propuesto en la definición del sprint backlog con el avance real. Cada reunión se respondía a tres interrogantes ¿En qué estoy trabajando? ¿Qué problemas he tenido? Y ¿Qué planeo hacer?, además se transmitían las dudas en caso de existir y se volvía a definir el sprint Backlog. Todas las reuniones fueron realizadas de manera remota.

El plan de trabajo se dividió en tres etapas principales:

1. Septiembre a diciembre, 2021: En este periodo se desarrollaron los objetivos específicos uno, dos y tres. Se realizó un análisis de las tecnologías y el funcionamiento de las componentes y su lógica a reutilizar, para finalizar con el desarrollo del módulo de notificaciones.
2. Septiembre a noviembre, 2022: Implementación mesa de ayuda. Se completaron los objetivos específicos cuatro y cinco. Se definieron los requisitos y la tabla comparativa de plataformas de mesa de ayuda *open-source* permitiendo la selección de la que fue implementada en este mismo periodo.
3. Diciembre, 2022: Validación. Se llevaron a cabo las pruebas de usabilidad para satisfacer el objetivo específico seis.

Para el control de versiones y alojamiento de código se utilizó GitHub². En esta plataforma se encuentra el repositorio principal de TrAC-FID a cargo de la empresa Puentes Digitales. Mediante un *fork* se creó una nueva rama, ver Figura 18.

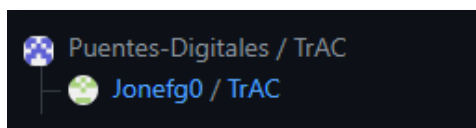


Figura 18: Ramas GitHub TrAC-FID

GitHub permite actualizar el código de una aplicación de manera colaborativa, es decir que muchos usuarios pueden contribuir a su desarrollo, mediante un *fork*³ se crea una copia idéntica del repositorio esto permite que los cambios realizados al código y que son subidos a la plataforma se almacenen en la copia y no afecten al repositorio principal. Una vez los cambios agregados a la copia del repositorio principal eran aprobados por el equipo, se realizaba una solicitud de *Pull request*, en donde se solicitaba pasar los cambios desde el repositorio copia a él repositorio principal, esta solicitud era aceptada por el encargado de este último. Los *Pull request* se realizaban al pasar una tarjeta de la columna PO Review a Done en el tablero de Trello.

² GitHub es una plataforma de alojamiento, propiedad de Microsoft, que ofrece a los desarrolladores la posibilidad de crear repositorios de código y guardarlos en la nube de forma segura, usando un sistema de control de versiones llamado Git.

³ Una bifurcación (*fork* en inglés), en el ámbito del desarrollo de software, es la creación de un proyecto en una dirección distinta de la principal u oficial tomando el código fuente del proyecto ya existente. Comúnmente se utiliza el término inglés. Como resultado de la bifurcación se pueden llegar a generar proyectos diferentes que cubren necesidades distintas, aunque similares.

4.2 Notificaciones

La siguiente sección se encuentra estructurada en base a la secuencia de pasos seguidas durante todo el proceso de desarrollo, el cual se dividió en los dos grandes componentes que se construyeron en este proyecto.

Primero el módulo de notificaciones, se identificaron las componentes existentes en la plataforma que podrían ser reutilizadas. Esto incluyó la revisión de las funciones existentes y la evaluación de cómo podrían ser utilizadas para cumplir con los requerimientos específicos del módulo. Una vez identificadas se procedió a la siguiente etapa en donde se crearon los diseños para el módulo de notificaciones agregando las lógicas necesarias para cumplir con los requerimientos específicos. Se trabajó en el diseño de la interfaz de usuario y la estructura del contenido de las notificaciones. Finalmente, se procedió a la implementación del módulo de notificaciones. En esta etapa, se desarrolló la nueva pestaña de notificaciones, se estableció la estructura del contenido de estas y de los correos de notificación.

Una vez completado el desarrollo del módulo de notificaciones, se procedió a la implementación de la mesa de ayuda. Esta etapa inició con un análisis general de las tecnologías utilizadas para el funcionamiento de UVdesk, luego se implementó la mesa de ayuda en un servidor propio y se analizaron y desarrollaron las funcionalidades necesarias para la correcta comunicación entre TrAC-FID y UVdesk.

4.2.1 Análisis de tecnologías y lógica de TrAC-FID.

El análisis de tecnologías de TrAC-FID fue presentado en el capítulo dos de este documento, en donde se analizó la situación actual de la plataforma. Para conocer las tecnologías se utilizó la documentación de TrAC, en donde se describen las principales de la plataforma tal cómo se muestra en la Figura 19, extraída del documento.

Software specification

- Programming Language: [TypeScript](#)
- Frameworks: [Next.js](#) [React.js](#) [Fastify](#) [TypeGraphQL](#) [Mercurius](#)
- Environment: [Node.js](#)
- Database: [PostgreSQL](#)
- Database Query Builder: [Knex.js](#)
- UI Libraries: [Chakra UI](#) [Semantic UI](#) [React](#)
- Mail service: [SendGrid: Email Delivery Service](#)
- Testing framework: [Jest](#)

Figura 19: Tecnologías TrAC - TrAC v2 *Documentation*

Además, en esta etapa fue necesario realizar las instalaciones necesarias para preparar el entorno de desarrollo en la máquina del autor, se realizaron instalaciones asociadas a la base de datos como PostgreSQL y pgAdmin, el primero mencionado en el capítulo uno y el segundo corresponde a una interfaz gráfica que permite una visualización amigable de la base de datos. Además, para la ejecución de la aplicación se necesitó instalar nodeJS y *pnpm*, el entorno de ejecución de JavaScript y un gestor de paquetes respectivamente. Las librerías necesarias fueron instaladas de manera automática con *pnpm*.

Luego de preparado el entorno para ejecutar la aplicación de TrAC-FID de manera local. Se procedió a un análisis del código y la estructura de las carpetas.

La organización de las carpetas y los archivos de código corresponden al estándar de organización de un proyecto NodeJS, una carpeta correspondiente a la API otra al Cliente. Dentro de la carpeta de la API se encuentran otras representadas con los nombres de los bloques descritos en la Figura 3, cómo se puede ver en la Figura 20.

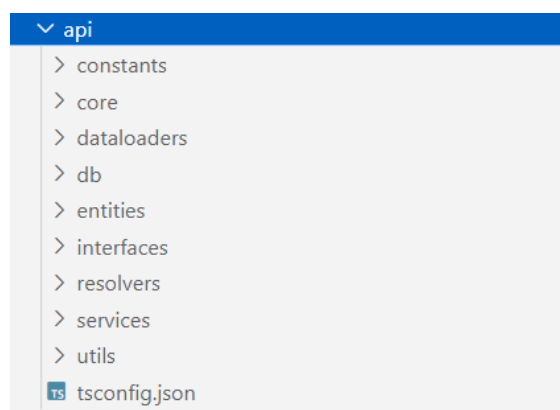


Figura 20: Carpeta api proyecto TrAC-FID

En la carpeta Client, se encuentra el código de los elementos desplegados en la interfaz gráfica, organizados por carpetas representativas de cada componente, siguiendo con la lógica de organización de React, ver Figura 21.

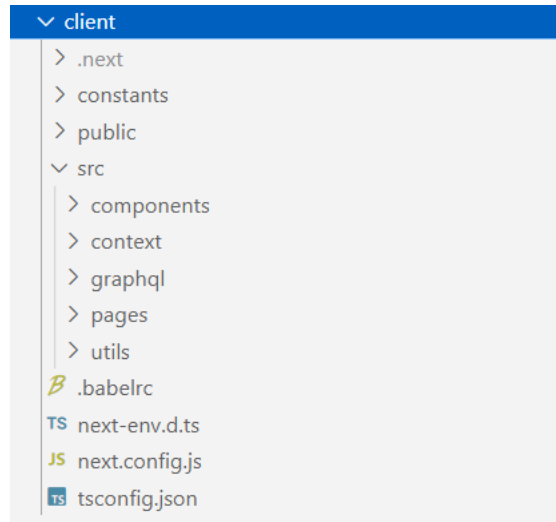


Figura 21: Carpeta *client* proyecto TrAC-FID.

Luego de que el entorno de desarrollo local se encontró en funcionamiento, haciendo uso de ingeniería inversa se analizó el funcionamiento de las distintas componentes que podrían ser reutilizadas para el desarrollo del módulo de notificaciones. A continuación, se presenta un breve resumen del análisis realizado y las conclusiones que se fueron obteniendo de cada una.

4.2.1.1 Envío de correos de activación de cuenta

TrAC-FID posee un sistema de creación de usuarios, el cual se encarga de enviar un correo de activación de cuenta a los usuarios creados. Para el envío de los correos se utiliza el servicio de envío de correos automáticos ⁴*SendGrid*. El funcionamiento del envío de correos de activación a todos los usuarios bloqueados se describe en la Figura 22.

⁴ SendGrid es un proveedor SMTP basado en la nube que le permite enviar correos electrónicos sin tener que mantener servidores de correo electrónico. SendGrid gestiona todos los detalles técnicos, desde la ampliación de la infraestructura hasta el alcance del ISP y el control de la reputación hasta la lista blanca servicios y análisis en tiempo real.

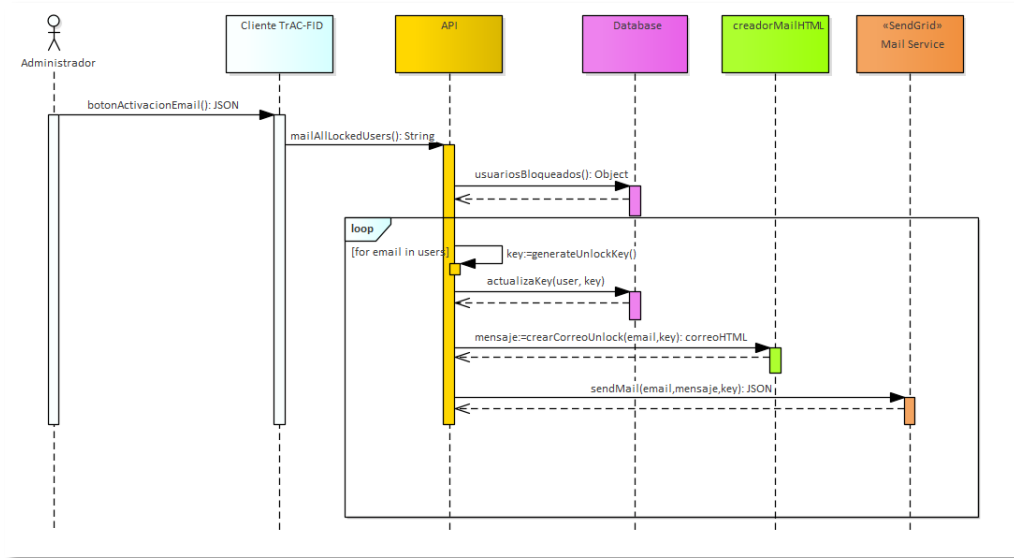


Figura 22: Diagrama de secuencia de envío de correos de activación a todos los usuarios bloqueados.

Esta funcionalidad permitió identificar los componentes y lógicas que están involucradas en el envío de correos electrónico y cuál es la labor que realiza cada uno. Los puntos más relevantes identificados gracias a este análisis fueron los siguientes:

- El contenido de cada correo electrónico es creado por una componente que a partir de textos predefinidos crea el mensaje y lo retorna cómo un objeto de tipo HTML (Lenguaje de Marcas de hipertexto), representado por la componente “creadorMailHTML” en el diagrama de la Figura 20.
- Luego de creado el mensaje este es enviado a una función que envía el correo electrónico haciendo uso de SendGrid, donde además del mensaje debe agregarse el email del destinatario y del emisor, junto a otras variables cómo el asunto.

Para el desarrollo del envío de notificaciones se reutilizaron los archivos encargados de crear los mensajes contenidos en los correos electrónicos, en donde se agregaron nuevas funciones encargadas de crear los mensajes de alerta. Se replicó la lógica de envío de correos mediante SendGrid. Además, se replicó la manera en que se comunican las funciones involucradas en todo el proceso.

4.2.1.2 Fechas y tipos de carga

La base de datos de TrAC-FID es alimentada mediante un proceso ETL que transforma determinados archivos y almacena los datos en las tablas correspondientes. Estos archivos

proviene de diferentes fuentes y son obtenidos y cargados en determinadas fechas en el año, tal cómo se detalla en la Tabla 9.

Tabla 9: Tipos de archivos y fechas aproximadas de carga

Nombre archivo(s)	Fecha aproximada de carga
mallas_curriculares.xlsx	Inicio año académico: Marzo
asignaturas_inscritas.xlsx, asignaturas_cursadas.xlsx, situación_estudiantes.xlsx, registro_admision.xlsx	Inicio de cada semestre: Marzo y Agosto
indicadoresFID_AAAAMMDD.xlsx	Al término de la matrícula: Abril aprox.
empleabilidad_AAAAMMDD.xlsx	Al recibir los resultados: Noviembre
EID_AAAAMMDD.xlsx	Al recibir los resultados: Mayo
END_AAAAMMDD.xlsx	Al recibir los resultados: Marzo

Cada vez que se realiza este proceso, se guarda en la Tabla *Parameters* de la base de datos las fechas de carga y el tipo. En la Figura 23, se explica con mayor detalle las columnas de esta Tabla.

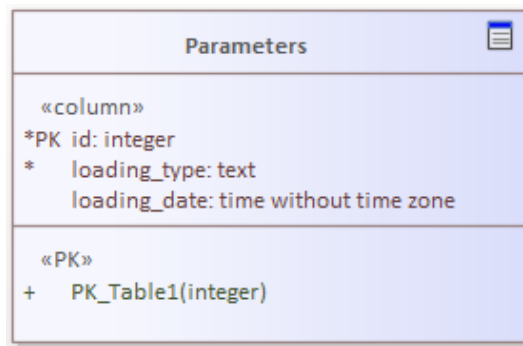


Figura 23: Estructura Tabla Parameters

El campo *id* corresponde al indicador único del registro, el valor de *loading_type* y *loading_date* corresponden a el tipo o nombre del archivo de carga y la fecha en que fue procesado por el ETL.

Esta información se muestra en un pequeño componente visual en la interfaz gráfica de la plataforma, ver Figura 24. Permite conocer el estado de actualización de los componentes mostrados en la interfaz gráfica, en otras palabras, se puede conocer la diferencia entre la fecha en que se visualizan los datos y en la que fueron cargados.



Figura 24: Componente Parameters Interfaz TrAC-FID.

El análisis a esta funcionalidad permitió:

- Conocer cuál es la tabla de la base de datos que contiene la información.
- A nivel de API, permitió identificar las funciones asociadas a la lectura de la tabla, y cómo éstas se comunican para analizar y transmitir la información al Cliente Web.

Se reutilizaron las funciones encargadas de obtener la información desde la base de datos en la API, además se replicaron algunas funciones que se utilizan para comunicar esta información al Cliente Web.

4.2.1.3 Situaciones de riesgo

Al igual que con la tabla *Parameters* las situaciones de riesgo son calculadas al procesar los archivos mencionados en la Tabla 9: Tipos de archivos y fechas aproximadas de carga. El proceso ETL realizado para alimentar la base de datos contempla dentro de sus tareas el análisis y cálculo de los estudiantes que se encuentran en situaciones de riesgo y son guardados cómo filas en la tabla *Risk_notification* que se puede ver en la Figura 25.

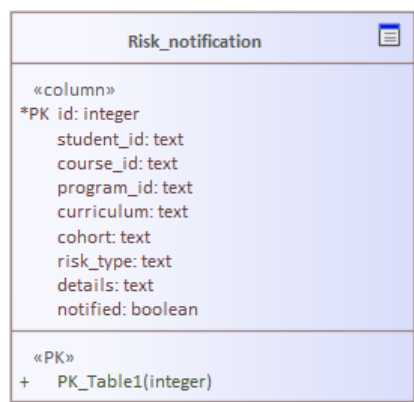


Figura 25: Estructura Tabla Risk_notification.

El campo id corresponde al identificador único del registro, *student_id* corresponde al identificador del alumno, *course_id*, *program_id*, *curriculum* y *cohort* corresponden a información relacionada con la carrera, año del plan que cursa y año de ingreso.

Esta información es presentada por un panel lateral flotante en la interfaz gráfica de TrAC-FID, cómo se muestra en la Figura 26. Ofrece a los usuarios la posibilidad de conocer y analizar directamente la información de estudiantes que requieren de mayor atención dada su situación.

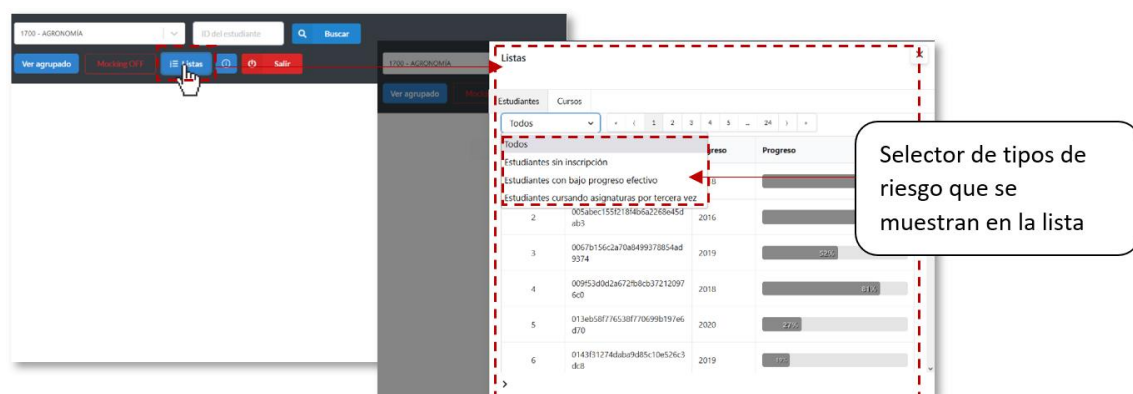


Figura 26: Panel lateral Listas Interfaz TrAC-FID.

Las conclusiones obtenidas mediante este análisis son similares a las que se obtuvieron a partir de la funcionalidad anterior, esto es, se reutilizó la tabla involucrada y que guarda la información, así como la lógica asociada a las funcionalidades a nivel de API.

4.2.2 Desarrollo

Una vez concluidos los análisis anteriormente mencionados, se determinó que se habían cumplido los dos primeros objetivos específicos y se procedió con el desarrollo del módulo de notificaciones.

En línea con el enfoque de desarrollo ágil, el módulo de notificaciones se desarrolló de manera iterativa y con requisitos flexibles. Esto significa que hubo cambios que tuvieron que ser revertidos y otros requisitos adicionales que se agregaron a medida que avanzaba el proceso, con el objetivo de brindar una solución que se ajustara lo más posible a las necesidades de los usuarios. Teniendo esto en consideración a continuación se presentan los resultados del desarrollo realizado.

4.2.2.1 Pestaña Notificaciones en panel de administrador

Como se mencionó en el Diseño de interfaz, se agregó un panel de notificaciones en la pestaña de la ventana de administrador de TrAC-FID, ver Figura 27. Se utilizó la misma biblioteca de iconos para obtener un icono de "correo" o "mensaje" que mantuviera el mismo formato que en las demás pestañas. El nombre de la pestaña es "Notifications", cumpliendo con el requisito de que las nuevas palabras agregadas deben estar en idioma inglés. El botón "Check Notifications" tiene el mismo formato visual que los demás botones de las pestañas contiguas a la de notificaciones, centrado y con bordes redondeados. En la tabla, se decidió separar el contenido de la notificación en los textos del mensaje (encabezado, saludo, pie de página, etc.), los tipos y fechas de cargas y los riesgos notificados, además se agregó un contador de veces que se ha enviado el mensaje mediante la funcionalidad de reenvío. Se eligió seguir esta estructura ya que es similar a la que describe la tabla de la base de datos.

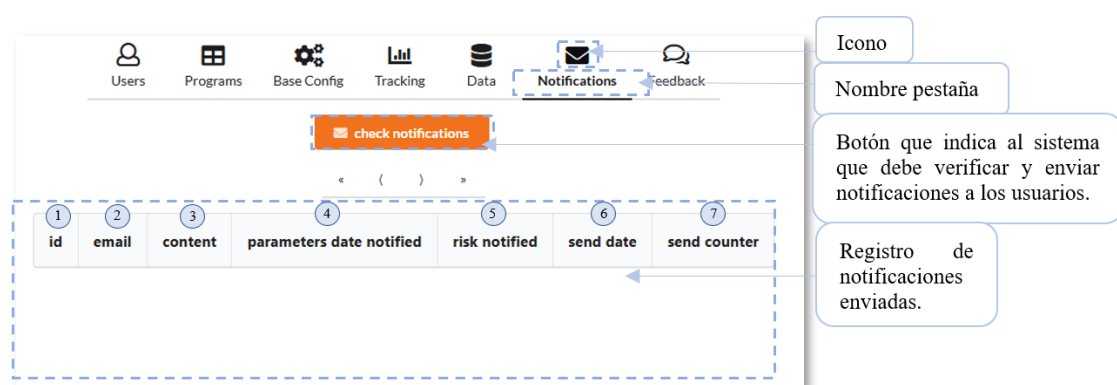


Figura 27: Pestaña *Notifications*, panel de administrador

Cada fila corresponde a un correo electrónico enviado y las columnas de la tabla se detallan a continuación:

1. **Id:** Número identificador del mail.
2. **Email:** Correo electrónico del usuario.
3. **Content:** Contenido del mail enviado, este elemento se detalla más adelante en el documento.
4. **Parameters date notified:** Indica las fechas y tipos de cargas que se notificaron al usuario.
5. **Risk notified:** Indica los tipos y cantidad de riesgos notificados.
6. **Send date:** Indica la fecha en que se envió el correo
7. **Send counter:** Contador de veces que se ha enviado la misma notificación

4.2.2.2 Correo de notificación

Para el diseño del mail se siguió una estructura bastante simple y teniendo en cuenta las consideraciones descritas en el capítulo dos referentes a el formato de los correos de notificación. La Figura 28 sirve de ejemplo del cómo se ven los emails enviados por el sistema de notificaciones, se cumple con todos los requisitos descritos durante la etapa de diseño y además se generó un archivo GIF que muestra a los usuarios cómo acceder al panel de situaciones de riesgos.

Este GIF puede ser visitado en el siguiente [enlace](#).

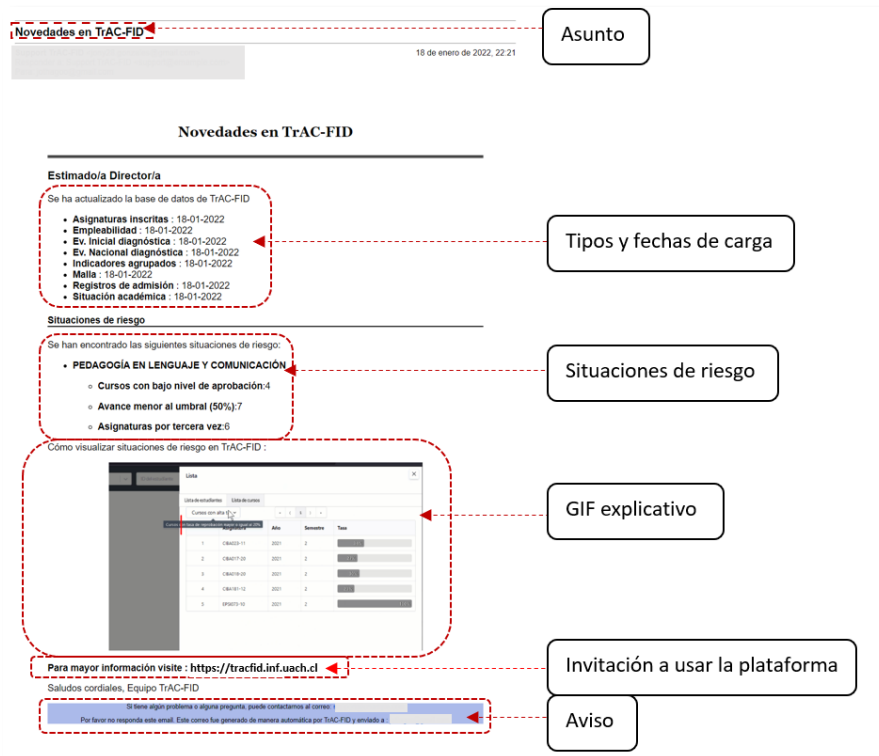


Figura 28: Formato correo electrónico de notificación TrAC-FID

4.2.2.3 Diagrama de secuencia

El diagrama de secuencia de la Figura 29 ilustra la comunicación implementada entre los distintos componentes de TrAC-FID, y cómo éstos trabajan juntos para llevar a cabo el proceso del envío de alertas mediante notificaciones. El proceso se inicia cuando el usuario presiona el botón "verificar notificaciones", lo que desencadena una serie de interacciones entre la API y los demás componentes del sistema.

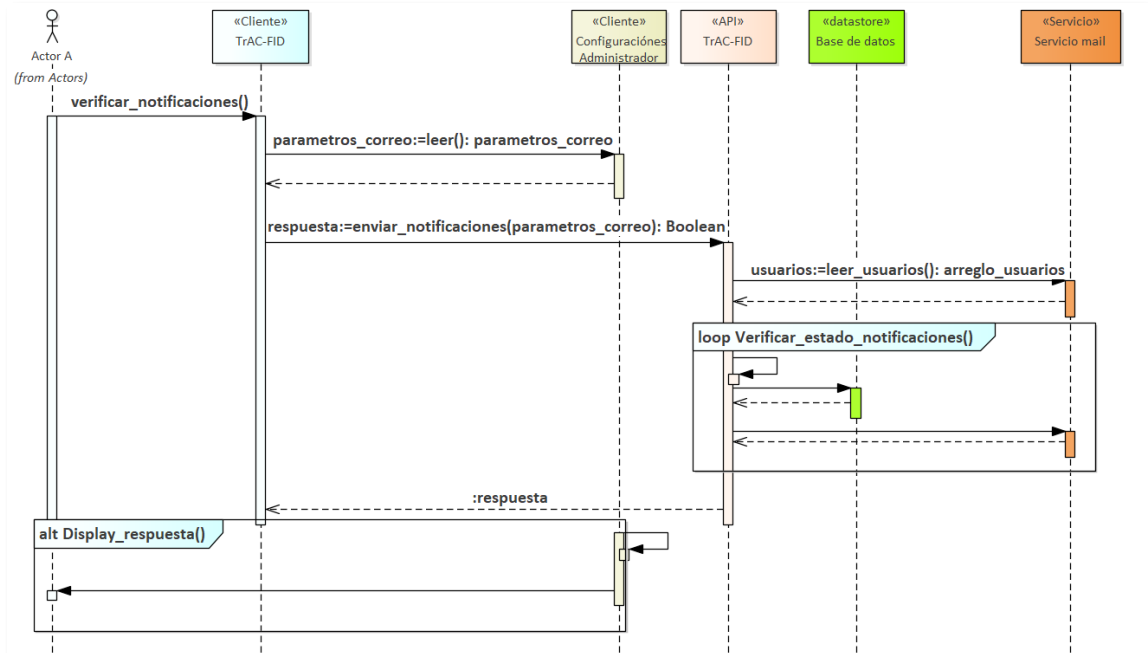


Figura 29: Diagrama de secuencia del envío automático de notificaciones.

Luego de que el usuario realiza la llamada a la función “verificar_notificaciones” mediante un clic sobre el botón correspondiente, el Cliente se comunica con la API para indicar que el usuario desea analizar el estado de las notificaciones, el mensaje posee en sus parámetros los textos asociados a la creación de la estructura del correo electrónico, cómo el asunto, el saludo de inicio, la despedida, etc. Que se encuentran en la configuración de administrador del Cliente y a los cuales la API no puede acceder directamente.

Una vez la API recibe el mensaje, prepara la respuesta. Comienza con un análisis sobre que notificaciones debe enviarse y los usuarios que deben ser notificados, este proceso se puede ver con mayor detalle en la Figura 30.

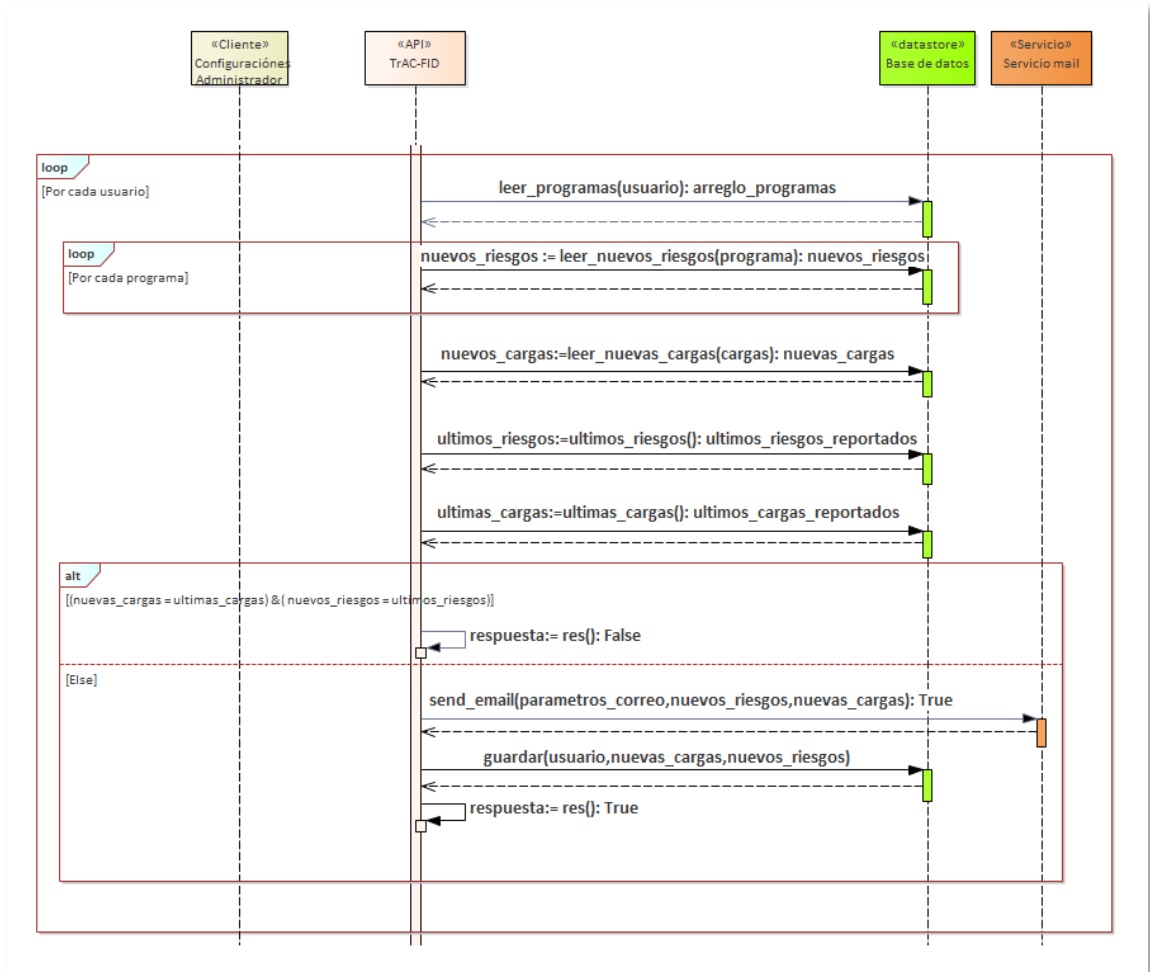


Figura 30: Diagrama de secuencia Verificar_estado_notificaciones().

En esta sección de la secuencia de mensajes, la API comienza a recorrer un arreglo o lista de todos los usuarios directores que se encuentran en el sistema, por cada usuario del arreglo lee los programas asociados a él, genera un arreglo de notificaciones que deben enviarse asociadas a ese programa, y genera un correo donde resume la información. En otras palabras, le envía a cada usuario las notificaciones asociadas a los programas o carreras que tengan registrados. Para verificar que no se envíen las mismas notificaciones una y otra vez, se decidió crear una tabla de registros de mensajes enviados, entonces podrá compararse el resultado del análisis hecho en el instante que se presionó el botón con el último registro de la base de datos, si estos son iguales significa que no debe volver a enviarse ese correo electrónico de notificación, en caso contrario (distintos) se envía el correo de notificación.

La tabla *Notification Data* se encargará de registrar todas las notificaciones enviadas a los usuarios. Su estructura es la que se muestra en la Figura 31.

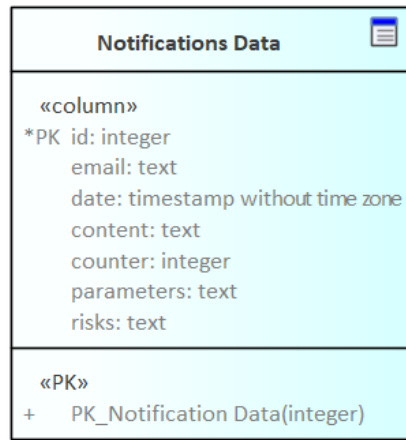


Figura 31: Estructura Tabla Notifications Data

El campo *id* corresponde al identificador único del registro, *email* corresponde al correo electrónico del usuario notificado, *date* almacena la fecha de envío, *content* corresponde a las partes del mensaje, *parameters* son las fechas y tipos de cargas, *risks* son los riesgos notificados y *counter* guarda el número de veces que se ha enviado el mismo correo.

Si resulta que no existen nuevas notificaciones que enviar, la API responde al Cliente quién se encarga de comunicar el resultado al usuario mediante un mensaje en la pantalla, en el caso que se hayan enviado notificaciones el Cliente además del mensaje indicando que las notificaciones han sido enviadas exitosamente, actualizará los registros que se muestran en la Tabla de la pestaña *Notifications* (Ver Figura 27). Este proceso puede verse de manera representado gráficamente en el Diagrama de secuencia de la Figura 32.

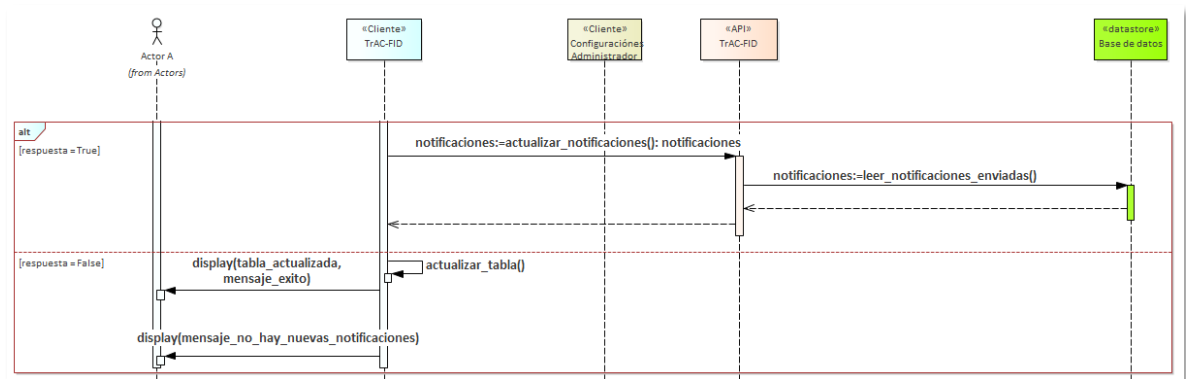


Figura 32: Diagrama de secuencia para analizar respuesta.

El Cliente indica a la API que necesita actualizar la tabla de registros, esta los lee de la tabla *NotificationsData* y los agrega a la respuesta al Cliente quién se encarga de actualizar los registros que se muestran en la tabla de la interfaz gráfica.

4.3 Mesa de ayuda

En esta sección se detallan los trabajos realizados para cumplir con los requisitos y diseños definidos para la mesa de ayuda y que se especificaron en el capítulo anterior. Se comenzó con la elaboración de manuales de usuario que explicaban las funcionalidades de TrAC-FID y las tareas relacionadas con su operación. Luego, se inició la implementación de la mesa de ayuda UVdesk, la cual planteó importantes desafíos técnicos que fueron superándose a medida que se avanzaba en su desarrollo.

Se describen inicialmente los manuales elaborados y se divide el proceso de implementación en seis etapas clave:

- Identificación de tecnologías.
- Implementación de la mesa de ayuda con un servidor web independiente.
- Configuraciones necesarias para la correcta integración de la mesa de ayuda a TrAC-FID.
- Persistencia de las configuraciones y la información.
- Configuración de dominio y servidor web.
- Comunicación con TrAC-FID.

4.3.1 Manuales

Uno de los requisitos de la mesa de ayuda era la creación de manuales para los distintos usuarios de TrAC-FID, estos fueron desarrollados y creados como documentos, en paralelo a la etapa de análisis y búsqueda de potenciales soluciones de mesa de ayuda.

A continuación, se da una breve reseña de los manuales desarrollados.

- **Manual de usuario**

Manual desarrollado para los usuarios de la plataforma TrAC-FID, en este se describen y especifican todas las funcionalidades que ofrece la plataforma desde la perspectiva de los usuarios del sistema. Además, ofrece información con respecto a procesos de gestión y creación de una cuenta.

- **Manual de operación de TrAC-FID**

Manual orientado a operadores o administradores de la plataforma y fue desarrollado para explicar funcionalidades relacionadas con la configuración de TrAC-FID y la administración de los usuarios del sistema. El documento detalla los pasos a seguir para

crear y gestionar usuarios, así como definir sus privilegios, además de configuraciones de texto y funcionalidades en desarrollo, todas accesibles desde el panel de control de administrador.

4.3.2 Identificación de tecnologías

Las funcionalidades ofrecidas por la mesa de ayuda UVdesk calzaban completamente con los requisitos previamente definidos. Pero fue necesario analizar con que tecnologías estaba construido el sistema para poder medir el nivel de desafíos técnicos que presentaría su puesta en marcha. La comunidad de UVdesk ofrece un repositorio alojado en GitHub y al cual se puede acceder de manera gratuita.

GitHub muestra en la interfaz del proyecto las principales tecnologías con las cuales está construido, ver Figura 33.

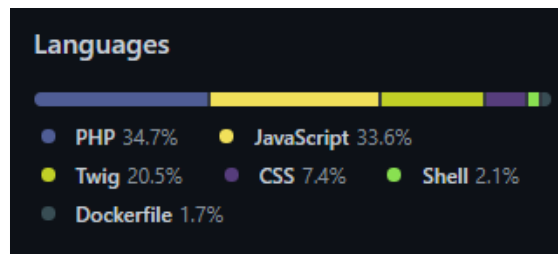


Figura 33: Principales tecnologías del proyecto UVdesk

UVdesk es un sistema de mesa de ayuda desarrollado principalmente en PHP⁵, que utiliza librerías adicionales como Twig⁶ para facilitar la implementación de lógicas e interfaces. También utiliza tecnologías web estándar como HTML, CSS y JavaScript. La instalación se realiza a través de Composer⁷, un framework de instalación de paquetes que se encarga de instalar todas las librerías necesarias para el funcionamiento y la puesta en marcha del sistema. Para el envío automático de notificaciones hace uso de SwiftMailer, un servicio de transporte de correos electrónicos para aplicaciones desarrolladas con PHP (Soni, 2022).

Para conocer más detalles sobre el proyecto UVdesk puede visitar el siguiente [enlace](#).

⁵ PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

⁶ Twig es un motor de plantillas para el lenguaje de programación PHP. Su sintaxis origina de Jinja y las plantillas Django.

⁷ Composer es un sistema de gestión de paquetes para programar en PHP el cual provee los formatos estándar necesarios para manejar dependencias y librerías de PHP.

4.3.3 Implementación de mesa de ayuda UVdesk como servicio web

Para poner en funcionamiento la mesa de ayuda, se utilizó un servidor distinto al que aloja el proyecto de TrAC-FID, esto con el fin de evitar posibles vulneraciones de seguridad y/o problemas de estabilidad y fiabilidad como se mencionó en la sección Diseño de software de la mesa de ayuda. Se hizo una copia del proyecto en el nuevo servidor y se comenzó con la puesta en operación. Para hacer funcionar la mesa de ayuda como servicio se utilizaron contenedores Docker. Se creó un contenedor para cada pieza de software de la mesa de ayuda, la aplicación principal y la base de datos. Mediante el archivo *Dockerfile*⁸ se realizó la configuración automática de la base de datos durante el proceso de creación del contenedor. La conexión con la mesa de ayuda fue realizada mediante la interfaz de instalación de UVdesk. La arquitectura del sistema hasta ese momento puede verse en la Figura 34.

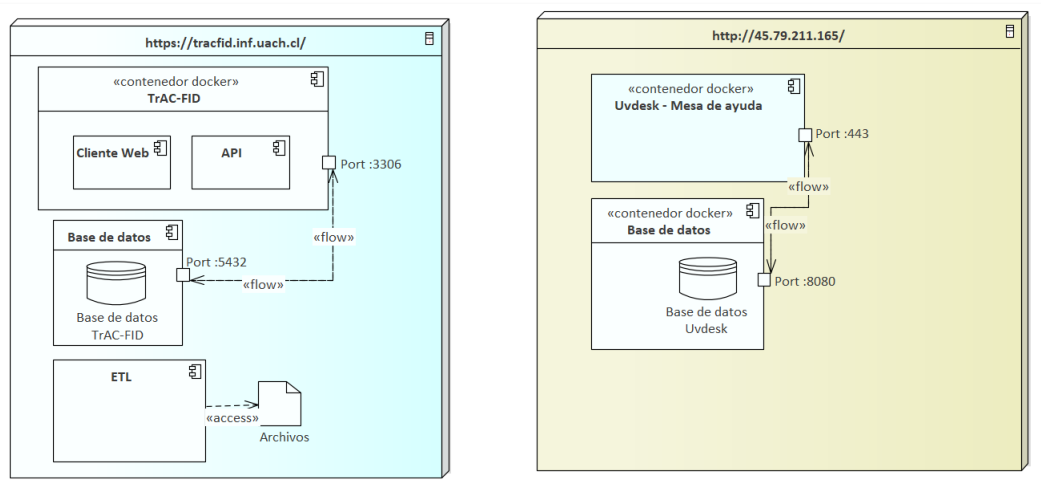


Figura 34: Arquitectura inicial mesa de ayuda

Los sistemas aún no se comunicaban y el dominio era accedido a través de la IP⁹ del servidor host, pero ya era posible interactuar con la mesa de ayuda y realizar las configuraciones necesarias.

⁸ Dockerfile es un archivo de texto que contiene las instrucciones necesarias para crear una nueva imagen del contenedor. Estas instrucciones incluyen la identificación de una imagen existente que se usará como base, los comandos que se ejecutarán durante el proceso de creación de la imagen y un comando que se ejecutará cuando se implementen instancias nuevas de la imagen del contenedor.

⁹ La dirección IP es una etiqueta numérica que identifica, de manera lógica y jerárquica, a una interfaz en la red de un dispositivo que utilice el Protocolo de Internet

Luego de que el servicio de la mesa de ayuda se encontraba en funcionamiento se procedió a la configuración del sistema. Los siguientes cambios no representan necesariamente el orden cronológico seguido.

4.3.4 Adecuaciones

A continuación, se presentan los cambios aplicados directamente en el código de los archivos de la plataforma.

4.3.4.1 Idioma

El proyecto UVdesk posee una carpeta en la cual se encuentran las traducciones de todas las palabras que se muestra en la interfaz gráfica, dentro de estas se encontraba el archivo correspondiente a la traducción al idioma español. Se corrigieron las palabras que no estaban correctamente traducidas.

4.3.4.2 Aspecto visual

Se creo un acceso a los manuales de usuarios a partir de la interfaz principal de los Directores. El pie de página indicando la empresa encargada de poner en marcha la mesa de ayuda, Puentes Digitales. Ambos cambios se pueden ver en la Figura 35.

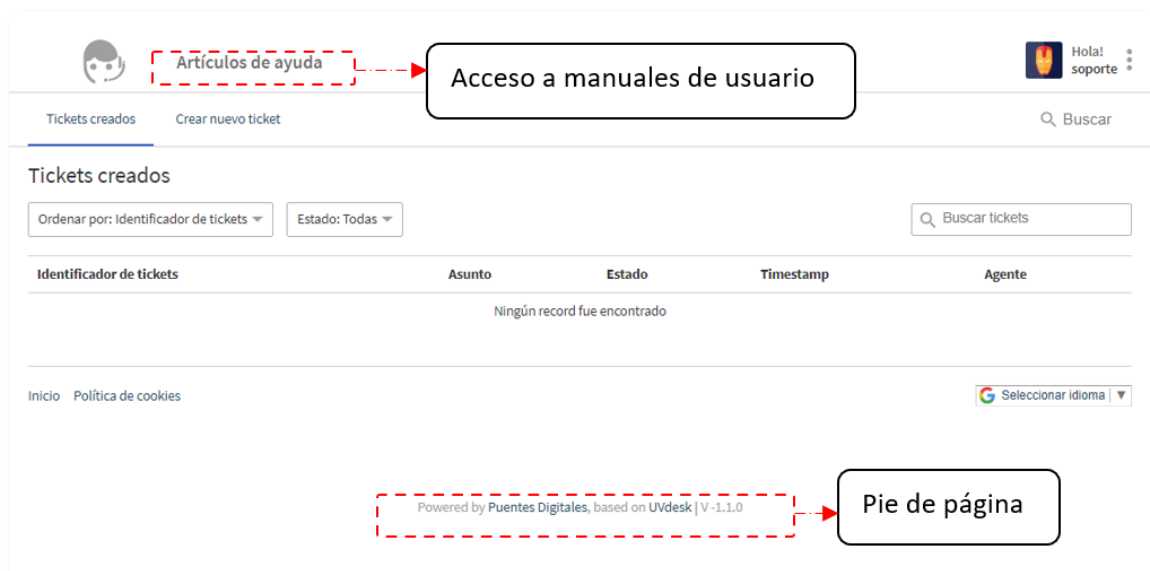


Figura 35: Cambios visuales a nivel de código en mesa de ayuda

Se ocultó la barra de desarrollador, ver Figura 36 y Figura 37.

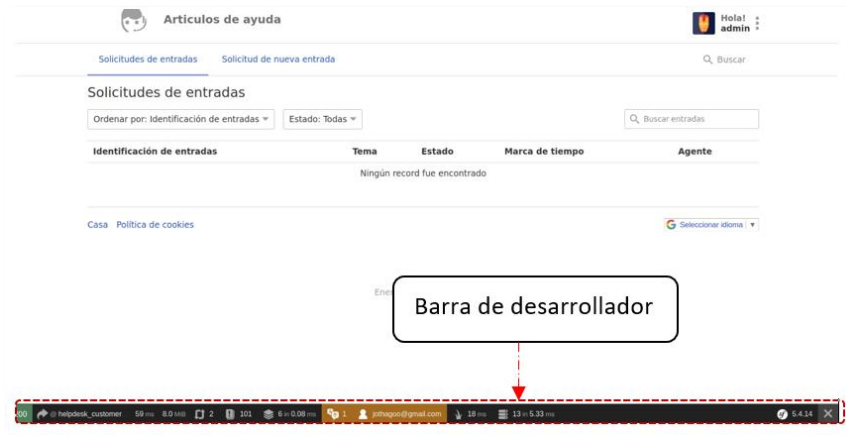


Figura 36: Interfaz gráfica UVdesk con barra de desarrollador



Figura 37: Interfaz gráfica UVdesk sin barra de desarrollador

4.3.5 Configuración de la mesa de ayuda

Las siguientes configuraciones fueron realizadas utilizando la interfaz gráfica de la mesa de ayuda.

Se definieron los colores principales de la mesa de ayuda siguiendo la paleta asociada a TrAC-FID, ver Figura 38.

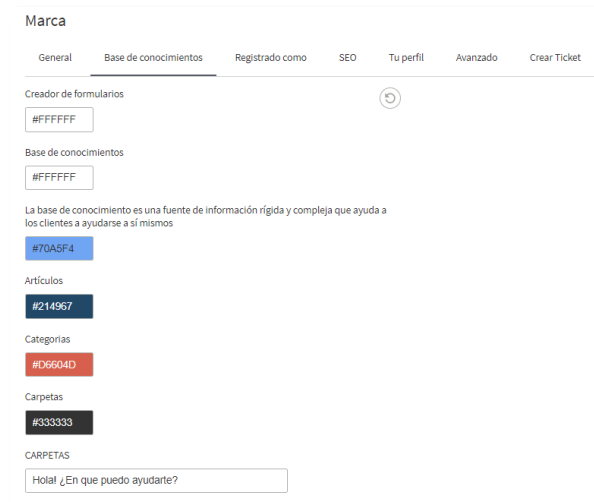


Figura 38: Paleta de color aplicada a mesa de ayuda UVdesk

4.3.5.1 Correo electrónico

Se realizó además la configuración de SwiftMailer¹⁰ para conectar la de ayuda con el correo electrónico de soporte y acceder las funcionalidades de envío de mails automáticos. Primero se creó un buzón que corresponde a la conexión entre la mesa de ayuda y el buzón de correos electrónicos de la cuenta de soporte en Gmail¹¹, ver Figura 39.

¹⁰ Swift Mailer es una biblioteca basada en componentes para enviar correos electrónicos desde aplicaciones PHP.

¹¹ Gmail es un servicio de correo electrónico proporcionado por la empresa estadounidense Google.

Figura 39: Configuración buzón UVdesk

Junto con la creación del buzón fue necesario implementar un servicio Cron¹² mediante la terminal de comandos del sistema contenedor de la mesa de ayuda. A este se le encargó la tarea de ejecutar un comando para leer el buzón de Gmail cada tres minutos, en el caso de que existan correos nuevos el sistema creará automáticamente un ticket con cada correo, el comando ejecutado se puede ver en la Figura 40.

```
*/3 * * * * php bin/console uvdesk:refresh-mailbox suportetracfid@uvdesk.puentesdigitales.cl
```

Figura 40: Comando Cron para la lectura de correos cada tres minutos.

Además, se realizó la configuración necesaria para que el sistema envíe correos a nombre del equipo de soporte. Existen varios procesos que dan inicio al envío de un correo, estos son detallados más adelante.

¹² Cron es un demonio de programación de tareas basado en el tiempo que se encuentra en sistemas operativos similares a Unix, incluso en distribuciones de Linux. Se ejecuta en segundo plano y las tareas programadas con él, llamadas “tareas de Cron”, se ejecutan de forma automática, lo cual hace que este demonio sea útil para automatizar tareas relacionadas con el mantenimiento.

4.3.5.2 Base de conocimiento

Durante el transcurso de la implementación se desarrollaron los manuales de usuario relacionados con la utilización del sistema TrAC-FID. En estos se describen y especifican todas las funcionalidades que ofrece la plataforma. Estos manuales fueron agregados al apartado de base de conocimiento de la mesa de ayuda, tal cómo se puede visualizar en la Figura 41.



Figura 41: Artículos de ayuda UVdesk TrAC-FID.

El Manual de usuario dio origen a 38 artículos dentro de la base de conocimiento de UVdesk. Cada artículo posee imágenes explicativas con los pasos que se van describiendo, además se utilizó la capacidad de agregar vínculos entre diferentes artículos para ofrecer al usuario la posibilidad de investigar funcionalidades relacionadas. Cada artículo posee su propia página dentro de la plataforma tal cómo se puede ver el ejemplo de la Figura 42.

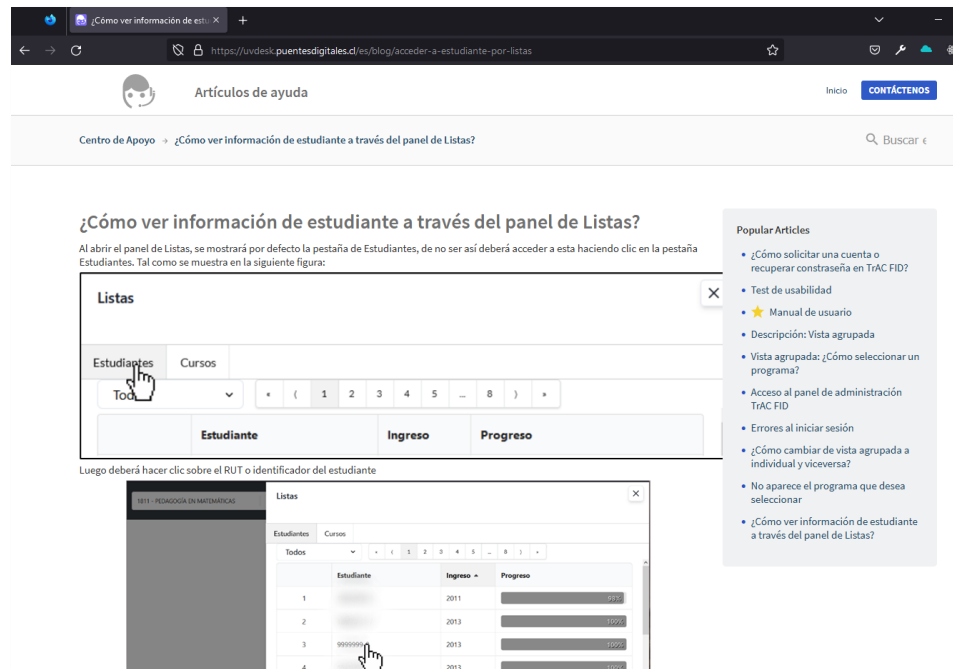


Figura 42: Ejemplo de artículo - ¿Cómo ver información de estudiantes a través del panel de Listas?

4.3.5.3 Flujos de trabajo

La automatización de tareas es una de las funcionalidades a las cuales se les sacó provecho durante la etapa de habilitación de la mesa de ayuda. Una de estas funcionalidades corresponde a los flujos de trabajos, tareas automatizadas que se realizan cuando ocurren ciertos eventos.

En la Figura 43 se puede ver la interfaz de configuración de un flujo de trabajo implementado y que se inicia cada vez que se crea un ticket. Primero se define el evento que da inicio al flujo, opcionalmente pueden agregarse condiciones del evento, luego se define el comportamiento a seguir.

PRODUCTIVIDAD

Resuestas preparadas
Resuestas guardadas
Etiquetas
Tipos de tickets
Flujos de trabajo

← atrás

Editar flujo de trabajo

Nombre

Ticket Created

Descripción

Automate actions when ticket is created.

Estado del flujo de trabajo

☒ El flujo de trabajo está activo

Eventos

Un evento se activa automáticamente para verificar las condiciones y realizar un conjunto de acciones predefinidas

Ticket

Ticket creado

✕

+ Añadir Más

Condiciones

Las condiciones son un conjunto de reglas que verifica escenarios específicos y se activan en ocasiones específicas

+ O

+ Y

Comportamiento

Una acción no solo reduce la carga de trabajo, sino que también facilita la automatización de tickets

Enviar al cliente

Ticket generated succes

✕

Asignar a agente

Soporte TrAC

✕

Enviar al agente

Seleccionar opción

Ticket Assign

✕

+ Añadir Más

Figura 43: Ejemplo Interfaz de configuración de Flujo de trabajo – Ticket Creado

Se crearon tres flujos de trabajo que se detallan a continuación y se ilustran mediante diagramas de actividad para una mejor comprensión.

Ticket creado

Cuando se crea un ticket el sistema se encarga de notificar mediante un correo electrónico al Soporte, ver Figura 44.

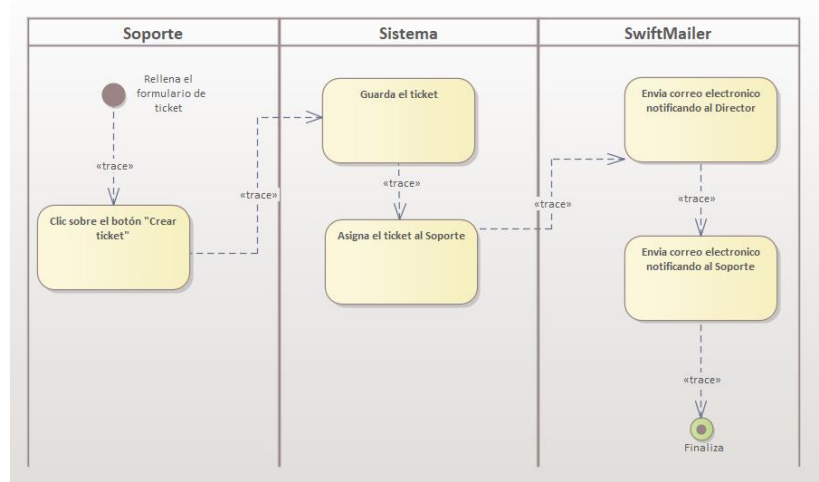


Figura 44: Diagrama de actividad de flujo de trabajo de la creación de ticket.

Soporte responde a ticket

Cuando el Soporte responde al ticket, se notifica al Director mediante un correo electrónico, ver Figura 45.

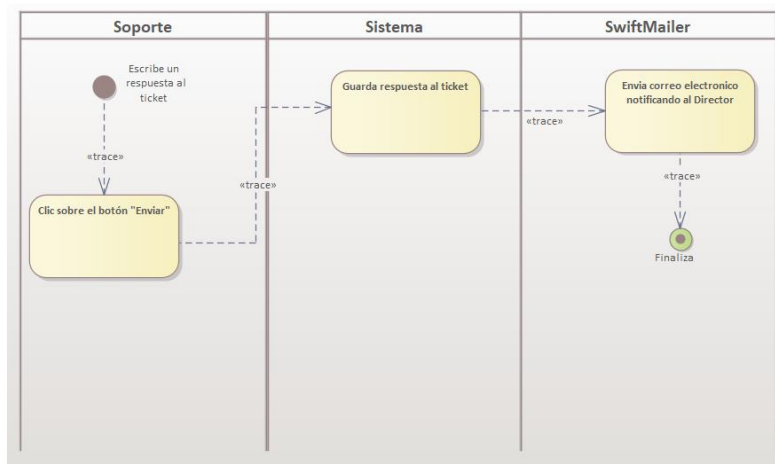


Figura 45: Diagrama de actividad de flujo de trabajo de Soporte responde a ticket.

Director responde a ticket

Cuando el Director responde al ticket se notifica al Soporte mediante un correo electrónico, ver Figura 46.

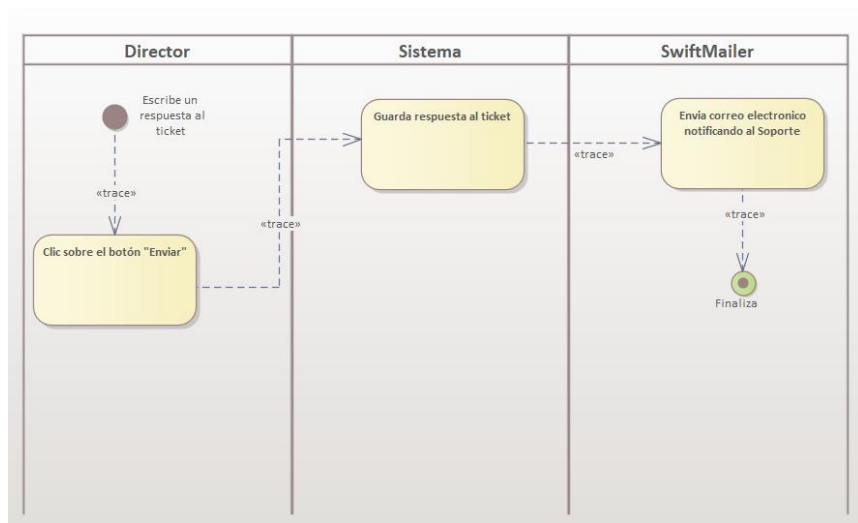


Figura 46: Diagrama de actividad de flujo de trabajo de Director responde a ticket.

4.3.5.4 Respuestas guardadas

UVdesk ofrece además la creación de repuestas guardadas, esto permite a los Soporte responder con mayor facilidad a las solicitudes de los usuarios. En la Figura 47 se puede ver un ejemplo de una respuesta guardada que es utilizada para responder a los usuarios que han solicitado el desbloqueo de su cuenta o actualización de contraseña.

PRODUCTIVIDAD < atrás

Respuestas preparadas

Respuestas guardadas

Etiquetas

Tipos de tickets

Flujos de trabajo

Editar respuesta guardada

Nombre

desbloqueo de cuenta

Nombre de respuesta guardado

Grupos

Compartir respuesta guardada con usuario(s) de estos grupos

SOPORTE x

Equipo

Compartir respuesta guardada con usuario(s) de estos equipos

Cuerpo

Hemos gestionado su solicitud.

Debe haber recibido un correo electrónico con un enlace para reestablecer su contraseña.

Si no logra visualizar un correo con título "Activación cuenta LALA TRAC" en su bandeja principal, por favor revise también la bandeja de "correo no deseado" o "SPAM".

Para más detalles de cómo activar su cuenta lo invitamos a que revise la información descrita en este [artículo](#).

Cuerpo de respuesta guardado

GUARDAR CAMBIOS

Figura 47: Ejemplo respuesta guardada - Desbloqueo de cuenta

4.3.5.5 Tipos de tickets

Se clasificaron y configuraron los tipos de tickets que el usuario puede crear. Esta funcionalidad permite categorizar las solicitudes y brindar información previa al Soporte sobre la naturaleza del ticket. Los tipos definidos pueden verse en la Figura 48.

Código	Descripción
CONSULTA	Consulta sobre el uso de TrAC-FID
REPORTAR ERRORES	Tipo de ticket utilizado para el reporte de errores.
GESTIÓN DE CUENTAS DE USUARIO	Gestión de usuarios nuevos y desbloqueo de ya creados
GESTIÓN DE CARRERAS	Solicitar cambios en las carreras que ve un director. Quitar y/o agregar alguna
OTRAS SOLICITUDES	Solicitudes no canalizadas por los otros tipos de tickets
CREAR CUENTA	Solicitud de creación de cuenta
RECUPERAR CONTRASEÑA	Solicitud de recuperación de contraseña

Figura 48: Tipos de tickets

4.3.5.6 Persistencia de datos

Cada vez que se realiza una configuración en la plataforma UVdesk se modifica directa o indirectamente un archivo del proyecto. Dada la arquitectura descrita anteriormente, la mesa de ayuda se encuentra siendo ejecutada dentro de un contenedor Docker, entonces los cambios que se realizan en la plataforma están siendo guardados dentro del mismo. Entonces surge el siguiente problema: El contenedor Docker debe ser reiniciado para poder cargar los nuevos cambios, esto hace que las configuraciones guardadas dentro del contenedor desaparezcan, por lo que deben volver a realizarse. Para resolver a este problema se utilizó la propiedad volúmenes¹³ de Docker, que permite indicarle al contenedor que cuando se construya guarde ciertos archivos dentro de una carpeta en la máquina que lo contiene y que cada vez que quiera acceder a estos los lea en esa ruta.

Además, se utilizó otro volumen para persistir la información de la base de datos. El resultado a nivel de arquitectura es que se muestra en la Figura 49.

¹³ Los volúmenes de Docker son utilizados para persistir los datos que manejan los contenedores. Esto nos permite crear contenedores e indicarles donde tienen que persistir nuestros datos.

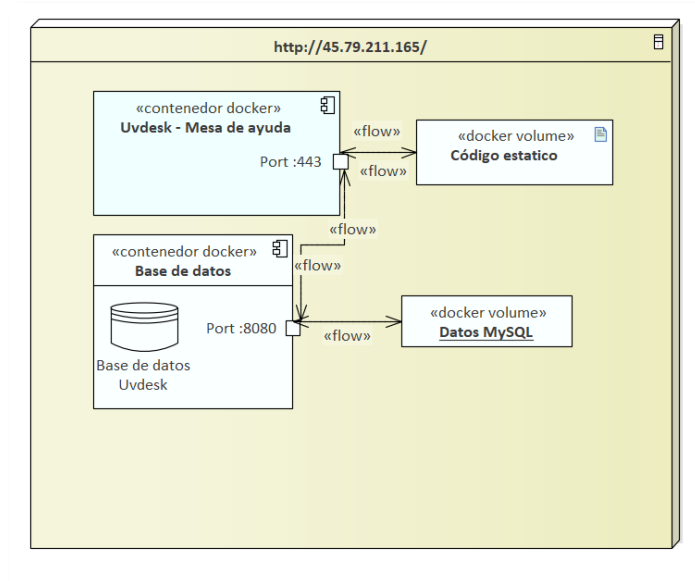


Figura 49: Arquitectura con volúmenes Docker

4.3.6 Comunicación de TrAC-FID con mesa de ayuda UVdesk

Considerando que deberían existir dos vías de acceso a la mesa de ayuda (usuario con sesión y usuario sin sesión). Se generaron dos flujos de acceso a la mesa de ayuda desde TrAC-FID.

4.3.6.1 Acceso sin sesión iniciada en TrAC-FID

El primer flujo de acceso se realiza a partir de un botón tipo texto implementado en el panel de inicio de sesión de TrAC-FID, ver Figura 50.

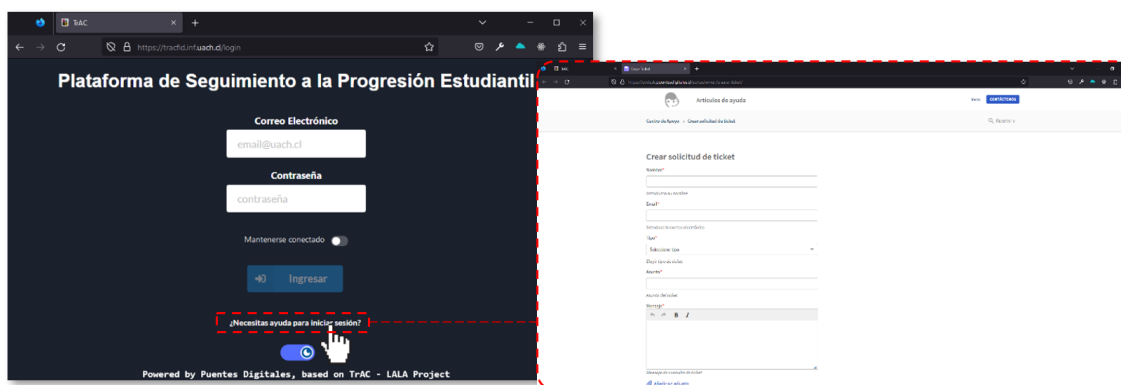


Figura 50: Acceso a la mesa de ayuda sin sesión iniciada en TrAC-FID

El texto corresponda a la pregunta “¿Necesitas ayuda para iniciar sesión?” y puede ser editado desde el panel de administrador.

Bajo este flujo de acceso al presionar el texto se abre una nueva pestaña con una interfaz para crear un ticket y acceder a los manuales de usuario, pero de manera anónima (sin sesión iniciada en la mesa de ayuda).

4.3.6.2 Acceso a la mesa de ayuda con sesión iniciada en TrAC-FID.

Para iniciar una sesión en la mesa de ayuda es necesario verificar las credenciales del usuario que intenta ingresar (usuario y contraseña). UVdesk no puede acceder a la base de datos de TrAC-FID para realizar esta verificación, por motivos de seguridad y por incompatibilidad de la estructura de las tablas con datos. Entonces es necesario que las credenciales del usuario que intenta acceder a la mesa de ayuda existan en ambas bases de datos. No se puede copiar la tabla de datos por incompatibilidad, se podrían haber creado los usuarios manualmente desde la interfaz gráfica de UVdesk, pero este proceso podría resultar extenuante y agregaría labores al equipo de soporte, específicamente, que cada vez que se cree un usuario en TrAC-FID deba crearse en la mesa de ayuda. Como ninguna de las opciones antes mencionadas, resulta viable, se decidió crear un API encargada de cumplir esta labor de manera automática.

Para evitar que el usuario deba ingresar sus credenciales manualmente en el sistema de UVdesk, es TrAC-FID quién se encarga de enviar un archivo HTML con estos valores.

Además, cuando se crea un usuario en UVdesk, la contraseña pasa por un proceso de transformación que convierte la contraseña en un hash¹⁴ y es guardada en la base de datos. Esta técnica de *hashing* también se utiliza para verificar las credenciales del usuario durante el inicio de sesión. Como la API no puede acceder a la función hash de la mesa de ayuda, tampoco puede transformar la contraseña recibida por TrAC-FID, y si se decidiera guardar sin transformación el inicio de sesión generaría un error, por lo que se llegó a la conclusión de que la contraseña debería estar “hasheada” al momento de enviarse desde TrAC-FID. Para ello se creó un usuario manualmente en la mesa de ayuda y se copió el hash creado por el sistema, este hash es usado para crear y verificar los inicios de sesión de las cuentas de todos los usuarios. Se utilizó un hash diferente para los usuarios administradores.

¹⁴ Una función criptográfica hash- usualmente conocida como “hash”- es un algoritmo matemático que transforma cualquier bloque arbitrario de datos en una nueva serie de caracteres con una longitud fija

La API fue desarrollada con JavaScript y NodeJS y fue puesta en funcionamiento con un contenedor Docker corriendo en el servidor de la mesa de ayuda.

En TrAC-FID se implementó un botón encargado de desplegar la mesa de ayuda, ver Figura 51.

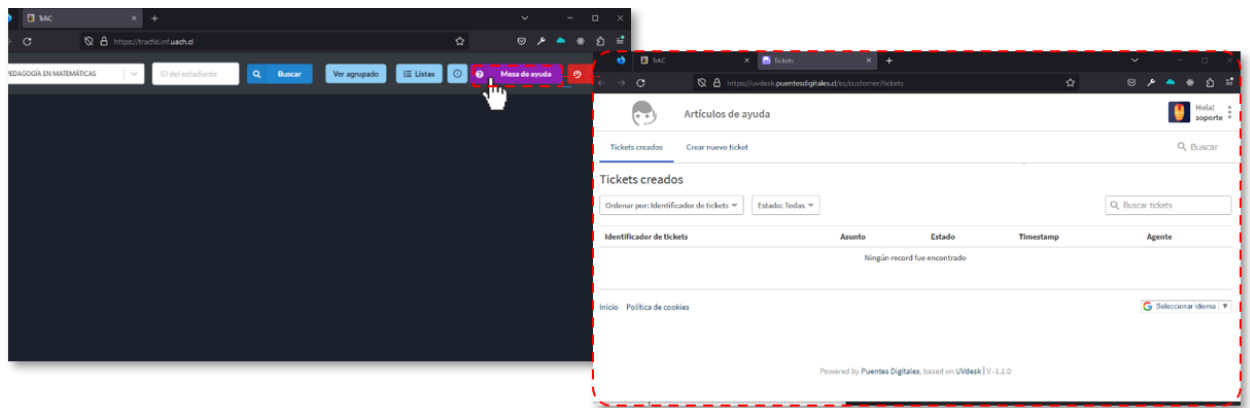


Figura 51: Acceso a la mesa de ayuda con sesión iniciada en TrAC-FID

El diagrama de secuencia de la Figura 52 retrata la lógica implementada que consiste en los siguientes pasos, al presionarse el botón TrAC-FID se comunica con la API de la mesa de ayuda, indicándole que el usuario desea acceder, además indica en esta llamada el tipo de usuario. La API de la mesa de ayuda consulta en la base de datos si el usuario en cuestión existe, de no ser así, lo crea y reporta cómo exitosa la transacción. La API de TrAC-FID envía las credenciales a la mesa de ayuda, desplegando una interfaz diferente según el tipo de usuario, administrador o director.

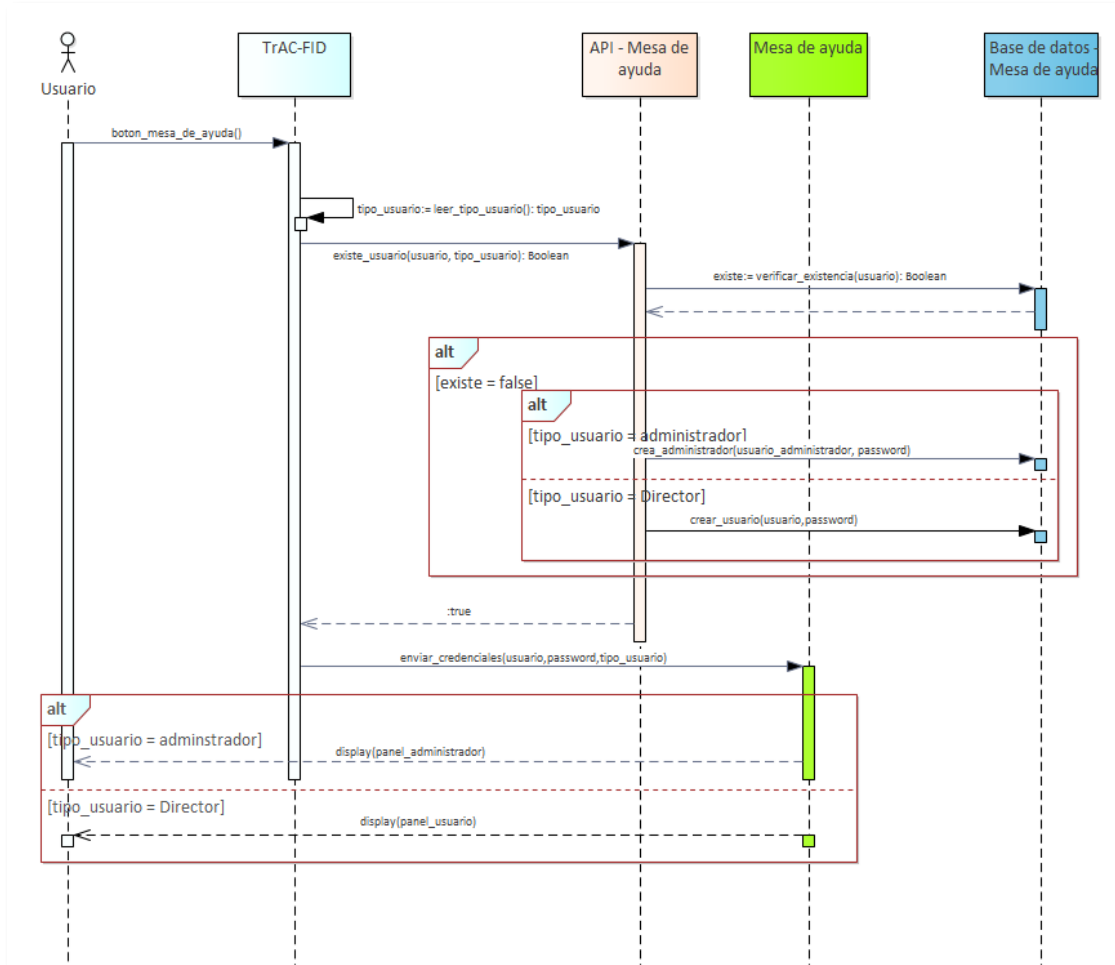


Figura 52: Diagrama de secuencia de acceso de usuario con sesión iniciada en TrAC-FID a la mesa de ayuda.

La Figura 53 corresponde a la interfaz desplegada a los directores y la Figura 54 a la interfaz desplegada a los usuarios administradores o soportes.

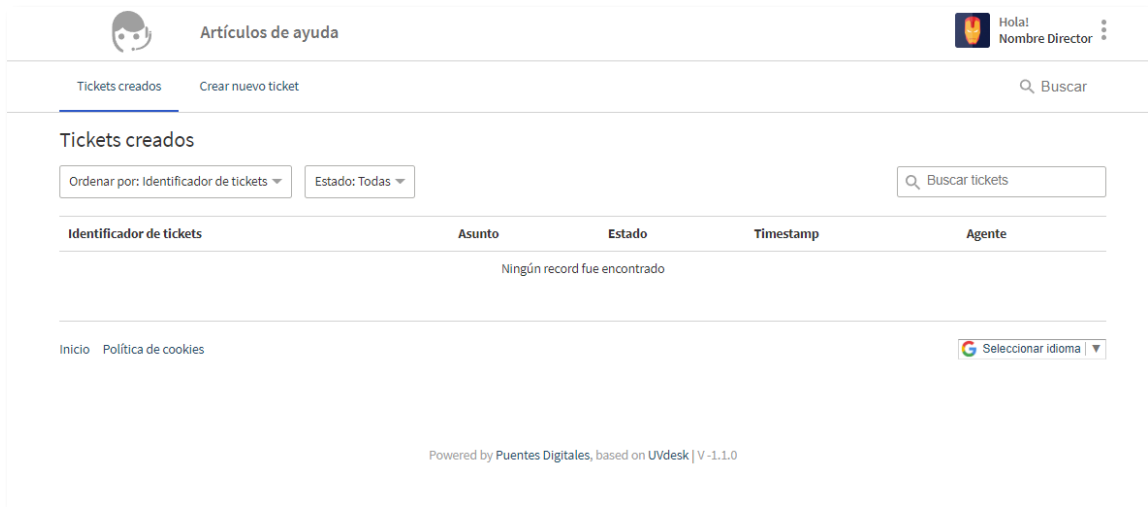


Figura 53: Interfaz usuario Director

La interfaz de usuario director permite la creación de un nuevo ticket y el acceso a los artículos de ayuda o manuales de usuario.

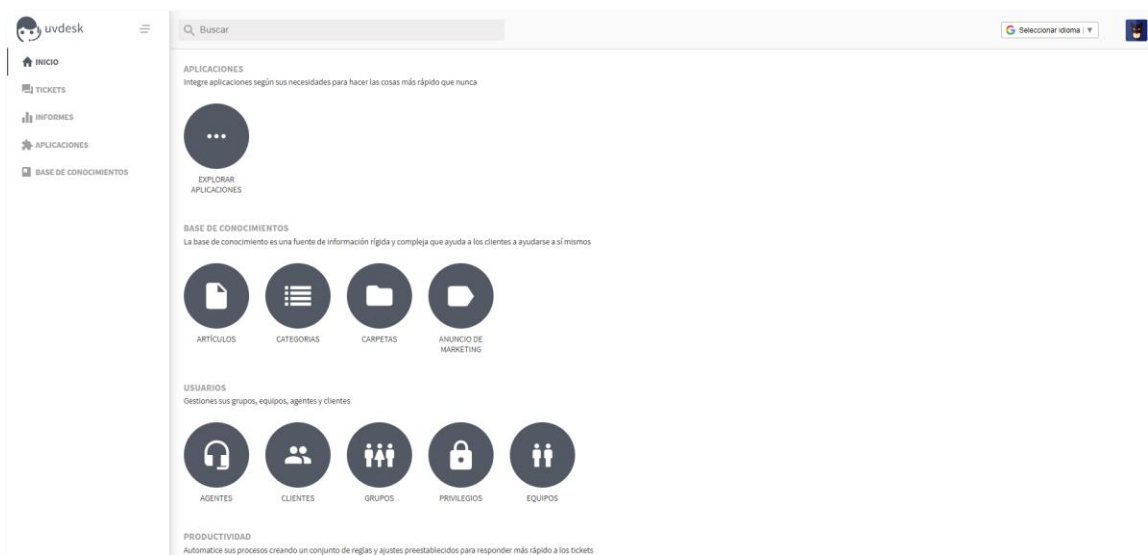


Figura 54: Interfaz usuario Soporte o administrador.

La interfaz de soporte permite acceder a las configuraciones de la plataforma, a los tickets, a la gestión de artículos y otras funciones como informes y configuración de módulos extras.

4.3.6.3 Dominio seguro y monitor de estado

Se implementó un certificado de seguridad en el servidor web de la mesa de ayuda y se dio un nombre de dominio a la IP para que sea leído por los servicios DNS. El certificado de la mesa de ayuda permite a los navegadores mostrar la plataforma web cómo un sitio seguro, y evitar alertas relacionadas con vulnerabilidades de seguridad que puedan comprometer la fiabilidad del sistema.

Junto con esto se implementó un servicio de análisis continuo del estado de los servidores de UVdesk y TrAC-FID. Este servicio se encarga de notificar a los encargados de soporte en el caso de que los servicios no se encuentren disponibles para así tomar acciones de manera inmediata.

4.3.6.4 Arquitectura final

El resultado final de la arquitectura implementada se puede ver en la Figura 55, en esta se puede ver que además de los contenedores con la mesa de ayuda y la base de datos se encuentra otro conteniendo la API desarrollada para la creación de usuarios. Las llamadas son estrictamente realizadas desde TrAC-FID hacia la API y la mesa de ayuda UVdesk. Además, se puede ver que el servidor de la mesa de ayuda ya no es accedido por la IP, sino que por su nombre de dominio.

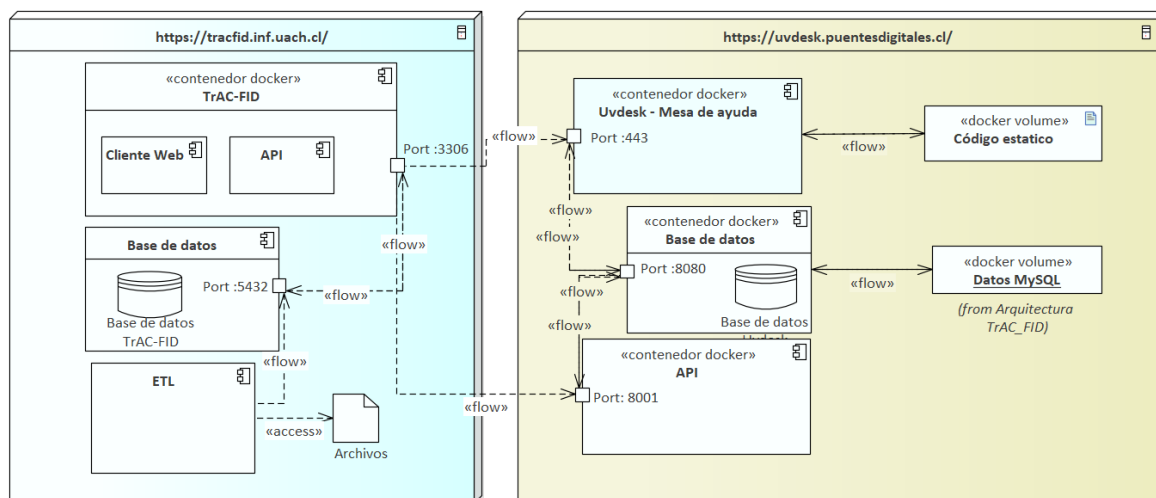


Figura 55: Arquitectura y comunicación TrAC-FID y UVdesk

5 VALIDACIÓN

En este capítulo se presentan distintas actividades realizadas con el fin de validar las capacidades de los sistemas implementados, desde pruebas de interfaz gracias hasta análisis de carga y concurrencia de usuarios.

5.1 Pruebas de usabilidad

La primera parte de esta etapa estuvo relacionada con la validación de las interfaces gráficas desarrolladas. Con el objetivo de conocer si cumplían con los requisitos de usabilidad se realizaron actividades en las que se pedía a los usuarios interactuar con el sistema y posteriormente rellenar un cuestionario, este tipo de pruebas se conoce con el nombre de pruebas de usabilidad de pasillo¹⁵. Se utilizó esta prueba ya que provee un formato general que se acomoda con el hecho de que se consideró que las funcionalidades que se muestran en las interfaces a probar son simples y no requieren de grandes conocimientos técnicos.

Esta prueba fue aplicada en su mayoría a estudiantes aleatorios de la Universidad Austral de Chile. Las pruebas fueron desarrolladas de la siguiente manera:

Primero se desarrolló una lista de actividades que corresponden a las principales utilidades de la mesa de ayuda y el módulo de notificaciones, las actividades se pueden ver en la

¹⁵ Las pruebas de usabilidad de pasillo refieren a un método para probar la funcionalidad de un sitio web, una aplicación u otro producto, y consisten en evaluar un producto o servicio probándolo con usuarios representativos reales mientras intentan completar tareas en él. El objetivo es identificar cualquier problema de usabilidad, recoger datos cualitativos y cuantitativos, así como determinar la satisfacción del participante con el producto.

Tabla 10.

Tabla 10: Lista de actividades de prueba de usabilidad módulo de notificaciones y mesa de ayuda

Actividades	Módulo para probar	
	Notificaciones	Mesa de ayuda
A1	Verificar el estado de las notificaciones	Crear un ticket en la mesa de ayuda
A2	Leer la respuesta del sistema e interpretar	Agregar una imagen al ticket
A3	Reenviar una notificación	Ver la respuesta al ticket creado
A4		Agregar una respuesta final al ticket
A5		Acceder a la base de conocimiento (Manuales de usuario)
A6		Acceder a un artículo

Además, haciendo uso del modelo SUS se desarrolló en la plataforma Formularios¹⁶ de Google un cuestionario para evaluar los puntajes de usabilidad, este cuestionario consta de 10 afirmaciones

1. Creo que me gustaría utilizar este sistema con frecuencia
2. Encontré el sistema innecesariamente complejo
3. Pensé que el sistema era fácil de usar
4. Creo que necesitaría el apoyo de un técnico para poder utilizar este sistema
5. Encontré que las diversas funciones de este sistema estaban bien integradas
6. Pensé que había demasiada inconsistencia en este sistema
7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente
8. Encontré el sistema muy complicado de usar
9. Me sentí muy seguro usando el sistema
10. Necesitaba aprender muchas cosas antes de empezar con este sistema

El valor de cada afirmación se mide siguiendo la Escala de Likert (Busquets, 2022) donde el usuario deberá seleccionar un valor entero entre 1 y 5, incluyéndolos y donde el 1 significa Totalmente en desacuerdo y el 5 totalmente de acuerdo y los numero intermedios se encuentran asociados a la aseveración de los extremos según su distancia con el valor, tal cómo se puede ver en la Figura 56.

¹⁶ Formularios de Google es un software de administración de encuestas que se incluye como parte del conjunto gratuito Google Docs Editors basado en la web que ofrece Google.

"Creo que me gustaría usar este sistema con frecuencia." *

1 2 3 4 5

Totalmente en desacuerdo ☐ ☐ ☐ ☐ ☐ Totalmente de acuerdo

Figura 56: Ejemplo afirmación en Formulario de Google

Una vez desarrollados los artefactos necesarios para la actividad, lista de pruebas y cuestionarios, se consultó a un grupo de estudiantes si deseaba participar, se fijaron fechas para realizar la reunión remota. Se realizó la reunión remota, en donde se comenzó explicando la actividad de manera general y evitando adelantar mucho sobre lo que se hará porque podría provocar que las pruebas sean pensadas antes de realizarlas. A medida que se desarrollaba la actividad se les pidió a los usuarios que realicen comentarios sobre la interfaz, cualquier detalle o hecho que haya llamado su atención en cuanto a usabilidad.

Luego de realizadas las reuniones se analizaron los cuestionarios. Para ello se utilizó el siguiente modelo de cálculo:

Cada enunciado tendrá una respuesta en la escala de Likert, que equivaldrá a 1, 2, 3, 4 o 5, en función de la respuesta. Estos son los pasos para obtener un valor final:

- Suma las respuestas de los enunciados impares y después resta 5
- Suma las respuestas de los enunciados pares y resta ese total a 25
- Suma ambos resultados y multiplícalo por 2,5.

Tras un estudio realizado en más de 500 webs y aplicaciones, Jeff Sauro concluye que el puntaje promedio es 68. Un resultado por debajo de esta cifra indica que hay varios aspectos a corregir. (Busquets, 2022).

En la Figura 57 se puede una representación gráfica del resultado SUS.

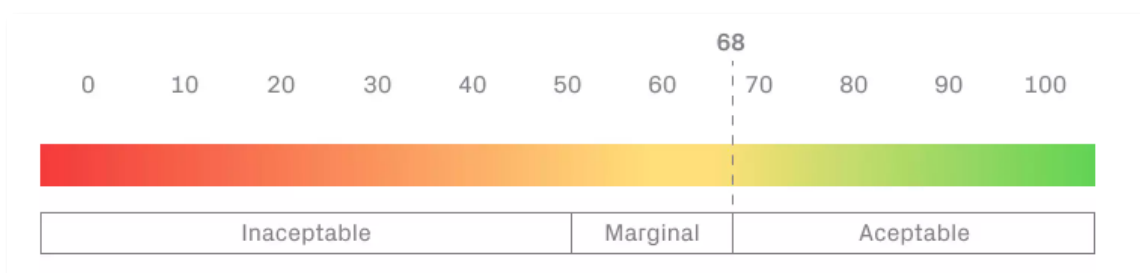


Figura 57: Representación de resultado SUS

A continuación, se detallan y analizan los puntajes obtenidos en cada una de las pruebas.

5.1.1 Mesa de ayuda

En la Tabla 11 detalla los resultados obtenidos en las pruebas de usabilidad de la mesa de ayuda, los valores A1 a A10 corresponden a las afirmaciones y los valores P1, P2, P3, P4, P5 y P6 corresponde a participantes, por lo que si interceptamos una columna P con una A tendríamos el valor que seleccionó el participante P en afirmación A. Luego haciendo uso del cálculo antes descrito se obtuvieron los siguientes resultados.

Tabla 11: Resultados pruebas de usabilidad de pasillo mesa de ayuda

	P1	P2	P3	P4	P5	P6
A1	5	5	4	4	5	4
A2	1	3	1	1	1	1
A3	5	4	5	5	5	5
A4	1	2	1	2	1	1
A5	5	4	5	5	4	3
A6	1	1	1	1	1	1
A7	4	5	5	5	5	5
A8	1	1	1	1	1	1
A9	5	4	4	5	5	4
A10	1	1	1	1	1	1
SUS	69	59,5	68	66,5	69	66

Se obtuvo un promedio de 66 SUS lo que ubica a la mesa de ayuda bajo el promedio establecido (68 puntos). Una de las posibles causas de esta valoración se deba a lo que muchos usuarios comentaron sobre la manera de llegar a los artículos de ayuda. Se consideró que el elemento era poco llamativo y se encontraba ubicado en una locación poco intuitiva, algunos usuarios mencionaron que el hacer clic en esos elementos lo relacionan con visitar la página de inicio, este elemento puede verse en la Figura 58.

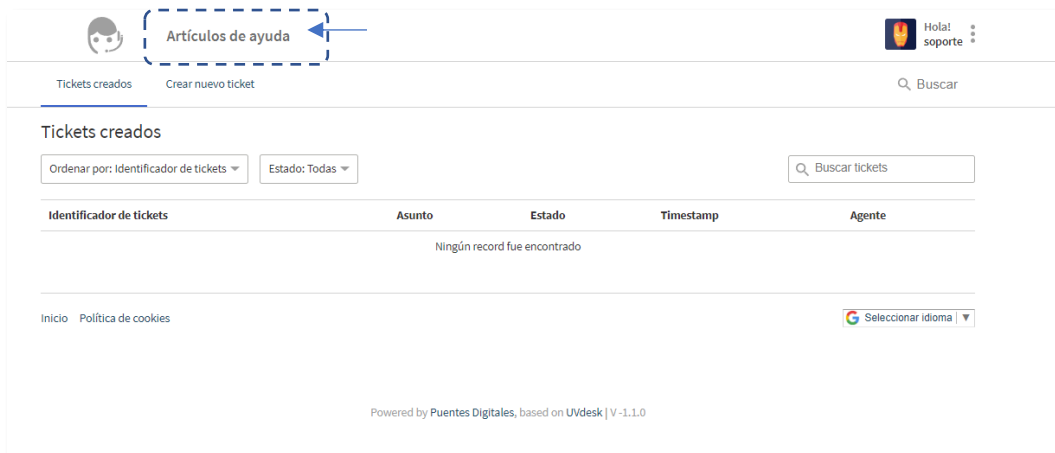


Figura 58: Interfaz de usuario mesa de ayuda - Artículos de ayuda

5.1.2 Notificaciones

Con respecto a las pruebas de usabilidad relacionadas con el módulo de notificaciones. Se consideró cómo no necesario realizar pruebas de usabilidad puesto que la interfaz reutilizaba lógicas de los otros elementos visuales, por lo que se estaría poniendo a prueba el diseño de TrAC. De todas maneras, se aplicó la prueba a la encargada de realizar el envío de notificaciones en el servidor de producción a la fecha de desarrollado este documento. El resultado se puede ver en Tabla 12.

Tabla 12: Resultados pruebas de usabilidad de pasillo módulo de notificaciones

	P1
A1	5
A2	1
A3	5
A4	1
A5	4
A6	1
A7	4
A8	1
A9	4
A10	1
SUS	67

Quién le otorgó un valor de 67, un valor bastante cercano al aceptable (68), pero aún en representado por el grupo de valores marginales.

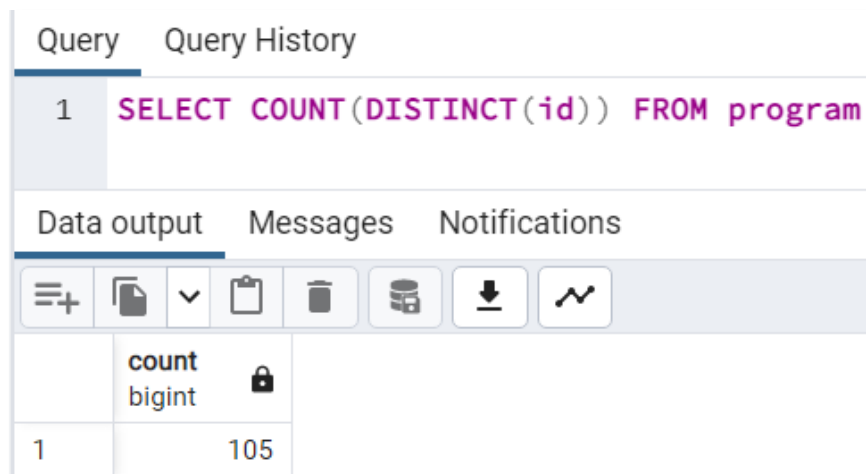
5.2 Mesa de ayuda: Pruebas de carga

La implementación de la mesa de ayuda requirió el desarrollo de una API que se encarga de la creación de usuarios. Cada vez que un usuario solicita ayuda, accede a esta API. Por esta razón, es esencial realizar pruebas de capacidad para determinar si puede manejar el escenario de mayor demanda en el que todos los usuarios intentan acceder simultáneamente.

Una prueba de carga es un tipo de prueba de rendimiento que verifica cómo funcionan los sistemas bajo una gran cantidad de usuarios virtuales simultáneos que realizan transacciones durante un cierto período de tiempo (Atentus, 2022).

Para llevar a cabo esta prueba se utilizó JMeter. JMeter es una herramienta que se utiliza para realizar pruebas de carga de aplicaciones web para conocer la cantidad de usuarios simultáneos que pueden manejar en tiempo real (LoadView, 2020).

Primero fue necesario identificar el número máximo de usuarios que podrían acceder a la API de manera concurrente, para ello se identificó el número total de carreras que se encontraban de momento en los registros de la base de datos mediante una consulta SQL a la tabla *Programs*, lo que arrojó un total de 105 carreras, ver Figura 59.



The screenshot shows a database query interface. At the top, there are tabs for 'Query' and 'Query History'. The 'Query' tab is active, displaying a SQL query: `1 SELECT COUNT(DISTINCT(id)) FROM program`. Below the query, there are tabs for 'Data output', 'Messages', and 'Notifications'. The 'Data output' tab is active, showing a table with one row and one column. The column is labeled 'count' with a data type of 'bigint' and a lock icon. The row contains the value '105'.

	count bigint
1	105

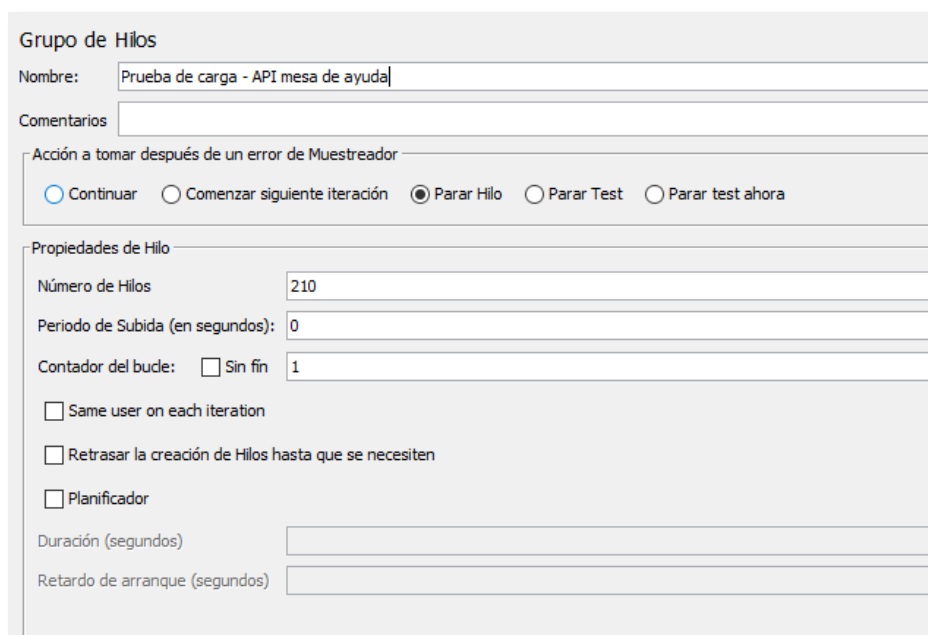
Figura 59: Cantidad de carreras registradas en la base de datos de TrAC-FID.

Se consideró que por cada carrera existe un único director, y que además estos pueden permitir el acceso de otro usuario, por lo que el máximo de llamadas a la API que se podrían realizar es de 210, dos usuarios por carrera. Considerando este valor se procedió a la configuración de JMeter para replicar ese número de llamadas.

Para probar la creación de usuarios es necesario que cada petición posea en su cuerpo un email distinto para ello se creó una lista de 210 correos electrónicos falsos con el formato “[numero](mailto:1@testtracfid.cl)”@testtracfid.cl, donde el valor número varía entre “1” y “210”, por ejemplo: 1@testtracfid.cl, 2@testtracfid.cl, 3@testtracfid.cl, y así sucesivamente hasta 210@testtracfid.cl .

La configuración de JMeter para realizar las 210 llamadas es la siguiente:

Se definió el número de hilos en 210 y el periodo de subida, que indica cada cuanto segundo enviar una petición, se dejó en 0 para que el envío se realice en el mismo instante. La configuración aplicada puede verse en la Figura 60.



Grupo de Hilos

Nombre:

Comentarios

Acción a tomar después de un error de Muestreador

☐ Continuar ☐ Comenzar siguiente iteración ☒ Parar Hilo ☐ Parar Test ☐ Parar test ahora

Propiedades de Hilo

Número de Hilos

Periodo de Subida (en segundos):

Contador del bucle: ☐ Sin fin

☐ Same user on each iteration

☐ Retrasar la creación de Hilos hasta que se necesiten

☐ Planificador

Duración (segundos)

Retardo de arranque (segundos)

Figura 60: Configuración de hilos JMeter

Luego de configurados los hilos, se procedió a la parametrización de las peticiones de cada uno. Se definió el Servidor, UVdesk.puentesdigitales.cl, puerto 8001 y ruta “/verifyCustomer” correspondientes a la ruta de acceso a la API. Además, fue necesario adjuntar el archivo CSV¹⁷ con los 210 correos electrónicos creados. Se definió el nombre de la variable como “email” en la configuración del CSV, ver Figura 61, que es accedido

¹⁷ Los archivos CSV (del inglés comma-separated values) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas.

por los hilos de manera secuencial. La configuración detallada puede verse en la Figura 62.

Figura 61: Configuración Data set JMeter

Figura 62: Configuración petición http JMeter

Luego de realizada la configuración se iniciaron las pruebas. El sistema soportó las 210 llamadas de manera paralela de manera exitosa. En la Figura 63 se puede ver una consulta SQL ejecutada en la base de datos que indica que se crearon los 210 usuarios con éxito.

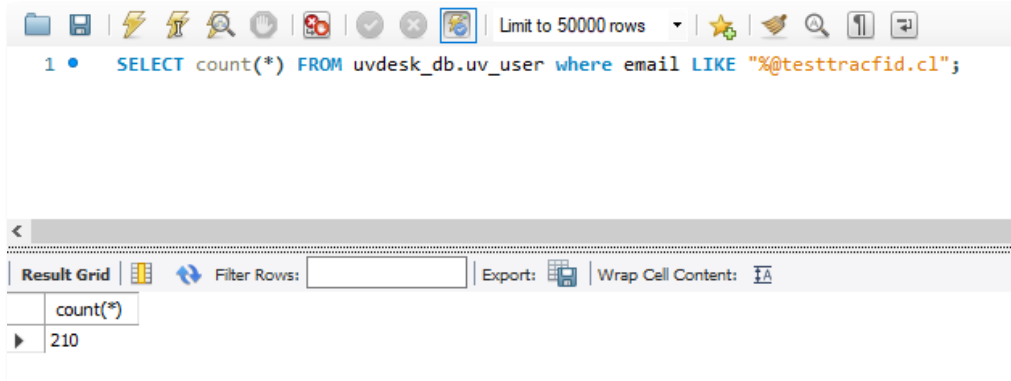


Figura 63: Resultado consulta SQL para verificar que se crearon los usuarios

En la Figura 64 se muestran los resultados arrojados por JMeter en donde cómo se puede ver en la columna “% Error” no existieron peticiones erróneas.

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: ☐ Escribir en Log Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Petición HTTP API...	210	2544	2139	2965	216,00	0,00%	70,8/sec	19,62	16,75	284,0
Total	210	2544	2139	2965	216,00	0,00%	70,8/sec	19,62	16,75	284,0

Figura 64: Resultado resumen prueba de carga JMeter

La prueba ha demostrado que el sistema puede manejar la cantidad máxima de usuarios simultáneos que buscan acceder a la mesa de ayuda, cumpliendo con los estándares de concurrencia requeridos.

5.3 Comentario de usuario

Luego de la implementación y puesta en marcha de la mesa de ayuda en entorno de producción, se recibió un correo por parte de uno de los usuarios finales de la plataforma implementada.

“¡¡Muchas gracias por la información!! Está genial el sistema de tickets de ayuda. “Me había demorado un poco buscando donde se llamaba, pero creo que no era literal lo del “llamadas a mesa de ayuda” jajaja. Está súper, sobre todo porque pueden ir navegando por temas en el manual de usuario...”

Correo electrónico enviado por usuario de la mesa de ayuda

Este mensaje muestra la satisfacción del usuario con la mesa de ayuda y su experiencia positiva al interactuar con ella. Con un tono entusiasta y agradecido, el mensaje se percibe como una validación por parte de él, además menciona positivamente los manuales de usuario.

6 CONCLUSIONES

6.1 Cumplimiento de objetivos

El objetivo general de este proyecto de título fue “Apoyar a los Directores de Escuela en la toma de decisiones oportunas respecto a situaciones irregulares de los estudiantes mediante el desarrollo de un módulo de notificaciones en TrAC-FID. Además, ofrecer una mesa de ayuda eficaz y eficiente con un foco en los usuarios finales”. Dicho objetivo se concluyó con éxito tras la implementación de ambos módulos y el cumplimiento de los objetivos específicos definidos.

- **Conocer el estado actual de las tecnologías asociadas al desarrollo del sistema TrAC-FID.**

Las tecnologías con las cuales fue construido TrAC-FID aún forman parte de las tecnologías más utilizadas a la fecha de escrito este documento. Estas responden a necesidades de manejo de grandes cantidades de datos, con lógicas inteligentes y enfocadas en la reutilización y eficiencia, que se transmite al usuario a través de interfaces más rápidas y amigables.

El análisis de estas tecnologías permitió comprender cómo funciona el sistema de TrAC-FID internamente, cómo se comunican las componentes y cómo cada tecnología cumple un rol fundamental en la entrega del servicio, con el fin de poder implementar el módulo de notificaciones e integrarlo al sistema de la manera más orgánica posible.

- **Identificar la lógica de los indicadores e información que es generada actualmente por el sistema y que será utilizada para generar las notificaciones.**

Además de estudiar las tecnologías fue necesario realizar un análisis de la lógica utilizada por el sistema, la cual además de apegarse a estructuras y formas de organizar el código responde a la capacidad de resolver problemas de sus desarrolladores.

Este objetivo permitió identificar que componentes dentro de la arquitectura de TrAC-FID podrían reutilizarse o replicar su lógica para construir el módulo de notificaciones.

Además, el primer y segundo objetivo específico se ven reflejados cómo cumplidos con la existencia del módulo de notificaciones.

- **Desarrollar el módulo de notificaciones para alertar de situaciones potencialmente riesgosas.**

Gracias a la metodología de trabajo utilizada fue posible desarrollar un sistema de notificaciones completamente apegado a las necesidades de los usuarios reales. Las reuniones semanales permitieron mantener el rumbo del desarrollo siempre en dirección a la solución óptima. El desarrollo abarcó desde la generación de estructuras en la base de datos hasta la creación de componentes visuales y que interactúan con el usuario, pasando por la API y cada uno de sus elementos.

- **Determinar los requerimientos e identificar tecnologías compatibles con el sistema TrAC-FID que permitan el desarrollo de una mesa de ayuda.**

Los requerimientos de la mesa de ayuda fueron determinados gracias al documento de solicitud PI FID y a las reuniones semanales con la *Product Owner*, quién se encargó de transmitir y velar por que se responda a las necesidades reales de los usuarios. En base a esto se creó una tabla de análisis de tecnologías que permitió evaluar plataformas que podrían responder a la necesidad de una mesa de ayuda.

- **Desarrollar mesa de ayuda para facilitar la comunicación entre los usuarios y el equipo técnico que opera TrAC-FID.**

La mesa de ayuda implementada cumple con creces los requisitos establecidos, además de ofrecer funcionalidades que facilitan aún más la comunicación entre los Soporte y los Directores. Se puede acceder directamente desde la plataforma TrAC-FID y no requiere de inicio de sesión para gestionar a los tickets propios de cada usuario. Junto a esto se desarrollaron manuales de usuario con el fin de ofrecer información de manera expedita a los usuarios y así evitar el proceso de comunicación con un encargado de soporte.

La mesa de ayuda integra una funcionalidad de lectura y envío de correos electrónicos automáticos, lo que permite a los usuarios responder desde la plataforma o desde sus correos electrónicos, lo que facilita aún más la velocidad de respuesta. Esta comunicación se puede realizar por ambos medios y aun así se encuentra contenida y siempre es posible visualizarla en la mesa de ayuda.

- **Validar el impacto real de las mejoras al sistema.**

Para evaluar el nivel de usabilidad de ambos sistemas, se llevaron a cabo pruebas de usabilidad de pasillo. En estas pruebas, se prestó especial atención a la mesa de ayuda, invitando a varios usuarios a interactuar con las interfaces y luego completar un cuestionario. Esto permitió obtener un puntaje del nivel de usabilidad de cada sistema.

Además, se realizó una prueba de carga en la API implementada para verificar que es capaz de dar respuesta a la mayor cantidad de usuarios que podrían conectarse con la mesa de ayuda en paralelo, en esta prueba se simulaban 210 llamadas a la mesa de ayuda que fueron resueltas sin errores por la API.

Durante el desarrollo de este proyecto, cada avance fue objeto de rigurosas evaluaciones tanto por parte del desarrollador como del Product Owner. Estas evaluaciones aseguraron el cumplimiento de los requisitos técnicos y funcionales antes de ser liberados al entorno de producción para su uso por parte de los usuarios. De esta manera, se garantizó que cada nueva funcionalidad cumpliera con los estándares establecidos y brindara una experiencia de usuario satisfactoria.

Al momento de desarrollado este documento, todas las funcionalidades descritas en este proyecto están siendo utilizadas por usuarios reales.

6.2 Trabajos futuros

Se considera que el trabajo realizado durante este proyecto satisface las necesidades de los usuarios, sin embargo, existen mejoras que aún podrían realizarse con el fin de hacer los sistemas más independientes de tareas manuales.

Para el módulo de notificaciones podrían realizarse la siguiente mejora:

- Automatizar el envío de notificaciones al realizar la carga periódica de archivos, es decir que el sistema automáticamente realice el envío de los correos electrónicos y que esta funcionalidad no dependa del botón en la pestaña de Notificaciones del panel de administrador. Se debería considerar además que se debe enviar el correo con las fechas de cargas y alertas a los usuarios que se han creado recientemente.
- Permitir desde la interfaz gráfica configurar reglas y ajustes de las notificaciones cómo activar o desactivar el envío de notificaciones por usuario y ajustar los umbrales definidos para considerar los alumnos en riesgo para las diferentes carreras.

Los trabajos futuros o mejoras que podrían realizarse en la mesa de ayuda son los siguientes:

- La mesa de ayuda permite a la creación de manuales, pero de momento se muestra solo el de usuarios Directores y no el de operación, la razón de esto es que este último no debe mostrarse a los Directores, si no que solo a los Soportes. El sistema

actualmente no cuenta con una funcionalidad que permita definir que manuales mostrar dependiendo del perfil de usuario.

- Automatizar la actualización del certificado de seguridad que debe realizarse mediante la terminal de comandos del contenedor de UVdesk de manera manual.

7 REFERENCIAS

- Acibeiro, M. (07 de 02 de 2022). *Qué es Node.js y para qué sirve*. Obtenido de Qué es Node.js y para qué sirve: https://www.lucushost.com/blog/que-es-node-js/#Que_es_Nodejs
- Atentus. (10 de Noviembre de 2022). *Pruebas de carga: ¿Qué es y cómo funciona?* Obtenido de Pruebas de carga: ¿Qué es y cómo funciona?: <https://atentus.com/2022/11/10/pruebas-de-carga/>
- Busquets, C. (2022). *Escala de Likert: qué es y cómo usarla en investigación UX*. Obtenido de Escala de Likert: qué es y cómo usarla en investigación UX: <https://www.uifrommars.com/escala-likert-que-es-investigacion-ux/>
- Cowan, A. (2022). *Venture Design Process*. Obtenido de Venture Design Process: <https://www.alexandercowan.com/venture-design/>
- Facebook Open Source. (s.f.). *React*. Obtenido de React: <https://es.reactjs.org/>
- Fernández, Y. (23 de Agosto de 2019). *API: qué es y para qué sirve*. Obtenido de API: qué es y para qué sirve: <https://www.xataka.com/basics/api-que-sirve>
- Google, LLC. (2023). *Google Workspace*. Obtenido de Google Workspace: https://workspace.google.com/intl/es-419/?utm_source=google&utm_medium=cpc&utm_campaign=latam-CL-all-es-dr-bkws-all-all-trial-e-dr-1605540-LUAC0011978&utm_content=text-ad-none-any-DEV_c-CRE_471186719512-ADGP_Hybrid+%7C+BKWS+-+EXA+%7C+Txt+~+Google+Workspac
- Grafana Labs. (2022). *Stress testing*. Obtenido de Stress testing: <https://k6.io/docs/es/tipos-de-prueba/stress-testing/#:~:text=%C2%BFQu%C3%A9%20es%20la%20prueba%20de,del%20sistema%20en%20condiciones%20extremas.&text=determinar%20c%C3%B3mo%20se%20comportar%C3%A1%20su%20sistema%20en%20condiciones%20extremas.>
- Khailee. (s.f.). *Josh Elman's Approach to Growth*. Obtenido de Josh Elman's Approach to Growth: <https://sites.google.com/site/growthhacking/josh-elman>

- Kinsta. (1 de Febrero de 2021). *¿Qué Es Nginx y Cómo Funciona?* Obtenido de ¿Qué Es Nginx y Cómo Funciona?: <https://kinsta.com/es/base-de-conocimiento/que-es-nginx/>
- LoadView. (4 de Noviembre de 2020). *5 ejemplos de pruebas de carga de JMeter*. Obtenido de 5 ejemplos de pruebas de carga de JMeter: <https://www.loadview-testing.com/es/blog/5-ejemplos-de-pruebas-de-carga-de-jmeter/#:~:text=JMeter%20es%20una%20herramienta%20que,identificar%20la%20capacidad%20de%20carga.>
- Microsoft. (01 de Enero de 2023). *Dockerfile en Windows*. Obtenido de Dockerfile en Windows: <https://learn.microsoft.com/es-es/virtualization/windowscontainers/manage-docker/manage-windows-dockerfile>
- Microsoft. (s.f.). *TypeScript is JavaScript with syntax for types*. Obtenido de TypeScript is JavaScript with syntax for types.: <https://www.typescriptlang.org/>
- Red Hat. (s.f.). *¿Qué es Docker y cómo funciona?* Obtenido de ¿Qué es Docker y cómo funciona?: <https://www.redhat.com/es/topics/containers/what-is-docker>
- SendGrid Team. (19 de Noviembre de 2014). *Why the Notification is the Most Important Email You Can Send for Growth*. Obtenido de Why the Notification is the Most Important Email You Can Send for Growth: <https://sendgrid.com/blog/why-the-notification-is-the-most-important-email-you-can-send-for-growth/>
- Sleeper, B. (23 de Febrero de 2018). *8 Rules of Great Email Notifications*. Obtenido de 8 Rules of Great Email Notifications: <https://www.sparkpost.com/blog/5-rules-email-notifications/>
- Soni, S. (30 de Abril de 2022). *Enviar CorreosEelectrónicos en PHP Usando Swift Mailer*. Obtenido de Enviar CorreosEelectrónicos en PHP Usando Swift Mailer: <https://code.tutsplus.com/es/tutorials/send-emails-in-php-using-the-swift-mailer--cms-31218>
- The GraphQL Foundation. (s.f.). *Introduction to GraphQL*. Obtenido de Introduction to GraphQL: <https://graphql.org/>
- The PostgreSQL Global Development Group. (s.f.). *PostgreSQL About*. Obtenido de PostgreSQL About: <https://www.postgresql.org/about/>

Universidad Austral de Chile. (Abril de 2016). *Procedimiento mesa de ayuda*. Obtenido de Procedimiento mesa de ayuda: https://www.uach.cl/uach/_file/procedimiento-mesa-de-ayuda-v22-5729decab18fc.pdf

Webkul Software . (2023). *UVdesk*. Obtenido de UVdesk: <https://www.uvdesk.com/es/>

Zammad. (2023). *Zammad*. Obtenido de Zammad: <https://docs.zammad.org/en/latest/about/zammad.html>

Zoho Corporation. (2023). *Zoho Mail*. Obtenido de Zoho Mail: <https://www.zoho.com/es-xl/mail/features.html?src=pd-menu>

8 ANEXOS

8.1 Anexo A: Revisión sistemática

Estudio de literatura asociada a la implementación de mesas de ayuda o helpdesk

Jonathan González B.
Estudiante Universidad Austral de Chile

Introducción

Para comprender en mayor medida la definición de una mesa de ayuda y las necesidades que se deben cubrir con su implementación, los principales desafíos y metas, se realizará una revisión sistemática.

Una revisión sistemática es un medio de evaluación e interpretación de toda la investigación pertinente disponible para una pregunta de investigación particular. Estas tienen como objetivo presentar una evaluación razonable de un tema de investigación mediante el uso de una metodología fiable, rigurosa y verificable.

Metodología

En este estudio se utilizó la metodología de revisión sistemática. Fueron consideradas publicaciones asociadas al tema y que fueron publicadas a partir del año 2000. Este proceso fue llevado a cabo durante la tercera semana del mes de Noviembre, 2021.

Objetivo y pregunta de revisión

La mesa de ayuda de TrAC-FID tiene por objetivo centralizar y organizar todas las solicitudes de soporte de sus usuarios. Estas necesidades pueden ser de distintas índoles, por lo que la mesa de ayuda a implementar debe permitir la entrada de todas estas necesidades. Para conocer cuáles son estas necesidades y funcionalidades que toda mesa de ayuda debe cumplir se define la siguiente pregunta de investigación:

- **¿Cuáles son las principales consideraciones a la hora de poner en marcha una mesa de ayuda?**

Búsqueda

Para realizar la búsqueda de artículos e información asociada a la respuesta de la pregunta de revisión se utilizó la librería digital ACM (<http://portal.acm.org>). En una primera instancia se revisaron los resultados obtenidos en la librería digital de *Web of Science* (<https://www.webofscience.com/>) pero las búsquedas no satisfacían la pregunta de revisión.

Para la organización, definición de criterios de aceptación y análisis de los artículos seleccionados se utilizó la plataforma web *Parsifal* (<https://parsif.al/>).

Antes de definir el string de búsqueda, se analizaron las palabras que arrojaban mayores cantidades de resultados asociados al tema, realizando diferentes combinaciones de estas. Finalmente se optó por la cadena de string que generaba resultados más asociados a la pregunta. Además, estas búsquedas se realizaron en inglés dado que la mayoría de los documentos de la librería se encuentran en este idioma.

Las palabras claves identificadas, a partir de la pregunta de revisión fueron las siguientes:

Helpdesk, Service desk, implementation, development, framework, integration, functions, benefits, support tickets.

Estas palabras claves fueron clasificadas en 3 grupos:

- Help Desk, Service Desk, support tickets.
- Functions, benefits.
- Implementation, Development, Framework, Integration.

Selección de estudios y asignación de calidad

El string de búsqueda generó 45 resultados, estos 45 resultados fueron filtrados en base a los criterios antes definidos.

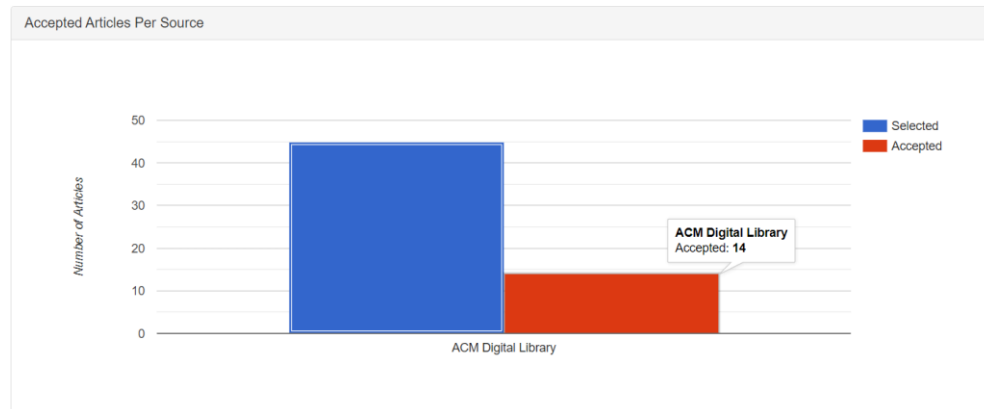


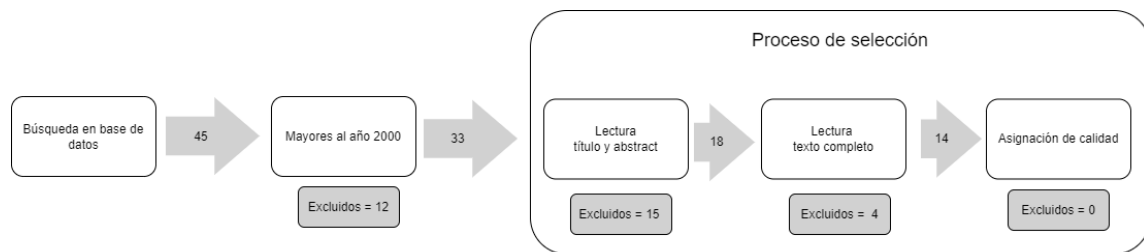
Figura 1. Total, de artículos aceptados del total de seleccionados

Al final se obtuvo un total de 14 artículos aceptados.



Figura 2. Número de artículos por año

Los artículos fueron seleccionados siguiendo tres etapas: (1) Lectura de títulos y abstract o resumen, (2) Lectura del texto completo y (3) Asignación de calidad.



En las primeras dos etapas, la relevancia de los estudios fue evaluada aplicando los siguientes criterios de exclusión o inclusión:

Criterios de selección

A partir de los resultados de búsqueda se aceptaron aquellos artículos que responden de alguna forma a la pregunta de revisión, ya sea porque presentan la implementación de una mesa de ayuda, explican funcionalidades, consideraciones o imprevistos. Además, se tuvo consideración con aquellos que presentaban un framework o tecnologías asociadas a la implementación de un help desk o mesa de ayuda. Por otra parte, se descartaron aquellos artículos menores al año 2000.

Criterios de inclusión

- Responde a la pregunta.
- Es actual (>2000)
- Presenta un framework de trabajo.
- Presenta el desarrollo de una mesa de ayuda.

Criterios de exclusión

- No es actual
- No disponible
- No presenta un abstract o resumen
- No responde a la pregunta

El número de artículos excluidos por cada criterio se pueden ver en la siguiente Tabla 2.

Tabla 2. Número de artículos excluido por criterio

Razón de exclusión	Cantidad
No es actual	12
No disponible	0
No presenta abstract o resumen	0
No responde a la pregunta	19
Total	31

En la tercera etapa del proceso fue asignado un puntaje de calidad a cada uno de los 14 estudios no excluidos en etapas anteriores. Este puntaje se asigna en base a 5 preguntas que se detallan en la Tabla 3.

El puntaje final de cada artículo se obtiene asignando un valor asociado al nivel en que responden a cada pregunta, este puntaje puede tomar 3 valores: cero si no responde a la pregunta, 0.5 si responde parcialmente a la pregunta y 1 si responde completamente a la pregunta o criterio. Se considera una buena calidad los puntajes iguales o mayores a 3 (60% de calidad), por lo que puntajes menores a este serán excluidos, la siguiente Tabla 4 resume los puntajes obtenidos por cada artículo.

Tabla 3. Rúbrica de evaluación de calidad de artículos

ID	Pregunta	Yes -1.0	Partially -0.5	No -0.0
PA1	¿Indica las funcionalidades más importantes de una mesa de ayuda?	Presenta o describe las funcionalidades de una mesa de ayuda	Presenta las funcionalidades, pero de manera implícita.	El artículo no se refiere a las funcionalidades de la mesa de ayuda.
PA2	¿Explica cómo se implementa una mesa de ayuda?	Presenta las consideraciones a tener en cuenta a la hora de implementar una mesa de ayuda.	Presenta la implementación de una mesa de ayuda de manera parcial o no directa.	No presenta la implementación de una mesa de ayuda.
PA3	¿Presenta algún framework para el desarrollo de mesas de ayuda?	Presenta uno o más frameworks para la implementación de mesas de ayuda.	Presenta un framework de implementación, pero no	No presenta un framework de trabajo.
PA4	¿Presenta el desarrollo de una mesa de ayuda?	Presenta el desarrollo de una mesa de ayuda	Presenta parcialmente el desarrollo de una mesa de ayuda	No presenta el desarrollo de una mesa de ayuda.
PA5	¿Presenta una experiencia sobre el uso de una mesa de ayuda?	Existe un contexto real en el cual está basado el artículo.	El contexto no es claro.	No existe un contexto o experiencia de implementación.

Tabla 4. Resultados de asignación de calidad

ID	Título	PA1	PA2	PA3	PA4	PA5	Puntaje
A01	Knowledge Integration with a 24-Hour Help Desk	1	0.5	0	0.5	1	3
A02	Evolution of Contact Point: A Case Study of a Help Desk and Its Users	1	0.5	0	0.5	1	3
A03	Track-It! Help Desk Automation	1	1	1	0	0.5	3.5
A04	Transforming a Help Desk from Average to Excellent	1	0.5	0	0.5	1	3
A05	Behind the Help Desk: Evolution of a Knowledge Management System in a Large Organization	1	0.5	0	0.5	1	3
A06	Building an IT Help Desk: From Zero to Hero	1	0.5	0	1	1	3.5
A07	Internal and External Communication and Collaboration: Building a Strong Help Desk Environment	1	0.5	0	0.5	1	3
A08	Winning All around, Providing a Help Desk Service Point at the Learning Commons	1	0.5	0	0.5	1	3
A09	Collaborating Outsourcing of the IT Help Desk	1	0.5	0	0.5	1	3

A10	Read All about It! Help Desk Newsletter Informs and Enlightens Organization	1	0.5	0	1	1	3.5
A11	Continuous Change: A Help Desk Motto	1	0.5	0	0.5	1	3
A12	Effective Zero-Cost Help Desk Software	1	0.5	1	0.5	1	4
A13	Getting the Most out of Your Help Desk Software	1	1	1	1	1	5
A14	Leveraging the 24x7 Operations Center to Extend Help Desk Service Hours	1	0.5	0	0.5	1	3

Ningún estudio evaluado tuvo un puntaje menor a 3, por lo tanto, todos los artículos tienen un valor aceptable de calidad

Extracción de información

Luego de filtrados los artículos, se analizaron aquellos que fueron aceptados. Para realizar esto fue necesario leer los artículos y extraer la información que resulta relevante para responder la pregunta de revisión. Para extraer esta información se tuvieron en consideración los siguientes tópicos:

- **Fecha.** Año en que se publicó el artículo
- **Experiencia.** Caso de estudio o contexto sobre el cual se basa el artículo. Ej: Help desk Universidad Austral de Chile.
- **Consejos.** Consideraciones que el autor del artículo presenta a la hora de implementar, desarrollar, o mejorar una mesa de ayuda, de manera explícita o implícita.
- **Puntos clave.** Contenido clave dentro del artículo, que permiten resaltar los puntos más importantes dentro del documento.

Resultados

De cada artículo aceptado se extrajo la información relevante (en base a criterios definidos), esta información fue analizada y guardada en una tabla.

Tabla 5. Extracto de la información extraída por artículo

article	Fecha	Experiencia	Consejos	Puntos clave
Knowledge Integration with a 24-Hour Help Desk	2000	The University of Pittsburgh's Computing Services and Systems Development organization	<p>Customers will benefit from the availability of online technical support resources.</p> <p>The provision of IT support via personalized one-on-one consulting versus the provision of self support tools is a critical decision that all IT support organizations must make, and frequently revisit, to ensure that all customers are being served appropriately.</p>	<p>The goal of a centralized Help Desk is to improve customer service by establishing a single point of contact, a virtual front door, for information technology services provided by our group.</p> <p>24 hours per day, 7 days per week operation.</p> <p>There was a need to capture the resolution to problems and answers to common questions and make them available to Help Desk</p> <p>A recent InformationWeek Research Survey of 200 IT executives indicated that 94% view knowledge management as "very strategic" to business or IT processes.</p> <p>The knowledge base currently includes solutions to common problems for most of the popular software packages. I</p>

Effective Zero-Cost Help Desk Software	2012			<p>We would centralize all desktop support work and create one central point of contact for our entire IT group.</p>
		Colorado State University (CSU) has a highly decentralized IT Environment.	<p>Organizations further gain efficiency by being able to assign a priority to tickets and respond to tickets in a centrally defined prioritized order.</p> <p>Clients appreciate having a central point of contact, which takes the guesswork out of who to call.</p>	<p>We are able to tell that over half of our help tickets have been help desk or desktop support related so we are certainly taking advantage of this resource. The other groups, like the systems and lab groups, have benefitted from a centralized ticket tracking system, since they do not have to handle many of the “frequently asked questions”, as well as try to determine what staff member is the best equipped to handle a problem.</p>
		We tested Trellis for only two months	<p>Determine what was most desirable in a Help Desk solution. We created a list of important components: cost, ability to assign tickets, staff-only notes, searchability, and customization.</p>	<p>In general, help desk software allows IT organizations to improve customer support, become more efficient at handling customer requests, prioritize customer requests, and generate reports. It has long been recognized that help desks, and in particular help desk software, add value to an organization.</p> <p>A help desk software tool provides a standard, central</p>

				communications method between staff and clients (internal or external), and among staff.
--	--	--	--	--

Principales Hallazgos y Factores Claves

La gran mayoría de los artículos presenta experiencias de *Help desk* o mesas de ayuda dentro de Universidades. Los puntos más relevantes a la hora de implementar una mesa de ayuda son los siguientes:

- FAQ
- Integración y análisis de datos
- Contexto, horarios y personal involucrado
- Experiencia de usuarios
- Conocer las debilidades y fortalezas de la mesa de ayuda. (A08, A07)

FAQ: Una componente muy importante de las mesas de ayuda y que tiene un impacto en la eficiencia con la que se responden a las solicitudes son las preguntas frecuentes (Frequently Asked Questions). El FAQ fue creado para recopilar y difundir preguntas frecuentes (A04). Esta función permite disminuir el trabajo que debe realizar el equipo de soporte, ya que en caso de que un usuario se encuentre con un problema que otros usuarios tienen de manera repetida, podrá acceder a esta sección sin la necesidad de generar un ticket, haciendo que su solicitud o necesidad sea respondida de manera expedita. (A01) (A04)

Integración y análisis de datos: Muchos artículos destacan la importancia de la integración de datos y el análisis de estos para realizar mejoras a la mesa de ayuda, así como la definición de métricas o KPI. Por ejemplo, un FAQ se va construyendo de manera incremental y es propio de cada mesa de ayuda, para descubrir cuales son las preguntas más frecuentes es necesario ir guardando de alguna manera la información solicitada por el usuario, así luego de análisis definir cuales tienen las cualidades para ser agregadas al panel FAQ. La siguiente gráfica muestra el análisis de solicitudes realizadas a la mesa de ayuda de la Universidad de Illinois (A12)

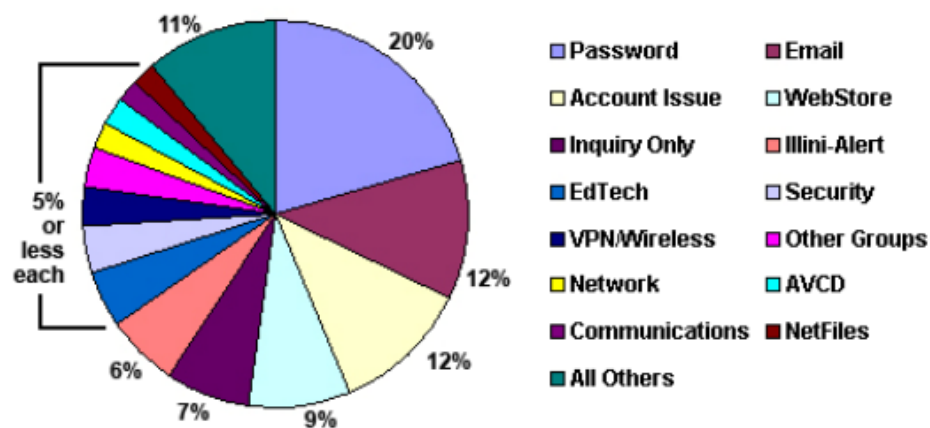


Figura 3. Analisis de solicitudes mesa de ayuda Universidad de Illinois

Este resumen de la información permite no tan solo conocer cuáles son las solicitudes más comunes sino que además puede reflejar ciertos problemas en las componentes del software, por ejemplo para los usuarios es difícil *resetear* sus contraseña, esto podría ser consecuencia de un error de diseño o un diseño visual poco amigable con el usuario. (A05)

Category	2009	2008	% chg
Password	1096	781	40%
Account Issue	669	311	115%
Email	553	424	30%
WebStore	391	336	16%
Illini-Alert	238	0	n/a
EdTech	233	86	171%
Security	211	143	48%
Other Groups	163	117	39%
Network	160	40	300%
VPN/Wireless	131	122	7%
Communications	98	110	-11%
AVCD	97	95	2%
Software	56	79	-29%
NetFiles	51	31	65%

Figura 4. Comparación solicitudes entre años 2009 y 2008, mesa de ayuda Universidad de Illinois

Un análisis comparativo como el de la figura 4 permite conocer el comportamiento temporal de las solicitudes, saber si una característica fue bien recibida o es muy difícil de comprender por los usuarios. Además, las solicitudes pueden agruparse por la cantidad de recursos que se utilizan para darles respuesta. (A14). Las organizaciones ganan aún más en eficiencia al poder asignar una prioridad a los tickets y responder a los tickets en un orden centralizado y ordenado (A02)

Además, una correcta integración y análisis de datos presenta varios beneficios(A13):

1. Proporciona al personal de la mesa de ayuda la información necesaria para ayudar a los clientes.
2. Aumenta el número de resoluciones de primera llamada.
3. Reduce los niveles de estrés.
4. Mejora la credibilidad de su departamento de TI.
5. Desarrolla el compromiso de apoyo y la aceptación.
6. Identifica el proceso de comunicación y apoyo.
7. en la delantera.

8. Libera tiempo para técnicos de segundo y tercer nivel
9. Apoyar al personal para que se concentre en la resolución de problemas, la investigación,
10. planificación y prueba de nuevas tecnologías.
11. Proporciona una voz coherente entre el cliente y el soporte técnico.

Contexto, horarios y personal involucrado: Una mesa de ayuda debe ser capaz de responder las 24 horas del día y los siete días de la semana, este es un punto que se remarca en varios artículos. (A08,A01,A09,A05). Esta capacidad de responder de manera continua debe alinearse con las capacidades del software y la disponibilidad del personal de soporte. Se debe asegurar que el cliente tenga la misma experiencia al contactarse con la mesa de ayuda a cualquier hora del día o la noche (A05). Por esta razón el sistema debe estar funcionando de manera continua, lo que implica la generación de un protocolo para el control de incidentes no planeados (A08).

Conocer las debilidades y fortalezas de la mesa de ayuda: Para ofrecer un buen servicio es necesario conocer cuáles son las capacidades y limitaciones de la mesa de ayuda, esto con el fin de no ofrecer funcionalidades que no se puedan cumplir. Una mesa de ayuda no puede mejorar a menos que conozca sus fortalezas y debilidades. Los usuarios pueden proporcionar mejor esta información y una encuesta es la forma más eficaz de recopilar los datos. (A06). Un ejemplo de encuesta de satisfacción aborda las siguientes preguntas (A06):

1. ¿Cómo calificaría usted la puntualidad de la respuesta a su problema o pregunta? (sobresaliente, por encima de las expectativas, como se esperaba, por debajo expectativas, inaceptable),
2. ¿La respuesta que recibió resolver su problema o responder a su pregunta por completo? (completamente y anticipado otras preguntas, completamente, suficientemente, no completamente, en absoluto).
3. ¿Fue el tono de la respuesta que recibió del servicio al cliente: (extraordinariamente servicial, atento, cortés, desinteresado, grosero).
4. ¿Cuándo fue el trabajo completado? (durante la visita inicial, durante múltiples 233230 visitas entre las cuales me mantuvieron informado, durante múltiples visitas entre las cuales NO me mantuvieron informado)
5. ¿Cualquier otra comentarios o necesidades que cree que no se están cumpliendo (pero que deberían reunir) Servicios de Computación? (caja de comentarios).

Lista de estudios seleccionados para la revisión sistemática

- [A01] Knowledge Integration with a 24-Hour Help Desk
- [A02] Effective Zero-Cost Help Desk Software
- [A03] Building an IT Help Desk: From Zero to Hero
- [A04] Behind the Help Desk: Evolution of a Knowledge Management System in a Large Organization
- [A05] Collaborating Outsourcing of the IT Help Desk
- [A06] Transforming a Help Desk from Average to Excellent
- [A07] Getting the Most out of Your Help Desk Software
- [A08] Leveraging the 24x7 Operations Center to Extend Help Desk Service Hours
- [A09] Winning All around, Providing a Help Desk Service Point at the Learning Commons
- [A10] Track-It! Help Desk Automation
- [A11] Continuous Change: A Help Desk Motto
- [A12] Read All about It! Help Desk Newsletter Informs and Enlightens Organization
- [A13] Internal and External Communication and Collaboration: Building a Strong Help Desk Environment
- [A14] Evolution of Contact Point: A Case Study of a Help Desk and Its Users

8.2 Anexo B: Toma de requisitos



Informe de toma de requisitos

INFO294 - Taller de Título

13/12/2021

Estudiante: Jonathan González B.

Profesora patrocinante: Valeria Henríquez

Profesora asignatura: Eliana Scheihing

Introducción: Declaración de posicionamiento del producto

Módulo de notificaciones:

Para Directores de Escuela de la Universidad Austral de Chile, principalmente aquellos pertenecientes a carreras de formación inicial docente. **quienes** necesitan conocer tempranamente y de manera oportuna las situaciones de riesgo académico de sus estudiantes.

El módulo de Notificaciones **es** una componente de software **que** aportarán a mejorar las prestaciones del sistema TrAC, permitiendo a los Directores realizar acciones preventivas a situaciones de riesgo. **A diferencia de** la revisión periódica del sistema.

Nuestro producto permitirá a los Directores de Escuela activar y parametrizar notificaciones de las situaciones académicas irregulares, que se entregarán de manera oportuna.

Mesa de ayuda:

Para Directores de Escuela de la Universidad Austral de Chile, principalmente aquellos pertenecientes a carreras de formación inicial docente y el equipo de soporte de TrAC-FID, **quienes** necesitan de un medio centralizado para comunicar y administrar las solicitudes de soporte.

La mesa de ayuda **es** una Componentes de software **que** permitirá a los Directores comunicar sus solicitudes de ayuda y al equipo de soporte administrarlas. **A diferencia del** uso de diferentes medios para comunicar y responder a las solicitudes, cómo correo electrónico, reuniones, llamadas, entre otros.

Nuestro producto permitirá al equipo de soporte administrar de manera adecuada las solicitudes de los Directores de Escuela a través de un único medio, y a estos últimos realizar y conocer el estado de sus solicitudes.

Personajes y escenarios problemas.

Nombre	Rol
Gabriela Gutiérrez	Directora de la Escuela de Pedagogía en Educación Básica.
Renato Riquelme	Encargado de soporte

Tablas de hipótesis de personajes

Módulo de notificaciones

Nombre, rol	Gabriela Gutiérrez la Directora de Escuela de Pedagogía en Educación Básica
Lo que piensa	[Gabriela] piensa [que la revisión periódica del sistema para visualizar las fechas de los datos y buscar las situaciones irregulares le quita tiempo que podría utilizar en otras actividades]
Lo que ve	Cada vez que revisa el sistema, [Gabriela] debe [asegurarse de que los datos se encuentren actualizados]. Además [debe revisar manualmente las situaciones irregulares de los alumnos].
Lo que siente	[Gabriela] siente [incertidumbre] por tener que [revisar el sistema regularmente para saber si la información mostrada puede ser de utilidad].
Lo que hace	[Gabriela] debe [analizar las situaciones irregulares de los alumnos al principio y final de cada semestre]

Mesa de ayuda:

Nombre, rol	Gabriela Gutiérrez la Directora de Escuela de Pedagogía en Educación Básica	Renato Riquelme el Encargado del área de soporte
Lo que piensa	[Gabriela] piensa que [debería existir un único medio de contacto con el área de soporte]	[Renato] piensa que [es difícil organizar y priorizar las solicitudes]. Esto ocurre porque [llegan solicitudes por diferentes medios]
Lo que ve	Cuando [Gabriela] quiere [enviar una solicitud de ayuda] ve [que no existe un medio específico para enviar su mensaje].	[Renato] ve [mensajes de los directores llegar por distintos medios] y algunas veces [estos mensajes se pierden].
Lo que siente	[Gabriela] siente [duda] sobre [qué medio usar para comunicarse] e [incertidumbre] por [no conocer el estado de su mensaje].	[Renato] siente [frustración] por [no poder organizar las solicitudes de los usuarios]. Además, siente [desmotivación] por no poder responder el 100% de las solicitudes]
Lo que hace	Enviar solicitudes de soporte de ayuda por el medio que estime conveniente. Cuando [Gabriela] tiene [un problema que no puede resolver envía un mensaje, llama al encargado de soporte o lo menciona durante una reunión].	[Renato] revisa [periódicamente los medios de contacto con los directores], además de [tomar nota durante las reuniones sobre las solicitudes de los directores]. [Renato] anota y organiza [las solicitudes en un cuaderno].

Propuestas de valor para los escenarios problema de los personajes

Escenarios Problemas	Alternativas Actuales	Tu Propuesta de Valor
Gabriela Gutiérrez, la directora de la Escuela desea conocer las situaciones académicas actuales de sus alumnos, para así poder distinguir aquellas que son de riesgo y necesitan más ayuda, pero cómo debe realizar tantas actividades se olvida de realizar esta revisión y algunas situaciones de riesgo podrían pasarse por alto.	Gabriela, la Directora, se dirige a la plataforma TrAC-FID mediante el navegador de su computador, ingresa sus claves de acceso y luego debe revisar las situaciones irregulares existentes en el sistema.	El módulo de notificaciones enviará, en cada carga de datos nuevos, un correo con las fechas de cargas y las situaciones irregulares de los alumnos directamente al e-mail de la Directora.

Gabriela, la Directora de Escuela necesita no sabe cómo cambiar la vista de datos agrupada a individual. Por lo que debe ponerse en contacto con Renato.	Envía un correo a Renato. Gabriela debe esperar la disponibilidad de Renato, para que le confirme que el mensaje ha llegado y le informe cómo solucionar su problema.	La mesa de ayuda va a actuar cómo el único punto de contacto entre los Directores y el área de soporte. Por lo que Gabriela ingresará su solicitud de soporte directamente en la mesa de ayuda, podrá ver en tiempo real el estado de su solicitud.
Renato Riquelme debe estar pendiente de las solicitudes de soporte de los directores, además debe responder a las solicitudes que ya le han hecho, organizarlas y priorizarlas.	Renato debe estar atento al correo de soporte, su móvil y sus apuntes tomados durante las reuniones. Además, responde a las solicitudes a medida que estas llegan.	La mesa de ayuda permitirá centralizar las solicitudes, administrarlas y conocer su estado. Además, catalogará y priorizará cada solicitud.

Preguntas e hipótesis para aplicación de entrevistas

Hipótesis sobre los personajes

Personajes	Preguntas
Gabriela Gutiérrez	<ul style="list-style-type: none"> ¿Qué tal es su trabajo cómo Directora? ¿Es muy demandante? ¿Qué tal ha sido su experiencia utilizando TrAC? <p><u>Módulo de notificaciones:</u></p> <ul style="list-style-type: none"> ¿Cómo se da cuenta de las situaciones irregulares actualmente? ¿Cuánto tiempo dedica a revisar estas situaciones? ¿Le interesaría saber cuándo el sistema está actualizado? ¿Cree que sería útil el envío de una notificación cuando se realice la actualización de datos? <p><u>Mesa de ayuda:</u></p> <ul style="list-style-type: none"> ¿Qué medio utiliza para generar sus solicitudes de ayuda o soporte? ¿Piensa que esta manera es eficiente?

	<ul style="list-style-type: none"> • ¿Qué tan a menudo envía solicitudes? • ¿Siente incertidumbre respecto al estado de su solicitud de soporte? • ¿Le gustaría conocer en tiempo real el estado de sus solicitudes?
Renato Riquelme	<p><u>Mesa de ayuda:</u></p> <p>¿Debe estar pendiente de las solicitudes de los Directores? ¿Por qué medio les hacen llegar estas solicitudes?</p> <p>¿Cómo las organiza? ¿Cuáles solicitudes cumple primero?</p>

Hipótesis sobre el problema

Personajes	Preguntas
Gabriela Gutiérrez	<p><u>Módulo de notificaciones:</u></p> <ul style="list-style-type: none"> • ¿Podría darse la situación de que el alumno no se dé cuenta que se encuentra en situación de riesgo? ¿Le ha pasado? • ¿Este proceso se lleva a cabo fuera de los plazos establecidos? ¿Qué tan complicado es? • ¿Cuál es la utilidad que podría tener conocer las situaciones de riesgo a tiempo? • ¿Qué opina sobre un sistema que le alerte mediante correo electrónico la carga de nuevos datos en el sistema TrAC-FID? <p><u>Mesa de ayuda:</u></p> <ul style="list-style-type: none"> • ¿Cómo comunica sus necesidades de soporte al equipo de TrAC-FID?
Renato Riquelme	<p><u>Mesa de ayuda:</u></p> <ul style="list-style-type: none"> • ¿Cuáles son las solicitudes de ayuda más comunes? • ¿Tiene problemas para organizar y priorizar las solicitudes? • ¿Con qué periodicidad revisa su correo electrónico u otros medios de contacto? • ¿Le gustaría centralizar todas estas solicitudes en un solo lugar?

Cierre de la entrevista e información adicional

Personaje	Preguntas
Gabriela	<p><u>Módulo de notificaciones:</u></p> <ul style="list-style-type: none">• ¿Qué funcionalidades le gustaría que tenga el sistema? <p><u>Mesa de ayuda:</u></p> <ul style="list-style-type: none">• ¿Qué otras características relacionadas con este módulo le parecerían, a usted cómo usuario, relevantes?
Renato	<p><u>Mesa de ayuda:</u></p> <ul style="list-style-type: none">• ¿Qué otras características deberían tener la mesa de ayuda?

Resultado y análisis de entrevistas

Entrevista 1: Marianna Villaroel, directora y profesora de la Escuela de Informática

Módulo de notificaciones

El Director no solo revisa TrAC, sino que además revisa el Sistema Académico oficial de la UACH. El Director realiza esta actividad al comienzo y al final de cada semestre, que es donde ocurren la mayor cantidad de solicitudes. Esta revisión es necesaria para identificar situaciones irregulares ya que actualmente no existe un sistema que realice este proceso de manera automatizada, esto puede llegar a provocar en el peor de los casos que un alumno deba realizar trámites en plazos extemporáneos, esto no solo afecta el flujo de trabajo del Director, sino que además afecta a otras entidades encargadas de participar en la tramitación de estas solicitudes.

El Director o Directora espera recibir una notificación cuando se realice una actualización en los datos, esta actualización deberá incluir la fecha de la última carga de cada tipo de dato, además de recibir las situaciones académicas irregulares de los alumnos asociadas a estas cargas de nuevos datos.

Mesa de ayuda

TrAC posee un correo de soporte que es donde llegan las solicitudes que es revisado constantemente por el encargado de soporte, a pesar de esto el Director prefiere además realizar una llamada directamente al encargado.

Entrevista 2: Fabián Moraga, empleado área de soporte TrAC

Mesa de ayuda

Los usuarios le hacen llegar solicitudes por diferentes medios. Aunque la experiencia que se obtuvo durante el funcionamiento del soporte de TrAC fue bastante baja, con solo una solicitud de ayuda, la inducción al uso de la plataforma reveló una gran cantidad de dudas por parte de los usuarios.

Entrevista 3: Julio Guerra, encargado del área de soporte de TrAC

Un hallazgo relevante para el proyecto durante esta entrevista fue el hecho de que actualmente en la plataforma existe una funcionalidad llamada “Feedback” que cómo su nombre lo indica permite a los usuarios hacer llegar retroalimentación o feedback sobre

el uso de la plataforma. Podría utilizarse esta funcionalidad como parte de las funcionalidades de la mesa de ayuda, realizando las mejoras correspondientes. Además, posee un alto valor para el usuario ya que permitiría que este se comuniqué con el área de soporte directamente en la plataforma sin la necesidad de hacer uso de otra tecnología distinta a las que se utilizan en el desarrollo e implementación de TrAC-FID actualmente. Este último punto plantea un enfoque distinto al que se tenía actualmente para el desarrollo de la mesa de ayuda, ya que ahora se buscará permitir a los usuarios acceder a la mesa de ayuda directa y estrictamente en la plataforma, a diferencia del enfoque anterior que sugería la utilización de una plataforma o tecnología “semi-integrada” a la plataforma.

Además, las notificaciones tendrán un valor para incentivar el uso de la plataforma, ya que además de informar invitarán al usuario a utilizar el sistema.

Historias de usuarios

Cómo [Tipo de usuario]	Necesito [Funcionalidad]	Con la finalidad de [Motivación]	Criterios de aceptación
Director(a)	Ser alertado cuando se realicen cargas de nuevos datos.	No tener que revisar estas fechas periódicamente en el sistema.	El correo enviado informa las fechas de actualización y los tipos de datos actualizados.
			Las fechas se condicen con la información que se encuentra en la base de datos.
Director(a)	Ser alertado(a) con las situaciones académicas irregulares de mis alumnos	Poder tomar acciones preventivas.	El correo enviado informa las situaciones irregulares de los estudiantes.
			La información entregada en el e-mail permite a los usuarios acceder rápidamente a la plataforma, además explica cómo visualizar las situaciones irregulares en el sistema.
Soporte	Enviar correos electrónicos a modo de	Informar a los Directores que los datos se han actualizado y además	El sistema genera notificaciones personalizadas para cada director, considerando los

	notificaciones a los usuarios.	alertarlos sobre las situaciones de riesgo.	programas o carreras pertenecientes a su escuela.
			El sistema muestra un registro de las notificaciones enviadas
			El sistema muestra un mensaje indicando si se han realizado o no el envío de los emails.

#	Cómo [Tipo de usuario]	Necesito [Funcionalidad]	Con la finalidad de [Motivación]	Criterios de aceptación
1	Director(a)	Compartir mis sugerencias	Poder aportar a crear un sistema más amigable.	Se pueden enviar comentarios en la plataforma.
2	Director(a)	Agregar imágenes a mis sugerencias o errores encontrados en el sistema	Poder explicar mejor el contexto y contenido de mi solicitud.	Los Directores pueden adjuntar imágenes en los comentarios/sugerencias/errores reportados. El Soporte puede ver las imágenes.
3	Director(a)	Compartir el error que encuentro en el sistema.	Aportar en la mejora del sistema.	Se pueden reportar errores en la plataforma.
4	Director(a)	Editar/eliminar mi comentario.	Modificar un comentario que envié o eliminarlo.	Se puede eliminar un comentario enviado o realizado. Se puede editar un comentario enviado o realizado.
5	Director(a)	Ver la sección de preguntas frecuentes	Evitar solicitar ayuda en temas que son comunes y acceder de manera más rápida a la información que necesito.	El director puede visualizar la sección de preguntas frecuentes.
6	Soporte	Conocer la opinión/solicitud	Mejorar su experiencia en la plataforma.	El Soporte puede visualizar los comentarios realizados por los Directores.

		de los Directores sobre el sistema.		
7	Soporte	Ver las solicitudes de ayuda de los Directores.	Resolver sus dudas o prestar ayuda.	El Soporte puede visualizar las solicitudes de ayuda. El Soporte puede responder a las solicitudes de ayuda.
8	Soporte	Crear y editar Respuestas a preguntas frecuentes.	Evitar consultas repetitivas y enfocarme en las necesidades más específicas.	El Soporte puede crear una pregunta frecuente y agregar una respuesta. El Soporte puede editar o eliminar una pregunta frecuente.

Anexo Entrevistas

Entrevista 1: Marianna, profesora y directora de la Escuela de Informática, UACH

1. ¿Qué tal es su trabajo cómo Directora? ¿Es muy demandante?

R: Si, ya que debo participar semanalmente en varios tipos de reuniones, atender solicitudes de los alumnos y la Escuela, lo que trae consigo otras situaciones cómo tener que comunicarse con otros departamentos. Además, cómo profesora debo preparar y realizar clases, revisar pruebas y trabajos, es un trabajo bastante demandante.

2. ¿Cuál ha sido su experiencia al utilizar TrAC-FID?

R: Pienso que es una herramienta súper útil y la ha utilizado para analizar situaciones junto a estudiantes, pero siente que aún necesita pulir ciertos aspectos cómo aquellos relacionados con la actualización de los datos, por lo que necesitaba visualizar la información en la página de la Universidad.

Módulo de notificaciones

3. ¿Cree que sería útil el envío de una notificación cuando se realice la actualización de datos?

R: El no saber si los datos están actualizados o no me obliga a abrir la otra plataforma. Pero si comienza a notificar sería algo bueno.

4. ¿Le interesaría saber cuándo el sistema está actualizado?

R: Me encantaría, ya que si las fechas que se me notificaron indican que el sistema se encuentra actualizado en relación con la fecha actual podría saber qué puedo hacer uso de la plataforma.

5. ¿Podría darse la situación de que el alumno no se dé cuenta que se encuentra en situación de riesgo? ¿Le ha pasado?

R: Si, y ha pasado, un alumno de último año pensó que había pasado una asignatura, pero en realidad no lo pasó, “Di la prueba y me fuí de vacaciones”, lamentablemente le fue mal en la prueba y quedó eliminado. Esto provocó que tenga que pedir la reincorporación por gracia, que por suerte la tenía.

6. ¿Este proceso se lleva a cabo fuera de los plazos establecidos? ¿Qué tan complicado es?

R: Claro, se lleva a cabo de forma extemporánea, la reincorporación, la inscripción de asignaturas. Es complicado, ya que yo tengo bastante trabajo que hacer y este tipo de situaciones requiere que deba hacer los trámites necesarios de manera manual lo que me quita bastante tiempo. Además, hay otras personas que están involucradas por lo tanto no solo se ve afectado mi trabajo.

7. ¿Cree que sería útil conocer las situaciones de riesgo a tiempo?

R: Tener más tiempo para analizar las situaciones particulares, un alumno podría arriesgar sus becas cómo la gratuidad. Ya que los problemas de estas situaciones pueden ser variados, cómo situaciones personales y familiares, o le cuesta integrarse a la comunidad, no tiene compañeros para estudiar, etc. Si yo supiera antes de las situaciones podría darle soluciones o alternativas, les puedo dar orientación a los estudiantes, hay estudiantes que no saben los beneficios que tienen. Además, apoyar a los estudiantes que están rindiendo, pero podrían mejorar.

8. ¿Cómo se da cuenta de las situaciones irregulares actualmente?

R: Cuando estoy haciendo otra cosa, estoy analizando otro tipo de situaciones y al realizar estos análisis me encuentro con situaciones irregulares, cómo inscripción de asignaturas o anulación. Al analizar las situaciones y el historial me doy cuenta de situaciones anormales. Es casual, no me siento uno por uno. Los profesores a veces me avisan, “este estudiante está faltando”, “no ha venido a las pruebas”, aunque son profesores puntuales.

Además, los mismos compañeros muchas veces nos avisan de estas situaciones, y es algo he tratado de fomentar, informar situaciones irregulares.

9. ¿Cuánto tiempo dedica a revisar estas situaciones?

R: Me dedico un tiempo limitado durante cada inicio de semestre, ya que debo realizar otro tipo de actividades.

10. ¿Qué tan complicado es el proceso de analizar las situaciones de riesgo?

R: Usualmente, me reúno con el estudiante y analizamos la situación, requiere de bastante tiempo porque son más de 300 estudiantes.

11. ¿Qué funcionalidades le gustaría que tenga el sistema?

R: Poder modificar los umbrales de las notificaciones, decir qué asignaturas tienen un % definido de reprobación y conocer cuales están causando cuellos de botellas.

Que el sistema acceda y muestre los datos en tiempo real, para ver situaciones de notas, por ejemplo, y atender casos en medio de semestre.

Poder hacer un seguimiento de las situaciones.

Mesa de ayuda

12. ¿Solicitó ayuda alguna vez y cómo lo hizo?

R: Si, llamaba a Daniel o enviaba un correo.

13. ¿Qué situaciones reportaba?

R: Solicitaba ayuda, password, por ejemplo, o informaba cosas que encontraba raras en el sistema, no me aparecía un optativo, por ejemplo. Informaba errores, mensajes que no entiendo y capaz no los tenía que ver.

14. ¿Cada cuánto le respondía los mensajes?

R: Me gustaría que lo hicieran al tiro, pero usualmente al otro día, o al subsiguiente. yo creo que deberían responder a lo más dentro de un par de horas.

15. ¿Qué le parece la idea de que exista un único punto o canal de contacto?

R: Es bueno, es más fácil priorizar e ir resolviendo si hay muchos canales podrían no enterarse de algo grave que esté pasando.

16. ¿Cómo le afecta una respuesta tardía?

R: Te frena a ti, te deja sin poder avanzar en algún tema, por ejemplo, si tenía citado algún estudiante para un día y aún no me respondían un tema en el que tengo problemas y necesito que esté resuelto para atender el estudiante voy a tener que citarlo para otro día.

17. ¿Le importaría conocer el estado del mensaje?

R: Es bueno que te digan “acusó recibo” pero no creo que sea necesario ver el estado, yo al menos no creo que lo miré, no serviría de mucho.

18. ¿Cuál cree usted que serían las solicitudes más comunes?

R: Lo ideal sería no tener que hacer solicitudes, pero depende de los usuarios. Pero creo que serán cosas técnicas o de uso, o cosas que no comprenden.

19. ¿Qué funcionalidades cree que debería tener esta mesa de ayuda?

R: Que explique algunas cosas como por ejemplo que significa un puntito en una asignatura, que significa el color. Para los nuevos sería bueno saber que significan algunas cosas que se podrían olvidar fácil.

Estas dudas surgen cuando uno comienza a interactuar con el sistema.

Entrevista 2: Fabian, Encargado área de soporte

1. ¿Cuál fue su rol en el área de soporte?

R: Revisaba el correo de soporte. Además, hicimos una introducción a 10 personas sobre el uso de TrAC.

2. ¿Qué preguntas tuvieron los profesores sobre el sistema?

El usuario hace llegar la siguiente lista:

- Dos carreras con nombre tecnología médica, códigos 1732 y 1832
- Problema con las asignaturas con menciones, por ejemplo, 1816 Licenciatura en ciencias con mención tiene muchas asignaturas y los "pre" y "fluj" se marcan para todas las menciones.
- Se siente falta de información inicial para guiar a los estudiantes de primer año (por ejemplo, PSU, tipo de colegio del que egresaron). Sobre todo, porque hay profesionales que atienden mayoritariamente a este tipo de estudiantes.
- No existe una representación para las suspensiones o "congelamientos"
- Necesidad de mostrar la fecha de la última carga de datos
- Usabilidad: les costó distinguir las asignaturas seleccionadas cuando hicimos el análisis por semestre. Al parecer el resalte no es suficiente.
- Usabilidad: la funcionalidad de mouseover de la gráfica de los promedios no les parecía clara. pensaron que la leyenda tenía la información promedio del promedio
- NF (nueva funcionalidad): Agregar información de si otro profesional ya está atendiendo a un estudiante. Si esto se fuera a hacer mi propuesta sería que basado en los logs se detallaran los últimos usuarios que vieron la información del estudiante.
- NF: agregar información de otras unidades de atención, por ejemplo, si están siendo atendidos por el CESU, becas, etc.
- NF: botón de imprimir, la opción de imprimir del navegador no permite ver las características de los cursos. Se puede ver lo avanzado en TrAC-FID respecto a esto.
- NF: ver notas parciales (sé que no se puede, pero dejo registro)
- Había confusión con los colores en algunos ramos, por ejemplo, curso con 4,2 (fondo verde) pero reprobado (círculo rojo). Parece ser algo recurrente en las carreras de la salud.

Esos son todos los comentarios que rescatamos de la introducción que se le hizo a la unidad de apoyo al aprendizaje

3. ¿Cuántas solicitudes le llegaron?

R: Solo nos llegó un correo y era una profesora de informática.

4. ¿A qué se debe esto?

R: El sistema era nuevo y poco utilizado.

5. ¿Qué medios utilizaban para realizar el soporte?

R: Solo el correo.

6. ¿Con qué periodicidad se revisaba el correo?

R: Lo ideal es revisarlo todos los días. Día por medio, ya que esperaba que no hubiera mucha actividad.

7. ¿Funcionalidades para la mesa de ayuda?

Preguntas frecuentes

8. ¿Qué tan activo fue el uso de TrAC?

R: En un comienzo lo utilizaron bastante, pero luego de un tiempo cómo dos o tres.

9. ¿Cuál es el problema con la integración de TrAC?

R: Les cuesta introducir la aplicación a sus actividades diarias, interiorizarla.

Entrevista 4: Profesor Julio Guerra, encargado área de soporte

Mesa de ayuda

1. ¿Solicitudes más comunes?

R: El tema fundamental va a ser la incongruencia de los datos, algún dato no se podría explicar y necesitaría ayuda para saber si es un error de datos.

Que no aparezcan los electivos, por ejemplo.

2. ¿Qué se debe tener en cuenta para ofrecer un buen servicio en la mesa de ayuda?

R: Contestar rápido, además de revisar, tratar de responder las solicitudes lo más rápido posible, estar atento. Incentivar a los usuarios a usar el sistema para mejorar el sistema.

3. ¿Qué impacto tendrá la mesa de ayuda en el desarrollo de TrAC-FID?

R: Se incentiva a los usuarios a dejar comentarios, esto hace que se pueda mejorar la herramienta.

4. ¿Problemas de no tener una mesa de ayuda?

R: Uno anota de repente cosas por ahí y se podrían perder.

5. ¿Qué le parece un sistema de tickets?

R: Si son muchas personas me parece buena idea, pero se podría transformar en otra aplicación que mantener y actualizar, podría significar una carga para los desarrolladores o encargas de la implementación de la mesa de ayuda. Tal vez con una población tan baja cómo la que tendremos podría ser innecesario, pero si se llegara a implementar con más estudiantes, ahí sería necesario. Por ahora con que alguien revise el correo electrónico podría ser suficiente.

6. ¿Qué características debería tener la mesa de ayuda?

R: Pensando en que muchas personas utilicen TrAC-FID, me parece buena idea unificar la mesa de ayuda. Debe existir un único punto de contacto con los usuarios. Si en un futuro se escala en cantidad de usuarios, sería bueno tener funcionalidades más inteligentes que generen FAQ automáticos en base a preguntas más comunes o que organicen estás solicitudes por tema.

Una buena idea sería facilitar que los usuarios puedan mandar feedback, tal vez automatizar capturas de pantalla, el feedback podría salir de la misma aplicación. Actualmente existe un sistema de feedback. El correo podría enviarse automáticamente.

Dado el bajo uso que podría tener el sistema en el peor de los casos se podrían tener 10-15 tickets.