

Smart Agencies

Alex Álvarez Gallardo

Ingeniería Informática

Desarrollo multiplataforma de aplicaciones móviles

Carlos Sanchez Rosa

Carles Garrigues Olivella

07/01/2024



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivad a [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

A) Creative Commons:



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivad a [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-Compartirlgual [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-Compartirlgual [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento [3.0 España de Creative Commons](#)

B) GNU Free Documentation License (GNU FDL)

Copyright © AÑO TU-NOMBRE.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3

or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

C) Copyright

© (el autor/a)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Smart Agencies</i>
Nombre del autor:	<i>Alex Álvarez Gallardo</i>
Nombre del consultor/a:	<i>Carlos Sanchez Rosa Jordi Almirall López</i>
Nombre del PRA:	<i>Carles Garrigues Olivella</i>
Fecha de entrega (mm/aaaa):	01/2024
Titulación:	<i>Ingeniería Informática</i>
Área del Trabajo Final:	<i>Desarrollo multiplataforma de aplicaciones móviles</i>
Idioma del trabajo:	Castellano
Palabras clave	<i>Agencies, Scrum y Azure</i>
<p>Resumen del trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.</i></p>	
<p>Este proyecto se ha llevado a cabo con la finalidad de desarrollar una aplicación integral para la gestión de agencias, respondiendo a las necesidades específicas del cliente. Adoptando la metodología ágil Scrum, el equipo se comprometió a realizar entregas incrementales y garantizar la adaptabilidad a cambios en el desarrollo del proyecto.</p> <p>La aplicación, construida utilizando tecnologías como React Native y ASP.NET Core con Entity Framework Core, se ha diseñado para optimizar la gestión de agencias mediante un enfoque ágil y moderno. La estrategia de dividir el proyecto en bloques manejables ha facilitado el control y la priorización de la calidad en cada etapa del desarrollo.</p> <p>En la planificación del trabajo, se destacó la utilización de un diagrama de Gantt para una visión detallada de las tareas y sus estados.</p>	

El enfoque ágil y la identificación proactiva de riesgos en la planificación han sido fundamentales. Se ha creado un backlog en Azure DevOps para futuros sprints, delineando tareas estratégicas para la evolución tecnológica del producto. Sin embargo, se eliminó la funcionalidad de inicio de sesión debido a la gestión interna del cliente.

Este proyecto concluye con éxito al ofrecer un producto que supera las expectativas del cliente en la gestión de agencias. A pesar de los desafíos, la transparencia en la gestión de riesgos y una planificación cuidadosa garantizan el éxito del producto. La estrecha colaboración con el cliente y la preparación para futuras mejoras evidencian un enfoque estratégico y sostenible en el desarrollo de software.

Abstract (in English, 250 words or less):

This project has been carried out with the aim of developing a comprehensive application for agency management, addressing the specific needs of the client. Embracing the agile Scrum methodology, the team committed to delivering incrementally and ensuring adaptability to changes in the project's development.

The application, built using technologies such as React Native and ASP.NET Core with Entity Framework Core, has been designed to optimize agency management through an agile and modern approach. The strategy of breaking down the project into manageable blocks facilitated control and prioritization of quality at each stage of development.

In work planning, the use of a Gantt chart was emphasized for a detailed view of tasks and their statuses.

The agile approach and proactive risk identification in planning have been crucial. A backlog has been created in Azure DevOps for future sprints, outlining strategic tasks for the technological evolution of the product. However, the login functionality was removed due to internal client management.

This project concludes successfully by delivering a product that exceeds client expectations in agency management. Despite challenges, transparency in risk management and careful planning ensure the product's success. Close collaboration with the client and readiness for future improvements demonstrate a strategic and sustainable approach to software development.

Índice

1. Introducción.....	9
1.1 Contexto y justificación del Trabajo.....	9
1.2 Objetivos del Trabajo.....	9
1.3 Enfoque y método seguido.....	11
1.4 Planificación del Trabajo.....	11
1.5 Breve resumen de productos obtenidos.....	12
1.6 Breve descripción de los otros capítulos de la memoria.....	12
2. Diseño Centrado en el Usuario (DCU).....	13
2.1 Justificación de la elección de métodos de indagación.....	13
2.2 Fichas de perfil de usuario.....	19
2.3 Point of view statements.....	21
2.4 Prototipo.....	22
2.4.1 Capturas de pantalla.....	22
2.4.2 Enlace al prototipo interactivo.....	24
2.5 Resultados del test con usuarios.....	25
2.6 Puntos de mejora detectados.....	29
3. Diseño técnico (DT).....	30
3.1 Definición de los casos de uso.....	30
3.1.1 Diagrama UML forma gráfica actores.....	30
3.1.2 Lista de los casos.....	31
3.2 Diseño de la arquitectura.....	33
3.2.1 Diagrama UML diseño de la base de datos.....	33
3.2.2 Diagrama UML diseño de las entidades y clases.....	34
3.2.3 Diagrama arquitectura del sistema.....	35
3.2.3.1 Diagrama Backend.....	35
3.2.3.2 Diagrama Frontend.....	36
3.2.3.3 Comunicación entre frontend y backend.....	37
4. Implementación.....	37
4.1 Desarrollo.....	37
4.1.1 Herramientas utilizadas en el desarrollo de la aplicación.....	37
4.1.2 Estado del proyecto.....	39
4.2 Pruebas sobre la aplicación.....	41
3. Conclusiones.....	43
3.1 Lecciones aprendidas a lo largo del trabajo.....	43
3.2 Reflexión sobre los objetivos planteados inicialmente.....	43
3.3 Reflexión sobre el seguimiento de la planificación y la metodología.....	44
3.4 Futuras líneas de trabajo.....	45
4. Glosario.....	46
5. Bibliografía.....	47
6. Anexos.....	48
Anexo I : Flujos de interacción DCU.....	48
Anexo II: Sketches escaneados DCU.....	51

Anexo III: Explicaciones con soluciones DCU.....	53
Anexo IV: Documento de preguntas DCU.....	53
Anexo V: Encuesta y resultados sobre el MVP.....	54

Lista de figuras

No se encuentran elementos de tabla de ilustraciones.

1. Introducción

1.1 Contexto y justificación del Trabajo

Una gran cadena hotelera ha contactado con nosotros para dar un paso al siguiente nivel, apostando por pasar su software interno a una aplicación móvil con el fin de actualizarse tecnológicamente y sobre todo para seguir gestionando la empresa desde cualquier lugar.

Al ser un producto totalmente propio de la casa, no tenemos el conocimiento suficiente si a nivel de negocio otras empresas utilizan algo similar, así que tendremos que solicitar parte del código sobre la versión desktop para entender los flujos y el funcionamiento real de la aplicación. Nos han comentado que tienen bastantes problemas de rendimiento por lo que también será un punto a tener en cuenta que la aplicación sea escalable.

1.2 Objetivos del Trabajo

- Cumplir las necesidades y expectativas del cliente.
- Respetar los tiempos acordados.
- Desarrollar una aplicación con React Native para el frontend.[2][3]
- Desarrollar una aplicación con ASP.NET Core para el backend con el uso de Entity Framework Core. [2][3]
- Utilizar Azure DevOps como herramienta para Scrum. [1]
- Usar un sistema cloud como AWS (Amazon Web Services) para desplegar la API (Application Programming Interface) y almacenar la base de datos.
- Utilizar como base de datos SQL(Structured Query Language).
- Generar un MVP (Minimum Viable Product) con las siguientes características:
 - Tendremos que loguearnos en la aplicación con las credenciales proporcionadas por la empresa.
 - Al acceder a la aplicación podremos filtrar por agencias por los siguientes campos:
 - Nombre de la agencia
 - Id de la agencia
 - Fecha de alta de la agencia
 - Ámbito de la agencia
 - Para acceder al detalle de la agencia deberemos de hacer doble click en la agencia para poder ver los datos. Dentro del detalle podremos:
 - Modificar los datos de la agencia
 - Dar de baja la agencia
 - Guardar la agencia

- En el detalle de la agencia encontraremos los siguientes datos:
 - Nombre de la agencia
 - Id de la agencia → generado automáticamente por el sistema
 - Fecha de alta de la agencia → generado automáticamente por el sistema al crear la agencia
 - Ámbito de la agencia → Cliente o Proveedor
 - Mercado → mercado al que pertenece la agencia
 - Tipo de agencia → B2C o B2B
- También encontraremos dos pestañas adicionales que son las siguientes:
 - Pestaña de datos comerciales ⁽¹⁾ :
 - Calle
 - Número
 - País Comercial
 - Comunidad
 - CP
 - Población
 - Teléfono
 - Pestaña de datos fiscales ⁽¹⁾ :
 - País facturación
 - CP
 - Población
 - Comunidad
 - Provincia
 - Calle
 - Número
 - Email
 - Teléfono
 - CIF/NIF
 - Nombre Fiscal
 - Fax

(1) Después de hablar con el cliente, nos ha notificado que estos campos no deben de tener ningún tipo de validación, los usuarios de la aplicación deben de ser los responsables de los datos que rellenan. En todo caso, se podrá tratar como una propuesta de mejora para desarrollar el producto.

1.3 Enfoque y método seguido

El equipo de gestión del proyecto ha decidido utilizar una metodología ágil como es Scrum para poder desarrollar el producto solicitado. Se podrá hacer una adaptación del código que se usa en la API y migrar a una tecnología más actualizada. Si que tendremos que desarrollar un producto nuevo en cuanto al framework ya que hasta el momento estaban trabajando con una tecnología con poca demanda.

Hemos elegido esta estrategia ya que en proyectos más grandes y complejos esta metodología se adapta a los diferentes cambios que puedan ocurrir en el proyecto, siempre hay una entrega y mejora continua con el fin de entregar en los diferentes plazos código funcional del producto y es esencial para mantenerse en el camino correcto y los intereses del cliente. Esto va ligado al dividir el proyecto en pequeños bloques, podemos mantener un mayor control sobre el proceso de desarrollo y asegurarnos de que la calidad sea una prioridad en cada etapa.

En resumen, nuestro enfoque se basa en la agilidad, la adaptación y la entrega incremental. Estamos comprometidos a mantener la calidad y el control a medida que avanzamos en este proyecto grande y complejo. Scrum nos brinda las herramientas y la estructura necesarias para alcanzar nuestros objetivos y satisfacer las necesidades cambiantes de nuestros interesados. Estamos emocionados por los desafíos que se presentan y confiamos en que este enfoque nos llevará al éxito en la entrega del producto solicitado.

1.4 Planificación del Trabajo

El diagrama de Gantt se ha adjuntado y se puede encontrar en la carpeta con el nombre de "*Diagrama de Gantt.pdf*" ya que si se insertaba se perdía la resolución.

Dentro de este diagrama podemos ver la planificación temporal para cada una de las tareas que componen el desarrollo de este producto.

Podemos observar que cada una de las tareas se encuentra desglosada en hitos parciales con su respectiva fecha de inicio, fecha final, su duración en días, su estimación en horas laborables y el estado de cada uno de los hitos para poder hacer un mejor seguimiento.

La tarifa ha sido negociada por el CIO (Chief Information Officer) de nuestra empresa, por lo que no hemos podido negociar directamente con el cliente. Esta tarifa es cerrada y muy difícilmente podrá ser modificada, aproximadamente se cobrará a 40€/h. Como hemos visto en el diagrama de Gantt se ha hecho una estimación total de 824 horas, por lo que nos daría un total de 32.960€, sin tener en cuenta los gastos que se han generado para la empresa, como puede ser nóminas, licencias, etc.

Como plan para poder mitigar posibles riesgos encontramos los siguientes escenarios sobre los que habría que actuar:

- Alcance cambiante por parte del cliente: que el cliente no tenga claro los objetivos que se definieron en su momento y quiera cambiar ciertos requisitos supondría un retraso en la implementación.
- Riesgos tecnológicos: hemos tomado un riesgo claro que es el desconocimiento de las tecnologías React Native y AWS, pero la intención es dar la mejor calidad posible al cliente. [6]
- Recursos insuficientes: falta de recursos para desarrollar todo el software, en este caso, hay falta de personal.

1.5 Breve resumen de productos obtenidos

- Backlog para el seguimiento del desarrollo del producto.
- Base de datos SQL donde se almacenarán los datos que se han ido creando o modificando en la aplicación.
- Frontend en React Native donde los usuarios interactúan con la aplicación.[2][3]
- Backend en ASP.NET Core quién se encargará de toda la lógica y el modelo de negocio. [5]
- Memoria de trabajo donde se explica todo el proceso que se ha hecho para realizar el desarrollo del software.
- Presentación que se hará al cliente para que pueda revisar el producto final y dar feedback de su opinión.

1.6 Breve descripción de los otros capítulos de la memoria

- Metodología de trabajo en Azure DevOps, podremos encontrar explicado la metodología de trabajo que se va a llevar a cabo, priorización de tareas, seguimiento, etc.[1]
- Arquitectura de la base de datos y razones de su uso, podremos ver cómo se implementó y su estructura.
- Arquitectura del frontend, tecnologías utilizadas e implementación.
- Arquitectura del backend, tecnologías utilizadas e implementación. [5]
- Presentación, explicación del producto final generado y feedback del cliente.

2. Diseño Centrado en el Usuario (DCU)

2.1 Justificación de la elección de métodos de indagación

Elección de los métodos de indagación

- Encuestas → obtendremos información cuantitativa sobre un grupo de usuarios. Realizaremos una encuesta online a un grupo de usuarios potenciales. Esta encuesta nos permitirá obtener información sobre características demográficas, hábitos de uso y preferencias de los usuarios.

[Google Forms utilizado para las encuestas.](#)

- Entrevistas → obtendremos información cualitativa sobre las experiencias y opiniones de los usuarios. Para este proyecto, aprovecharemos los usuarios que están utilizando la aplicación de escritorio del cliente, aquí tendremos un muy buen sitio para poder sacar feedback ya que son usuarios con cierta experiencia. Estas entrevistas nos permitirán conocer sus necesidades y objetivos específicos, así como sus preferencias y expectativas.

[Plantilla usada para las entrevistas](#)

Desarrollo

La cadena hotelera nos ha proporcionado unos cuantos usuarios de la aplicación para la toma de información, por lo que esa parte de buscar a las personas ya ha sido hecha por el cliente. Han decidido elegir un grupo de personas con diferentes perfiles para facilitarnos el trabajo y poder obtener las diferentes opiniones de cada uno de estos perfiles.

Resultados

Una vez realizada las encuestas y las entrevistas tenemos que analizar los resultados.

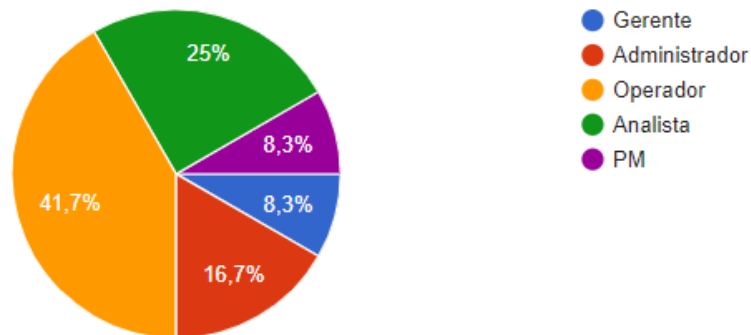
- Encuestas

Para las encuestas que hemos realizado a los usuarios, hemos obtenido los siguientes resultados.

Perfiles de usuario

¿Cuál es tu rol en la empresa?

12 respuestas

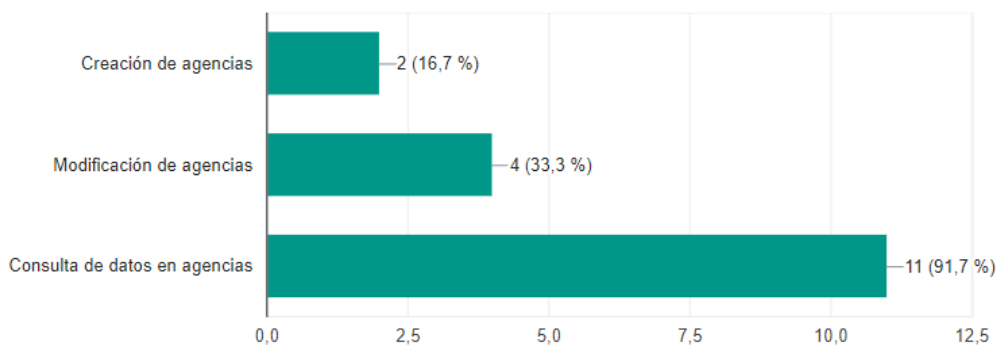


Perfiles de usuario

¿Qué tipo de tareas realizas con Smart Agencies Desktop?

12 respuestas

 Copiar

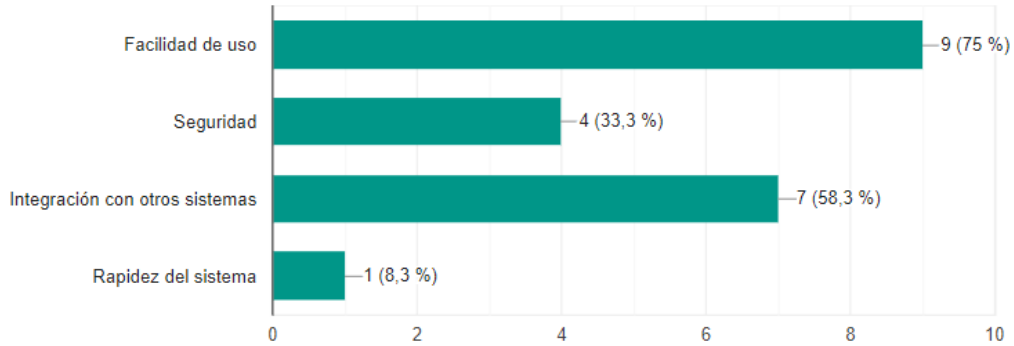


Necesidades y objetivos

[Copiar](#)

¿Qué características o funcionalidades son importantes para ti en una aplicación de gestión de agencias?

12 respuestas

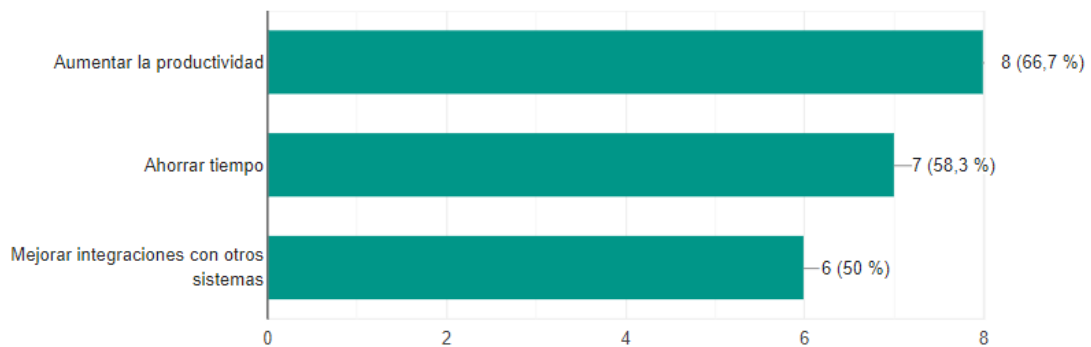


Necesidades y objetivos

[Copiar](#)

¿Cómo crees que una aplicación de Smart Agencies podría ayudarte a mejorar tu trabajo?

12 respuestas

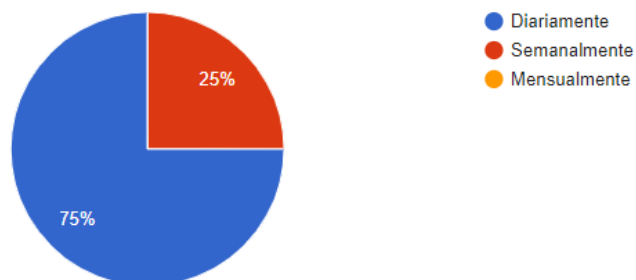


Contexto de uso

[Copiar](#)

¿Con qué frecuencia utilizas Smart Agencies Desktop?

12 respuestas

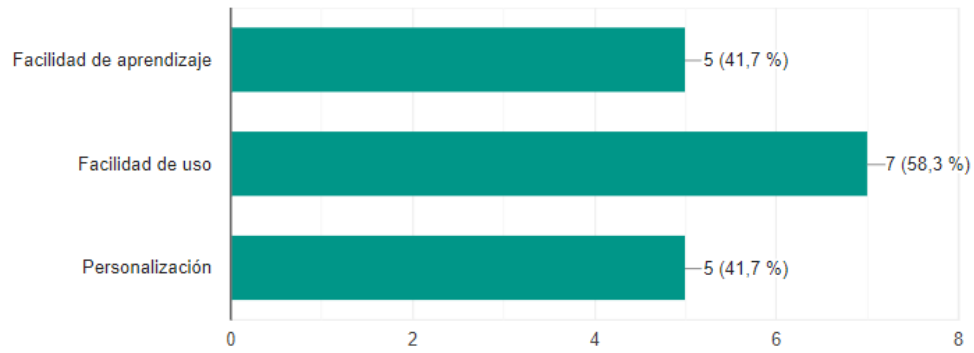


Usabilidad

 Copiar

¿Qué aspectos de la usabilidad de una aplicación de Smart Agencies son importantes para ti?

12 respuestas

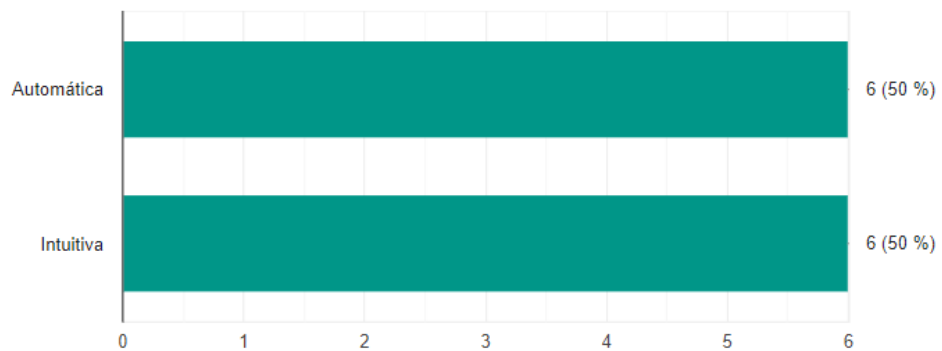


Usabilidad

 Copiar

¿Cómo te gustaría que funcionara una aplicación de Smart Agencies?

12 respuestas

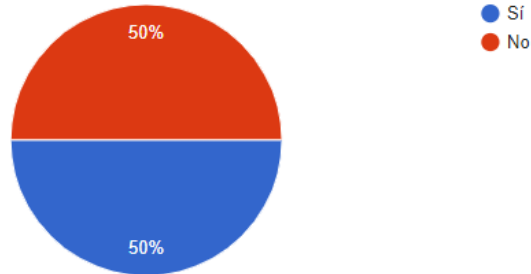


Pruebas

¿Te gustaría participar en las pruebas de una aplicación de Smart Agencies?

12 respuestas

 Copiar

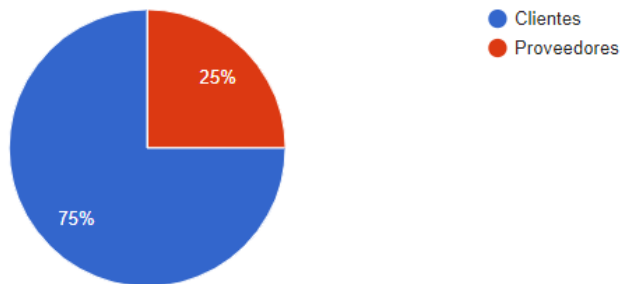


Preguntas específicas

¿Qué tipo de agencias gestionas?

12 respuestas

 Copiar

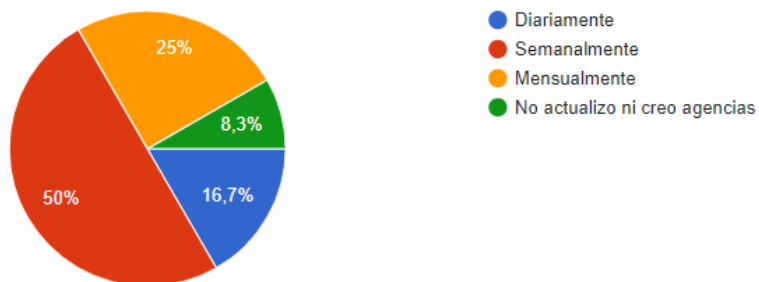


Preguntas específicas

¿Con qué frecuencia necesitas crear o actualizar agencias?

12 respuestas

 Copiar



- Entrevistas

Los resultados de las entrevistas los podemos encontrar en el siguiente repositorio de Google Drive. Por protección de datos, no se ha dejado reflejado ningún nombre de los participantes.

[Resultados](#)

Conclusiones

- Encuestas

Hemos recopilado la información de las encuestas y hemos obtenido estas conclusiones cuantitativas:

- El rol que predomina en la empresa es el de Operador.
- Se espera que este nuevo desarrollo use la información de una manera más eficiente y amigable.
- Aumentos de productividad y ahorrar tiempo a los usuarios son los mayores beneficios que pueden aportar nuestra aplicación según los usuarios.
- Hay igualdad en la forma de utilizar la herramienta, si de manera automática o de manera manual.
- La empresa trabaja más con clientes que con proveedores.
- Tienen una mayor necesidad de modificar las agencias semanalmente.

- Entrevistas

Después de analizar las entrevistas hemos observado estos datos cualitativos:


- La gran mayoría trabaja y vive en la sede de Palma de Mallorca.
- Una gran parte de los entrevistados coinciden en que debe de ser una aplicación:
 - Intuitiva
 - Fácil de utilizar
 - Integrada con otros sistemas
 - No tenga una interfaz cargada de elementos
- Varios usuarios han mencionado cambios en la interfaz por parte de los diseños, por lo que será un punto a tener en cuenta cuando estemos haciendo mejoras en Sprint.


- Tienen expectativas en la aplicación de móvil para poder hacer un mejor trabajo.


También hemos podido recopilar información para poder aplicar evolutivos de calidad del código para la integración de datos con los analistas y desarrolladores de la aplicación Smart Agencies Desktop dando un mejor servicio para ambas partes en el mismo presupuesto y tiempo.

2.2 Fichas de perfil de usuario

Para identificar los perfiles hemos utilizado las entrevistas y encuestas realizadas en el punto anterior. A partir de los datos recopilados, encontramos los siguientes perfiles de usuario:

<p style="text-align: center;">Lida</p> 	<p style="text-align: center;">Lista de comportamientos</p> <ul style="list-style-type: none"> → Le gustan las redes sociales → Soñadora → Organizada
<p style="text-align: center;">Demografía</p> <ul style="list-style-type: none"> → 23 años → Nació y vive en Palma de Mallorca → En una relación estable → Sin hijos 	<p style="text-align: center;">Necesidades y objetivos</p> <ul style="list-style-type: none"> → Viajar y descubrir mundo → Aprender y crecer tanto personal como laboralmente → Independizarse → Lleva poco en la empresa

<p style="text-align: center;">Lucas</p> 	<p style="text-align: center;">Lista de comportamientos</p> <ul style="list-style-type: none"> → Le gusta el mar → Siempre que puede le gusta disfrutar de su soledad → Se adapta rápido a las tecnologías
<p style="text-align: center;">Demografía</p> <ul style="list-style-type: none"> → 46 años → Nació fuera de Palma, pero se mudó a Palma de Mallorca → Soltero → 2 hijos 	<p style="text-align: center;">Necesidades y objetivos</p> <ul style="list-style-type: none"> → Disfrutar de sus hijos y darles lo mejor → Desconectar del trabajo → Hace poco que aceptó el trabajo, pero tiene mucha experiencia en este modelo de negocio

<p style="text-align: center;">Antonio</p> 	<p style="text-align: center;">Lista de comportamientos</p> <ul style="list-style-type: none"> → Olvidadizo → Le gusta tener el control → Torpe con las tecnologías
<p style="text-align: center;">Demografía</p> <ul style="list-style-type: none"> → 59 años → Nació en Madrid y vive en Barcelona → Casado → 4 hijos 	<p style="text-align: center;">Necesidades y objetivos</p> <ul style="list-style-type: none"> → Estabilidad → Viajar con su familia → Lleva muchos años en la empresa por lo que ya domina el trabajo

2.3 Point of view statements

- Perfil de usuario Lida

1. Las personas con un perfil de usuario con poca experiencia laboral necesitan una oportunidad para empezar en este mundo porque quieren tener éxito en una empresa y así poder optar a generar esa experiencia para poder progresar a mejores ofertas.
2. Las personas con un perfil de usuario de 23 años necesitan una forma de gestionar su tiempo y tareas de operadora porque tienen un calendario muy ajetreado y eso supondría retrasos en el resto de clientes.

- Perfil de usuario de Lucas

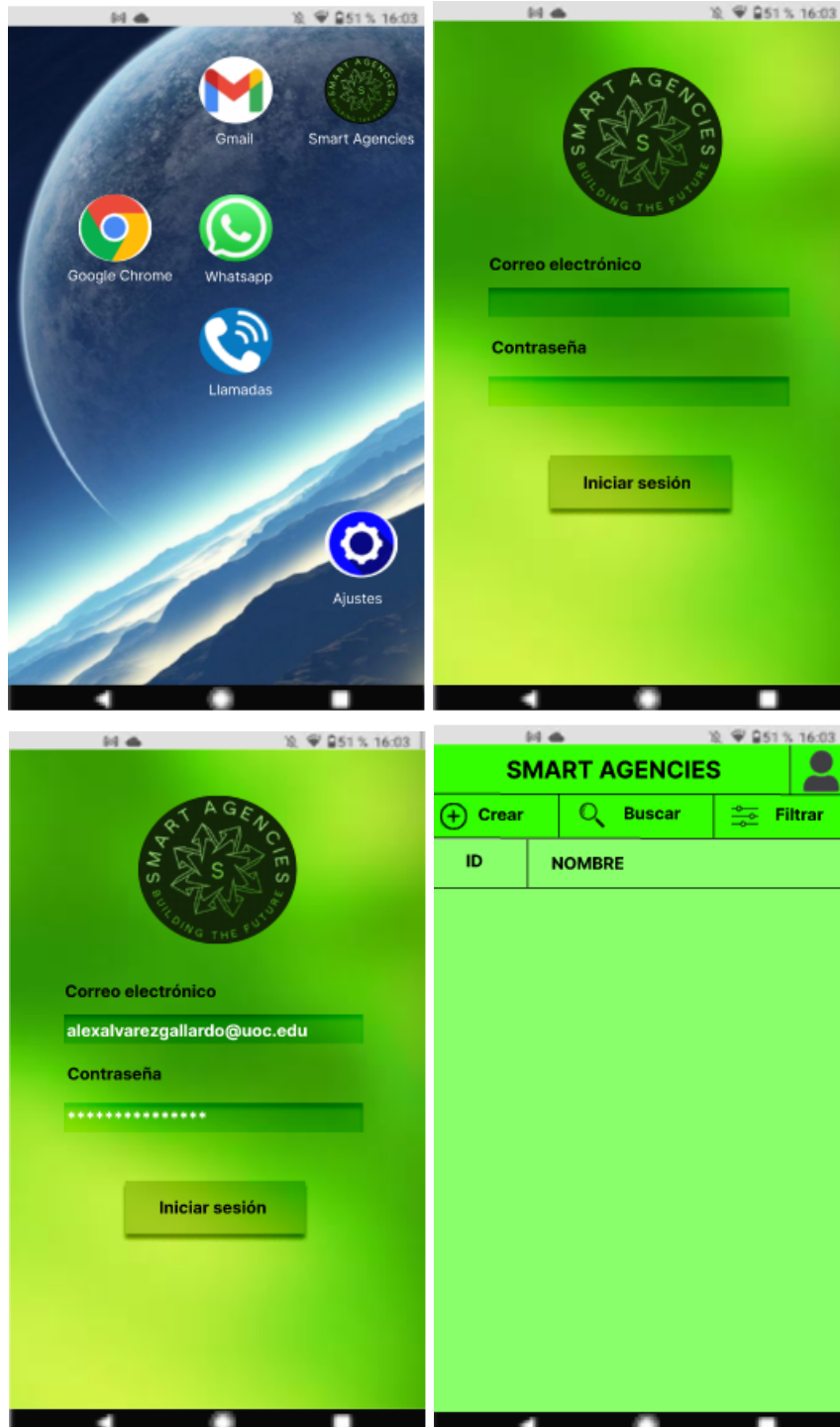
1. Las personas que disfrutan del mar y de la soledad necesitan una forma de gestionar agencias de forma flexible y personalizada para poder conciliar su vida personal y profesional.
2. Las personas que se adaptan rápido a las tecnologías de forma sencilla e intuitiva porque saben mucho del modelo de negocio y qué es lo que beneficia al cliente.
3. Las personas que tienen mucha experiencia en el sector necesitan una forma de gestionar agencias de forma eficiente y eficaz porque así podrán ofrecer un servicio de alta calidad a sus clientes.
4. Las personas que acaban de empezar en el trabajo necesitan una forma de gestionar las reservas de forma guiada y con apoyo porque sino podrían generarle problemas a la empresa.

- Perfil de usuario Antonio

1. Las personas que la tecnología no es su mejor fuerte, necesitan una aplicación intuitiva y sencilla de utilizar para realizar su trabajo porque quieren ofrecer un gran servicio de calidad y sentirse útiles.
2. Las personas que llevan muchos años en la empresa y que quieren viajar con su familia necesitan una forma de gestionar agencias de forma eficiente para poder ahorrar tiempo y dinero.
3. Las personas que son olvidadizas necesitan gestionar agencias de una forma automatizada para poder dedicarle tiempo a otras tareas con mayor prioridad.

2.4 Prototipo

2.4.1 Capturas de pantalla



FILTROS DE BÚSQUEDA

+ Crear

ID

Id

Nombre

Tipo de agencia ▼

Ámbito ▼

Reset **Buscar**

SMART AGENCIAS

+ Crear **Buscar** **Filtrar**

ID	NOMBRE
1801	Prueba agencia id 1801

1801 - PRUEBA AGENCIA 1801 ←

General **Datos comerciales** **Datos fiscales**

Nombre

Tipo de agencia ▼

Ámbito ▼

Baja **Guardar** **Cerrar**

1801 - PRUEBA AGENCIA 1801 ←

General **Datos comerciales** **Datos fiscales**

Calle

Número

CP

País comercial



2.4.2 Enlace al prototipo interactivo

Este prototipo de alta fidelidad es una representación visual interactiva de nuestro diseño Mínimo Producto Viable. Ha sido cuidadosamente elaborado para proporcionar una experiencia lo más cercana posible a la versión real. Cada detalle, desde la disposición de elementos hasta la interacción del usuario, ha sido perfeccionado para reflejar fielmente cómo funcionará el producto final. Puedes acceder al [prototipo en Figma](#) a través de este enlace.

2.5 Resultados del test con usuarios

Utilizando la plantilla que se ha adjuntado anteriormente hemos podido obtener las siguientes conclusiones que han sido representadas en forma de tabla.

[Google Drive con las respuestas a las entrevistas.](#)

	Usuario tipo perfil Lida. Entrevista 1. Poca experiencia en la empresa pero con muchas facilidades para la tecnología.	
	Inputs positivos	Inputs negativos
Concepto general		No acaba de entender el modelo de negocio
OnBoarding	Bien, parece que lo ha entendido todo y cómo va a funcionar la prueba.	
Iniciar sesión en la aplicación	Sin ningún problema para realizarlo.	
Filtrar por nombre una agencia	Sin ningún problema para realizarlo.	
Filtrar por Id una agencia	Sin ningún problema para realizarlo.	
Filtrar por tipo de agencia una agencia	Sin ningún problema para realizarlo.	
Crear una agencia	Sin ningún problema para realizarlo.	
Modificar una agencia	Sin ningún problema para realizarlo.	
Dar de baja una agencia	Sin ningún problema para realizarlo.	
Conclusión	<p>Positiva.</p> <p>No ha tenido ningún problema para seguir las tareas de la prueba y explicarnos que va haciendo para ir dando feedback. Por lo que ha transmitido, le ha resultado mucho más intuitiva que la aplicación de escritorio.</p> <p>No aporta mejoras.</p>	

	Usuario tipo perfil Antonio. Varón de unos 30 años, con mucho conocimiento de negocio y hay veces que se le cuesta adaptarse a nuevas tecnologías.	
	Inputs positivos	Inputs negativos
Concepto general	Bien, conoce el negocio muy bien y está contento por poder participar en este desarrollo.	
OnBoarding	Bien, parece que lo ha entendido todo y cómo va a funcionar la prueba.	
Iniciar sesión en la aplicación	Sin ningún problema para realizarlo.	
Filtrar por nombre una agencia	Sin ningún problema para realizarlo.	
Filtrar por Id una agencia	Sin ningún problema para realizarlo.	
Filtrar por tipo de agencia una agencia		Menciona que falta información, pero se le ha explicado que son datos estáticos. Lo ha entendido
Crear una agencia	Sin ningún problema para realizarlo.	
Modificar una agencia	Sin ningún problema para realizarlo.	
Dar de baja una agencia	Sin ningún problema para realizarlo.	
Conclusión	<p>Positiva.</p> <p>Ha tenido algún problema puntual pero nada que comentar. Es normal que surjan problemas en cuanto a datos se refiere ya que no se están trabajando con datos ni reales ni dinámicos.</p> <p>Mencionó estar contento con el primer prototipo de aplicación. No aporta mejoras. Ha comentado si puede seguir estando para futuros sprints.</p>	

	Usuario tipo perfil Lucas. Mujer con dos hijos. Tiene mucho conocimiento de negocio, pero acaba de empezar en la empresa	
	Inputs positivos	Inputs negativos
Concepto general	Lo ha entendido bien.	
OnBoarding		Ha marcado que participó en la primera fase de DCU, pero no se acuerda que hizo.
Iniciar sesión en la aplicación	Sin ningún problema para realizarlo.	
Filtrar por nombre una agencia	Sin ningún problema para realizarlo.	
Filtrar por Id una agencia	Sin ningún problema para realizarlo.	Problema al leer la tarea.
Filtrar por tipo de agencia una agencia	Sin ningún problema para realizarlo.	
Crear una agencia	Sin ningún problema para realizarlo.	
Modificar una agencia	Sin ningún problema para realizarlo.	
Dar de baja una agencia	Sin ningún problema para realizarlo.	
Conclusión	<p>Positiva.</p> <p>Ha tenido algún problema leyendo las tareas, pero se ha dado cuenta sola y las ha hecho sin ningún problema.</p> <p>No aporta mejoras.</p>	

	Usuario 4 tipo perfil Antonio. Varón muy exigente con las tecnologías y torpe a su vez.	
	Inputs positivos	Inputs negativos
Concepto general	Es la mejor entrevista que hemos hecho ya que hemos podido sacar puntos de mejoras para la aplicación y a tener en cuenta.	Ha sido el usuario que ha puesto peores valoraciones en la encuesta.
OnBoarding	Lo ha entendido bien.	
Iniciar sesión en la aplicación	Sin ningún problema para realizarlo.	
Filtrar por nombre una agencia	Sin ningún problema para realizarlo.	
Filtrar por Id una agencia	Sin ningún problema para realizarlo.	
Filtrar por tipo de agencia una agencia	Sin ningún problema para realizarlo.	
Crear una agencia		Problemas a la hora de crear la agencia, ya que no era capaz de guardarla desde otras pestañas.
Modificar una agencia		Problemas a la hora de modificar la agencia, ya que no era capaz de guardarla desde otras pestañas.
Dar de baja una agencia	Sin ningún problema para realizarlo.	
Conclusión	<p>Negativa.</p> <p>Ha mencionado que el botón de Buscar no aporta nada en la aplicación. No le ha gustado el diseño de la aplicación, pero comenta que la Smart Agencies Desktop también tiene un diseño muy feo.</p> <p>Remarca que los botones de crear/buscar/filtrar llamen más la atención. Y los botones en el detalle de la agencia deberían estar en cada una de las</p>	

	pestañas y no solo en general. Menciona como puede distribuirse la pestaña de integraciones con otros sistemas.
--	--

2.6 Puntos de mejora detectados

Como puntos de mejora que hemos podido obtener sacamos las siguientes conclusiones:

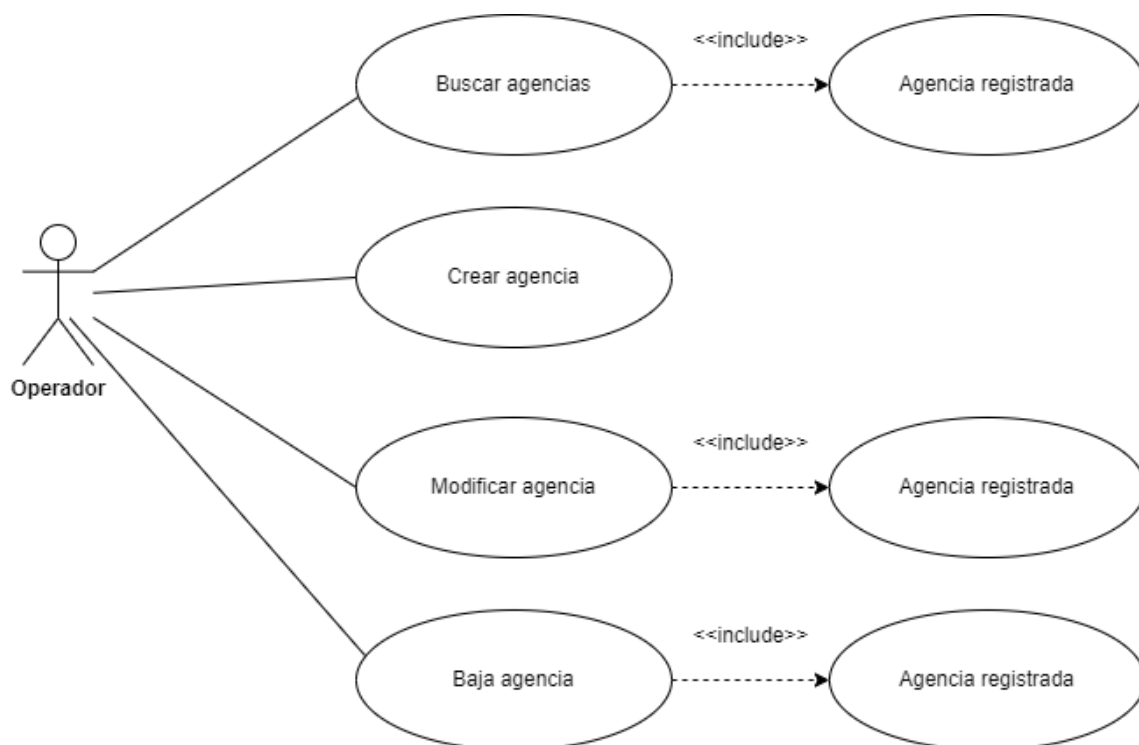
- Poner el botón de guardar agencia y cerrar agencia en todas las pestañas del detalle de la agencia para no forzar al usuario a que vuelva a la pestaña general y mejorar su UX.
- Un usuario ha comentado una idea de cómo puede distribuirse la pestaña de integraciones con otros sistemas, claramente es una funcionalidad que queda fuera del alcance inicial del MVP, por lo que dicha funcionalidad se implementará en un futuro sprint. Tendremos la información del usuario en cuenta cuando desarrollemos dicha funcionalidad.
- Deshabilitamos el botón de buscar agencias de la pantalla principal, ya que de momento según el cliente no vamos a tener mucho volumen de datos, por lo que esta funcionalidad será implementada cuando sea necesaria.

3. Diseño técnico (DT)

3.1 Definición de los casos de uso

Los casos de uso son una herramienta importante para el desarrollo de software. Al documentar las interacciones entre los usuarios y el sistema, los casos de uso ayudan a los desarrolladores a comprender las necesidades de los usuarios y a crear sistemas que sean fáciles de usar y efectivos.

3.1.1 Diagrama UML forma gráfica actores



3.1.2 Lista de los casos

Identificador	CU-001
Nombre	Buscar una agencia
Prioridad	Alta
Descripción	Buscar la información completa de una agencia en concreto
Actores	Operador
Pre-condiciones	Estar logueado en la aplicación y que la agencia exista
Iniciado por	Interés por la información que hay en una agencia
Flujo	Iniciar aplicación/Buscar nombre agencia/Obtener detalle agencia
Post-condiciones	Información completa de la agencia
Notas	

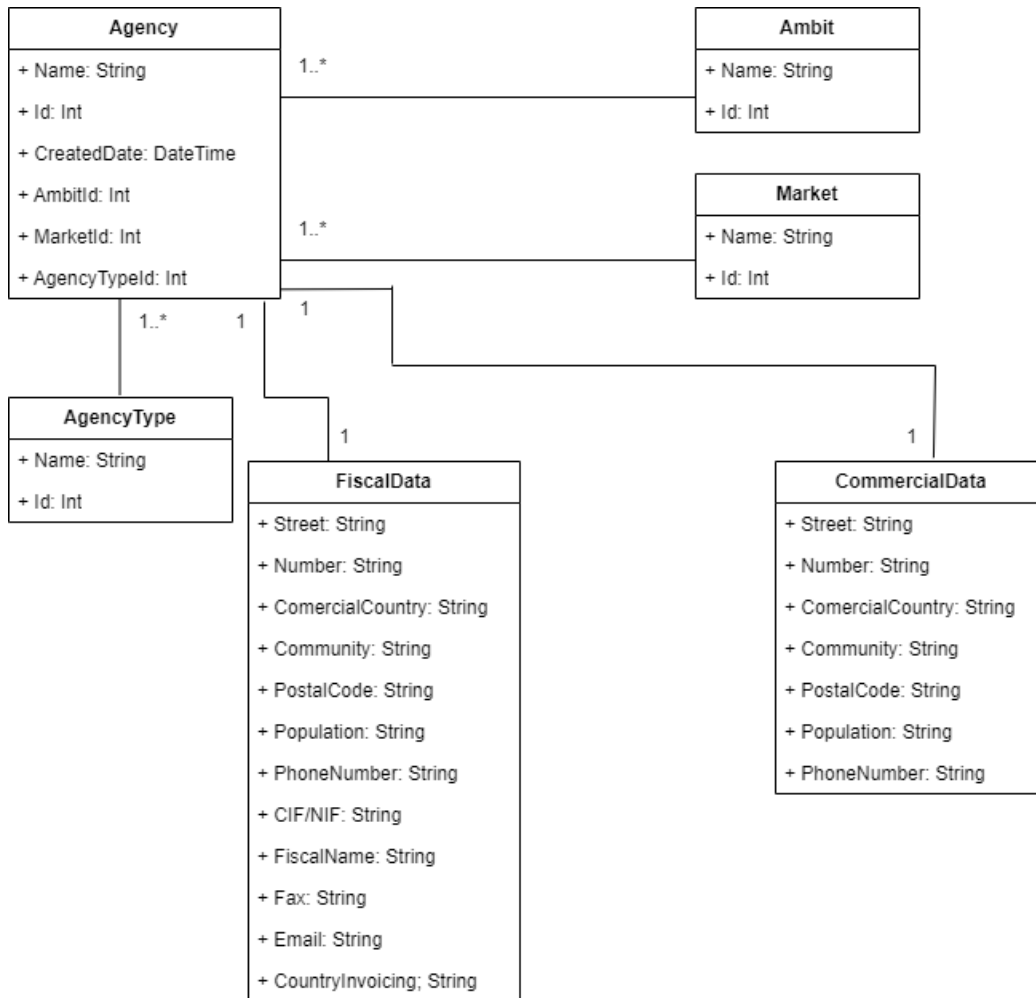
Identificador	CU-002
Nombre	Crear una agencia
Prioridad	Alta
Descripción	Crear una agencia para almacenar cierta información de esta
Actores	Operador
Pre-condiciones	Estar logueado en la aplicación
Iniciado por	Nuevo contrato creado con el cliente o proveedor
Flujo	Iniciar aplicación/Crear agencia/Rellenar datos de la agencia/Guardar la agencia
Post-condiciones	Información completa de la agencia
Notas	

Identificador	CU-003
Nombre	Modificar una agencia
Prioridad	Alta
Descripción	Modificar una agencia para almacenar cierta información de esta
Actores	Operador
Pre-condiciones	Estar logueado en la aplicación y que la agencia exista
Iniciado por	Modificado el contrato creado con el cliente o proveedor
Flujo	Iniciar aplicación/Buscar la agencia/Modificar datos de la agencia/Guardar la agencia
Post-condiciones	Información completa de la agencia
Notas	

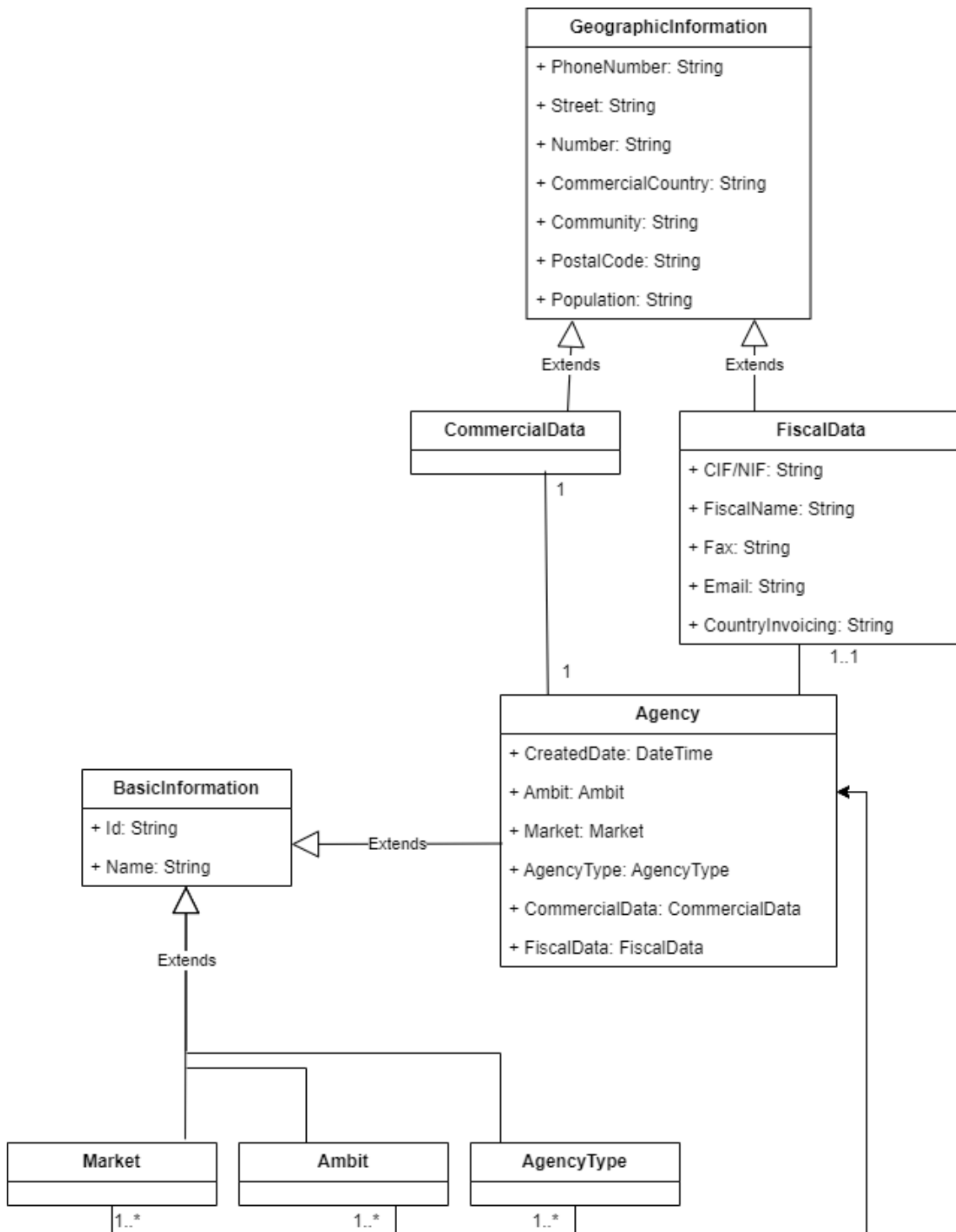
Identificador	CU-004
Nombre	Dar de baja una agencia
Prioridad	Alta
Descripción	Dar de baja una agencia
Actores	Operador
Pre-condiciones	Estar logueado en la aplicación y que la agencia exista
Iniciado por	Dado de baja el contrato creado con el cliente o proveedor
Flujo	Iniciar aplicación/Buscar la agencia/Dar de baja la agencia
Post-condiciones	-
Notas	

3.2 Diseño de la arquitectura

3.2.1 Diagrama UML diseño de la base de datos

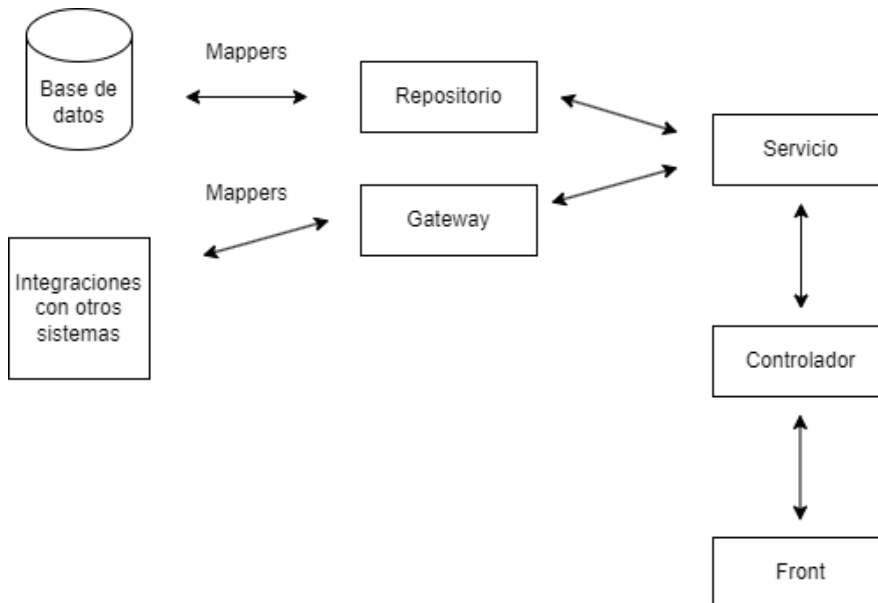


3.2.2 Diagrama UML diseño de las entidades y clases



3.2.3 Diagrama arquitectura del sistema

3.2.3.1 Diagrama Backend



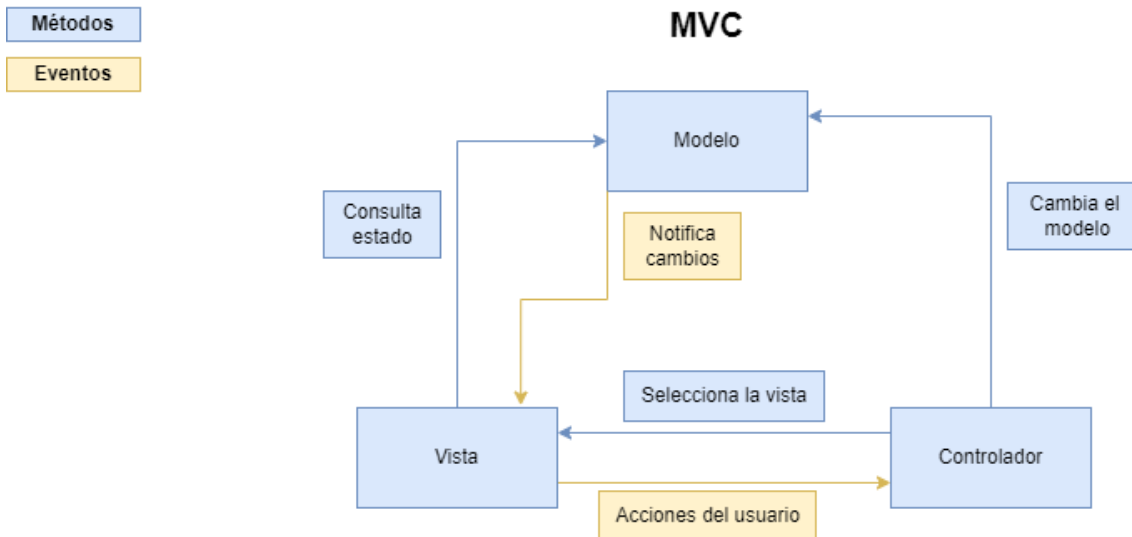
En nuestra aplicación para el backend hemos implementado una arquitectura por capas para poder modularizar las responsabilidades de cada una de las partes del código. El flujo para una interacción general sería:

1. **Controladores:** Cuando se recibe una llamada desde el frontend o desde una herramienta como Postman, esta solicitud es manejada por los controladores. Los controladores actúan como puntos de entrada para las solicitudes y rutas, gestionando la interacción inicial con la aplicación.
2. **Servicios:** Una vez que la solicitud ha sido manejada por los controladores, se pasa a la capa de servicios. Aquí se encuentra la lógica de negocio de la aplicación. Los servicios aplican las reglas y la lógica necesaria para procesar la solicitud de manera adecuada.
3. **Repositorios y Gateway:** Para obtener los datos necesarios, los servicios interactúan con los repositorios. Estos repositorios son responsables de interactuar con la fuente de datos de la aplicación, como una base de datos interna. Además, se menciona la existencia de un Gateway que, aunque actualmente no se utiliza, está diseñado para interactuar con API de sistemas externos. El Gateway y el Repositorio deben realizar el mapeo de modelos para asegurar la comunicación adecuada con los sistemas que interactúan.
4. **Respuesta del Repositorio o Gateway:** Una vez que los repositorios o el Gateway han interactuado con la fuente de datos, devuelven la

respuesta al servicio correspondiente. Esto puede incluir datos que se han recuperado o procesado.

5. **Interacción del Servicio:** En el servicio, se aplican las reglas de negocio necesarias a los datos recibidos de los Repositorios o del Gateway. Esto asegura que los datos se procesen de acuerdo con las lógicas específicas de la aplicación.
6. **Respuesta del Controlador:** Una vez que los servicios han procesado la solicitud y aplicado las reglas de negocio, devuelven el resultado al controlador. El controlador se encarga de transmitir la respuesta de vuelta al frontend, asegurando que los datos se presenten de manera adecuada al usuario o a la herramienta desde la cual se originó la solicitud.

3.2.3.2 Diagrama Frontend



En nuestra aplicación, hemos implementado el patrón MVC para gestionar las solicitudes y presentar datos al usuario de manera organizada. Este patrón implica:

1. El controlador recibe las solicitudes de los usuarios y dirige las solicitudes al modelo apropiado.
2. El modelo accede a la base de datos y procesa los datos necesarios, que luego devuelve al controlador.
3. El controlador selecciona la vista adecuada en función de la lógica de la aplicación.
4. La vista se carga con los datos del modelo seleccionado y se presenta al usuario.

El patrón MVC divide claramente las responsabilidades en la aplicación, lo que facilita la flexibilidad y el mantenimiento a medida que la aplicación evoluciona, asegurando un diseño organizado y sólido.

3.2.3.3 Comunicación entre frontend y backend

La comunicación entre el frontend y el backend sigue el modelo Cliente-Servidor mediante solicitudes HTTP. Cuando el usuario realiza una acción en el frontend, se genera una solicitud HTTP que viaja al backend. Allí, un controlador procesa la solicitud, aplicando la lógica de negocio en la capa de servicio y persistiendo datos en la capa de repositorio. El backend responde con datos o un estado (éxito/error), y el frontend actualiza la interfaz de usuario en consecuencia.

4. Implementación

4.1 Desarrollo

Se ha llevado a cabo la implementación de un sistema integral utilizando React Native para el Frontend y .NET Core 6 para el Backend. Este apartado detalla las herramientas empleadas, el estado actual del proyecto y las pruebas realizadas. React Native se seleccionó por su capacidad multiplataforma, mientras que .NET Core 6 aporta escalabilidad y rendimiento al Backend.

Se explorarán las decisiones clave de implementación, los logros alcanzados, desafíos superados y el estado general del proyecto. Además, se revisarán las pruebas realizadas en todas las capas del sistema para garantizar su calidad y fiabilidad. Este análisis técnico destaca cómo estas elecciones contribuyen al éxito de los objetivos establecidos en este “Trabajo de Fin de Grado”.

4.1.1 Herramientas utilizadas en el desarrollo de la aplicación

El desarrollo de este proyecto se ha visto potenciado por una cuidadosa selección de herramientas que abarcan todas las fases del proceso. Desde el entorno ágil de desarrollo en React Native con Visual Studio Code hasta la gestión de bases de datos en Microsoft SQL Server Management, el despliegue en la nube mediante el Portal de Azure y el seguimiento de metodologías y control de versiones con Azure DevOps. Cada herramienta ha sido elegida estratégicamente.

Este apartado detalla la funcionalidad y justificación detrás de cada elección, destacando cómo estas herramientas han convergido para asegurar un desarrollo eficiente y de alta calidad del proyecto.

- Visual Studio Code para React Native

En el desarrollo del Frontend con React Native, se optó por utilizar Visual Studio Code como entorno de desarrollo principal. Esta elección se basa en la eficiencia y la extensibilidad que ofrece este entorno. Las extensiones específicas para React Native en Visual Studio Code desempeñaron un papel crucial al proporcionar herramientas esenciales como resaltado de sintaxis, autocompletado, depuración y soporte para la gestión de paquetes. Esta combinación facilita un flujo de trabajo fluido y centrado en la calidad del código.

- Visual Studio para .NET Core 6

Para el desarrollo del Backend con .NET Core 6, se seleccionó Visual Studio como la herramienta principal. Este entorno integral proporciona un conjunto completo de herramientas que simplifican el diseño, la depuración y el despliegue de aplicaciones basadas en .NET Core. La elección de Visual Studio se fundamenta en su estatus como herramienta estándar en el ecosistema .NET, ofreciendo funcionalidades avanzadas y una interfaz de usuario robusta que mejora significativamente la eficiencia del desarrollo.

- Seguridad en la API: Aparte, se ha implementado un sistema de seguridad en la API mediante el uso de una Api Key. Esta medida garantiza un acceso controlado y autorizado a los recursos del sistema consiguiendo una confidencialidad de los datos. No es algo obligatorio pero, asegura que solo las entidades autorizadas puedan interactuar con la API, contribuyendo así a la seguridad integral del proyecto

- Microsoft SQL Server Management para SQL

En la gestión de la capa de datos a través de SQL, se empleó Microsoft SQL Server Management Studio. Esta herramienta se destaca por su capacidad para administrar y desarrollar bases de datos en servidores SQL de Microsoft. Su interfaz intuitiva y sus funcionalidades completas permitieron una administración eficiente de las bases de datos, facilitando el diseño de esquemas, la ejecución de consultas SQL y el monitoreo del rendimiento.

- Portal de Azure para el sistema cloud

Para la implementación en la nube, se recurrió al Portal de Azure. Esta plataforma web proporciona una interfaz unificada para administrar y desplegar servicios en la nube de Microsoft Azure. La elección de Azure se basa en su posición como una plataforma de nube ampliamente utilizada, mientras que el Portal de Azure garantiza una experiencia sencilla y completa para la gestión de recursos en entornos cloud, permitiendo un despliegue y supervisión eficientes de la aplicación.

- Azure DevOps para seguimiento de metodologías ágiles, sistema de versionado y CI/CD

El seguimiento de metodologías ágiles y el control de versiones se gestionaron mediante Azure DevOps. Esta plataforma incorpora herramientas como Azure Boards para el seguimiento ágil, Azure Repos para el control de versiones y Azure Pipelines para el CI/CD. La elección de Azure DevOps se justifica por su capacidad para proporcionar una

solución completa para la gestión del ciclo de vida del desarrollo de software, simplificando la planificación, el seguimiento y la colaboración en equipos de desarrollo.

- Genymotion para el emulador de dispositivo móvil

En el ámbito de pruebas de dispositivos móviles, se integró Genymotion como emulador. Este software se distingue por su rapidez y eficiencia en la simulación de dispositivos Android para pruebas y depuración. La elección de Genymotion proporcionó un entorno de prueba confiable para el desarrollo en React Native, permitiendo la simulación de diversos dispositivos Android con diferentes especificaciones y versiones del sistema operativo.

- Postman para Pruebas en la API

Postman fue esencial en las fases iniciales para realizar pruebas en la API. Su interfaz intuitiva y capacidad para gestionar colecciones de solicitudes facilitaron la validación eficiente del Backend antes de la implementación de Swagger, mejorando la confiabilidad y corrección de la API.

4.1.2 Estado del proyecto

El proyecto ha avanzado de acuerdo con la planificación establecida, cumpliendo con los plazos de entrega definidos. La planificación detallada se encuentra registrada y actualizada en nuestro entorno de Azure DevOps, proporcionando una visión clara de cada hito, el estado actual de las tareas y la información acumulada durante el desarrollo. Este enfoque ha permitido un seguimiento efectivo y ha facilitado la toma de decisiones basada en datos concretos.

Si bien se han alcanzado metas importantes, se reconoce la necesidad de fortalecer la estrategia de pruebas. Aunque se han completado pruebas fundamentales, como las unitarias, se identifica la oportunidad de ampliar la cobertura con tests de integración y pruebas de carga. Además, se contempla la incorporación de tests específicos para la parte de React Native, contribuyendo así a la robustez y confiabilidad del sistema en su totalidad.

En el ámbito de la nube, se ha suscrito a una plataforma que permite el despliegue de los servicios esenciales para el funcionamiento de la aplicación. El portal Azure proporciona un acceso directo a esta infraestructura, facilitando la administración y monitorización continua de los recursos en la nube.

Para una evaluación más detallada, se adjuntan los enlaces pertinentes al portal de Azure y Azure DevOps. Estos recursos proporcionan una visión completa y transparente del estado del proyecto, permitiendo a los interesados acceder a la información clave y seguir de cerca el progreso general.

Aquí podremos encontrar más información sobre el proyecto en el [Portal de Azure](#) y en [Azure DevOps](#).

Desviaciones del Proyecto y Adaptaciones Estratégicas

Durante la fase inicial de la toma de requisitos, se llegó a la conclusión de que el despliegue del proyecto en la nube debería realizarse utilizando los servicios de AWS, dado que el cliente ya contaba con una infraestructura en esta plataforma. Aunque se había advertido previamente sobre la falta de conocimiento específico en AWS, se intentó adaptar al entorno de trabajo existente. Sin embargo, ante los desafíos encontrados y la amenaza de posibles retrasos en el cronograma debido a la necesidad de investigar a fondo AWS y a la falta de familiaridad por parte del desarrollador con dicho entorno, se tomó la decisión estratégica de cambiar a la plataforma Azure. Esta elección se basó en la experiencia previa del equipo con Azure y en la necesidad de garantizar la viabilidad del proyecto en términos de plazos y conocimiento del entorno.

En segundo lugar, se había considerado la implementación de un sistema de inicio de sesión desde cero. No obstante, al explorar y evaluar las capacidades de Azure, se planteó la posibilidad de gestionar el acceso a la aplicación desde la nube utilizando la cuenta corporativa existente. Esta adaptación estratégica, discutida con el departamento de sistemas, brindó una solución eficiente que simplificó el proceso de autenticación y mejoró la experiencia del usuario al consolidar el acceso a través de una única cuenta corporativa.

En tercer lugar, se tenía la idea de implementar los pipelines con Azure Pipelines para poder hacer CI/CD y agregar controles de calidad sobre el código como puede ser SonarQube. Debido al plan contratado con Azure por parte del cliente (Azure free for students) no se permite realizar esta funcionalidad.

Finalmente, se había propuesto la implementación de una arquitectura Modelo-Vista-Controlador (MVC) en React Native. Sin embargo, durante la planificación del proyecto, se identificó como riesgo la falta de experiencia en frameworks específicos, lo que podría afectar la aplicación de la arquitectura prevista. Para abordar este desafío, se ha decidido proponer una solución alternativa al cliente. Esta solución no sólo mitigará los riesgos asociados con la falta de experiencia, sino que también se ofrecerá a un costo reducido, demostrando nuestro compromiso con la entrega del proyecto.

Estas adaptaciones estratégicas son el resultado de una evaluación continua de los riesgos y desafíos del proyecto, así como de la búsqueda proactiva de soluciones para garantizar la calidad y la eficiencia en la ejecución del trabajo. En última instancia, estas decisiones están alineadas con el objetivo de lograr un proyecto exitoso y satisfacer las necesidades del cliente de manera efectiva

4.2 Pruebas sobre la aplicación

El proceso de prueba de la aplicación móvil se ha diseñado considerando el uso de Genymotion como simulador de dispositivos móviles, para el frontend se usará Visual Studio Code y Visual Studio para el backend. Aunque, en la fase inicial, no se contemplan pruebas a nivel de frontend, se establecerán en futuros sprints. La ejecución de pruebas se centrará en garantizar la funcionalidad adecuada del backend y su integración con los recursos de Azure. Adicionalmente, se explorará la posibilidad de crear un entorno de pruebas en el portal de Azure denominado "UAT" (User Acceptance Testing).

Configuración del Entorno de Desarrollo

- Instalación de Genymotion:
 - Se requerirá la instalación y configuración previa de Genymotion, el emulador de dispositivos móviles elegido para las pruebas de la aplicación Android.
- Configuración de Visual Studio Code:
 - Verificar que Visual Studio Code esté debidamente configurado para el desarrollo del frontend, asegurando la presencia de todas las dependencias y herramientas necesarias.
- Configuración de Visual Studio:
 - Verificar que Visual Studio esté debidamente configurado para el desarrollo del backend, asegurando la presencia de todas las dependencias y herramientas necesarias.

Despliegue del Backend y Pruebas Locales

- Arranque de la API en Visual Studio:
 - Iniciar la API desde Visual Studio, asegurándose de que el proceso se complete sin errores y la API responda correctamente a las solicitudes.
- Pruebas Locales:
 - Realizar pruebas exhaustivas en el entorno local para verificar la funcionalidad y la correcta respuesta de la API a diversas solicitudes.

Pruebas con Genymotion:

- Integración con el Simulador:
 - Ejecutar la aplicación móvil en Genymotion para evaluar la interoperabilidad y el rendimiento en un entorno simulado.

Pruebas de Integración con Recursos de Azure:

- Utilización de Recursos en el Portal de Azure:

- Emplear el recurso disponible en el portal de Azure para pruebas, garantizando una correcta conexión y comunicación entre la aplicación y los servicios en la nube.

Creación de Entorno de Pruebas en Azure (Opcional):

- Configuración del Entorno de Pruebas UAT:
 - Explorar la posibilidad de establecer un entorno de pruebas dedicado en el portal de Azure denominado "UAT", proporcionando un espacio controlado para pruebas futuras y aceptación del usuario.

Este enfoque sistemático de pruebas garantizará la estabilidad y funcionalidad del backend, preparando el terreno para la implementación de pruebas de frontend en sprints posteriores y permitiendo una integración fluida con los servicios en la nube de Azure.

3. Conclusiones

3.1 Lecciones aprendidas a lo largo del trabajo

Realizar el trabajo de final de grado ha representado una travesía formativa donde he desarrollado habilidades esenciales en la gestión integral de proyectos, desde la toma de requisitos hasta la implementación de arquitecturas. La adopción de React Native ha ampliado mi conjunto de habilidades, aportando una perspectiva valiosa en el desarrollo de aplicaciones móviles.

Adicionalmente, he consolidado mis competencias en la gestión de proyectos en un entorno cloud como Azure. Esta experiencia ha ampliado mi comprensión sobre la infraestructura y servicios en la nube, destacando su relevancia en el desarrollo de soluciones escalables y eficientes.

Asimismo, he mejorado mis habilidades de redacción para informes finales, reconociendo la importancia de comunicar de manera clara y efectiva los detalles técnicos y los resultados del proyecto. Estos aprendizajes no solo han fortalecido mi capacidad para abordar proyectos complejos, sino que también han sentado las bases para mi crecimiento continuo en el campo de la ingeniería de software y la gestión de proyectos.

3.2 Reflexión sobre los objetivos planteados inicialmente

Al evaluar los objetivos planteados inicialmente para este proyecto, es evidente que hemos alcanzado la mayoría de las metas establecidas de manera satisfactoria. La aplicación desarrollada cumple con las expectativas del cliente al ofrecer funcionalidades clave tanto en el frontend como en el backend, utilizando tecnologías modernas como React Native y ASP.NET Core con Entity Framework Core.

Sin embargo, se destaca un ajuste necesario en relación con la elección del proveedor cloud. Aunque el objetivo original era utilizar AWS, se tomó la decisión de migrar a Azure debido a la preferencia y gestión específica del cliente en esta área. Esta adaptación nos permitió mantener la coherencia con la visión del cliente y garantizar una integración fluida en la infraestructura existente.

En consecuencia, debido a esta modificación, se optó por eliminar la funcionalidad de inicio de sesión en la aplicación. El cliente ya disponía de una solución de gestión de inicio de sesión a través de una intranet, haciendo innecesaria la duplicación de esta funcionalidad en nuestra aplicación. Esta adaptación demuestra la flexibilidad y adaptabilidad del equipo para ajustarse a las necesidades cambiantes del cliente, priorizando la eficiencia y la integración sin comprometer la calidad del producto final.

Aunque se ha producido un cambio en la dirección original del proyecto, la capacidad de respuesta y la comunicación efectiva con el cliente han sido

fundamentales para mitigar cualquier impacto negativo. Este proceso ha fortalecido nuestra comprensión de la importancia de la colaboración continua con el cliente y la necesidad de adaptarse ágilmente a los cambios en las circunstancias del proyecto.

En conclusión, a pesar de los desvíos en los objetivos iniciales, la capacidad de ajuste y la colaboración efectiva han permitido cumplir con las expectativas del cliente y ofrecer un producto que satisface sus necesidades específicas. Esta experiencia destaca la importancia de la flexibilidad y la comunicación en el desarrollo exitoso de proyectos de software.

3.3 Reflexión sobre el seguimiento de la planificación y la metodología

La elección de adoptar Scrum como metodología ágil para el desarrollo del proyecto ha demostrado ser una decisión acertada. Esta metodología proporciona la flexibilidad necesaria para afrontar cambios en un proyecto de gran envergadura, permitiendo entregas incrementales y una mejora continua. La capacidad de dividir el proyecto en bloques más manejables ha facilitado el control del proceso de desarrollo, asegurando la priorización de la calidad en cada etapa.

La adaptación del código de la API a una tecnología más actualizada subraya el compromiso con la innovación y la eficiencia. Si bien supuso el desarrollo de un nuevo producto en cuanto al framework, esta transición se considera esencial para mantenerse al día con las demandas del mercado y proporcionar un producto moderno y sostenible a largo plazo.

La planificación del trabajo, representada de manera clara en el diagrama de Gantt, ha proporcionado una visión detallada y estructurada del proyecto. El desglose de tareas en hitos parciales con sus respectivas fechas, duración y estado, ha permitido un seguimiento riguroso y eficiente. Sin embargo, la inflexibilidad de la tarifa negociada por el CIO plantea un desafío financiero, ya que no permite ajustes directos con el cliente. Esto destaca la importancia de una gestión cuidadosa de los recursos y la necesidad de considerar gastos adicionales generados para la empresa.

En cuanto a la planificación de riesgos, la identificación de posibles desafíos, como cambios en el alcance del cliente, riesgos tecnológicos y recursos insuficientes, demuestra una mentalidad proactiva y la disposición para abordar problemas potenciales. La transparencia sobre estos riesgos proporciona una base sólida para la toma de decisiones y la implementación de estrategias de mitigación.

En resumen, la combinación de una metodología ágil y sólida, una planificación detallada y la identificación proactiva de riesgos reflejan un enfoque estratégico y cuidadoso para el desarrollo de este proyecto. Aunque se presenten desafíos, la dedicación a la calidad y la adaptabilidad demuestran la preparación del

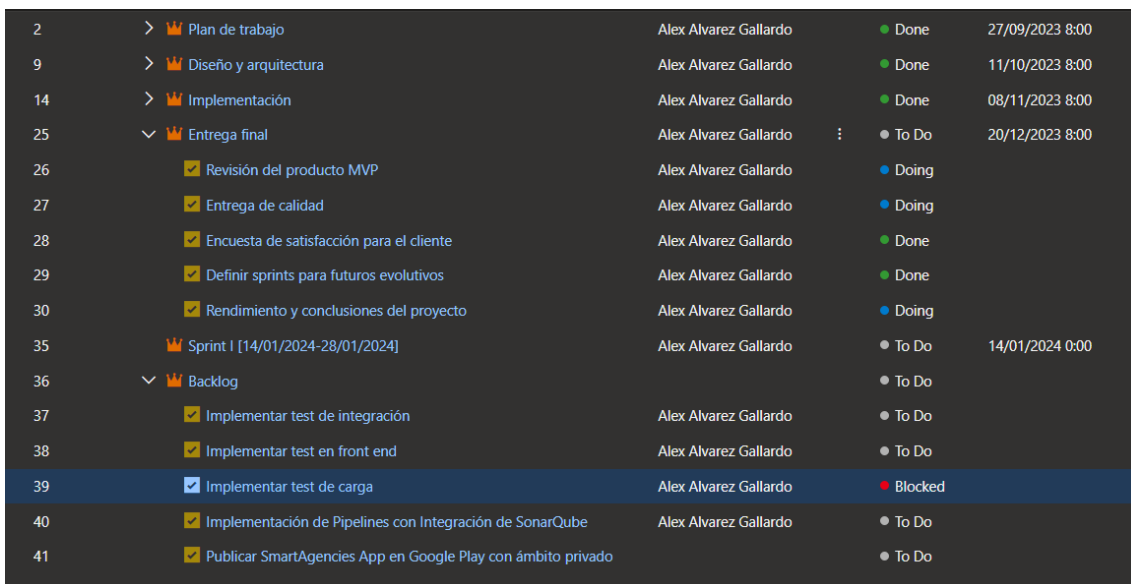
equipo para superar obstáculos y alcanzar el éxito en la entrega del producto solicitado.

3.4 Futuras líneas de trabajo

Con el proyecto en curso y enfocándonos hacia el futuro, se ha elaborado un backlog en Azure DevOps que delinearé las futuras líneas de trabajo y mejoras técnicas en nuestra plataforma. Estas tareas, seleccionadas estratégicamente, se centran en la evolución tecnológica y la optimización continua del producto, alineadas con los objetivos de innovación y eficiencia.

El backlog abarca diversas áreas, desde la implementación de nuevas funcionalidades hasta mejoras en la arquitectura y la experiencia del usuario. Cada tarea se ha priorizado considerando su impacto en la calidad, la usabilidad y la alineación con los objetivos del cliente. No obstante, la ejecución de estas tareas dependerá de la negociación con el Product Owner designado por el cliente, quien será clave en la toma de decisiones sobre las prioridades y la dirección futura del producto.

Entre las tareas propuestas y ya analizadas se encuentran:



ID	Tarea	Asignado a	Estado	Fecha
2	Plan de trabajo	Alex Alvarez Gallardo	Done	27/09/2023 8:00
9	Diseño y arquitectura	Alex Alvarez Gallardo	Done	11/10/2023 8:00
14	Implementación	Alex Alvarez Gallardo	Done	08/11/2023 8:00
25	Entrega final	Alex Alvarez Gallardo	To Do	20/12/2023 8:00
26	Revisión del producto MVP	Alex Alvarez Gallardo	Doing	
27	Entrega de calidad	Alex Alvarez Gallardo	Doing	
28	Encuesta de satisfacción para el cliente	Alex Alvarez Gallardo	Done	
29	Definir sprints para futuros evolutivos	Alex Alvarez Gallardo	Done	
30	Rendimiento y conclusiones del proyecto	Alex Alvarez Gallardo	Doing	
35	Sprint I [14/01/2024-28/01/2024]	Alex Alvarez Gallardo	To Do	14/01/2024 0:00
36	Backlog		To Do	
37	Implementar test de integración	Alex Alvarez Gallardo	To Do	
38	Implementar test en front end	Alex Alvarez Gallardo	To Do	
39	Implementar test de carga	Alex Alvarez Gallardo	Blocked	
40	Implementación de Pipelines con Integración de SonarQube	Alex Alvarez Gallardo	To Do	
41	Publicar SmartAgencies App en Google Play con ámbito privado		To Do	

Es crucial destacar que la ejecución de estas tareas dependerá de la aprobación y priorización por parte del Product Owner del cliente, asegurando que cada avance esté alineado con las expectativas y necesidades en evolución. Este enfoque iterativo y colaborativo es esencial para garantizar que el producto no solo cumpla, sino que supere, las expectativas del cliente a lo largo del tiempo.

4. Glosario

- API → Application Programming Interface
- ASP.NET Core → framework de desarrollo web de código abierto y multiplataforma
- AWS → Amazon Web Services
- CIO → Chief Information Officer
- EntityFramework Core → marco de trabajo que permite a los desarrolladores de .NET Core trabajar con bases de datos relacionales de forma sencilla y eficiente.
- MVP → Minimum Viable Product
- React Native → framework de JavaScript de código abierto para crear aplicaciones móviles nativas para iOS y Android
- SQL → Structured Query Language
- Api Key → cadena de caracteres que se utiliza para autenticar y autorizar las solicitudes a una interfaz de programación de aplicaciones (API)
- CI/CD → Continuous Integration/Continuous Deployment
- UAT → User Acceptance Testing
- MVC → Modelo-Vista-Controlador
- SonarQube → plataforma de código abierto diseñada para evaluar y analizar la calidad del código fuente en proyectos de software.

5. Bibliografía

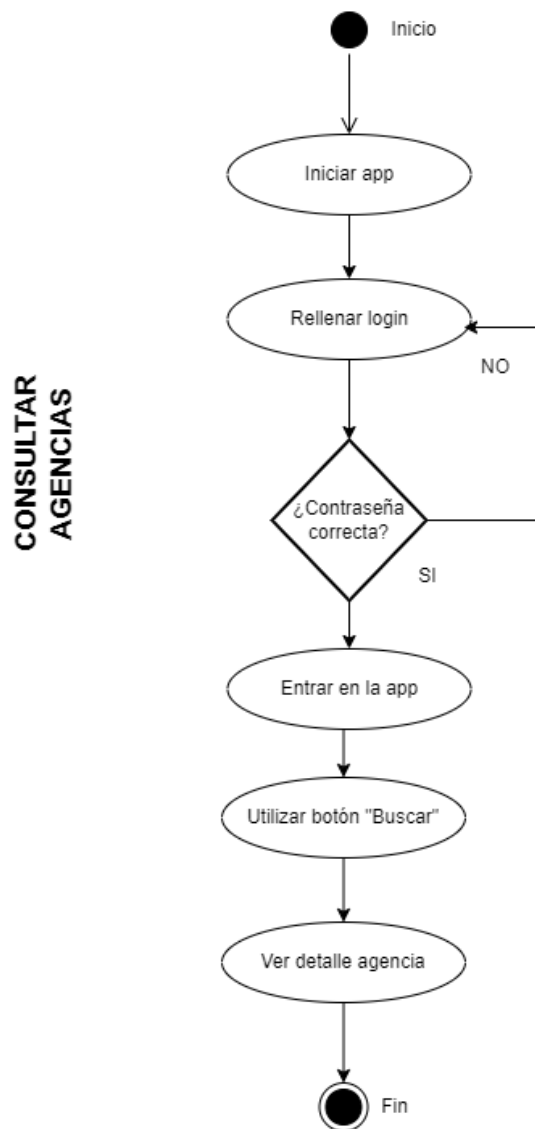
- [Información sobre Azure DevOps](#) [1]
- [Información sobre React Native](#) [2]
- [Documentación oficial React Native](#) [3]
- [Página para generar el diagrama de Gantt](#) [4]
- [Material del campus sobre trabajo final de grado](#)
- [Material del campus sobre entornos de programación móvil](#) [5]
- [Material del campus sobre desarrollo de aplicaciones basadas en Android](#)
- [Introducción a AWS](#) [6]
- [Instalación Node.js y npm](#)
- [Instalación Git](#)
- [Instalación JDK 8](#)
- [Instalación Genymotion](#)
- [Instalación Android Studio](#)
- [Instalación Visual Studio Code](#)
- [Instalación Visual Studio \(IDE\)](#)
- [Iconos aplicación](#)
- [Documentación oficial Portal Azure](#)

6. Anexos

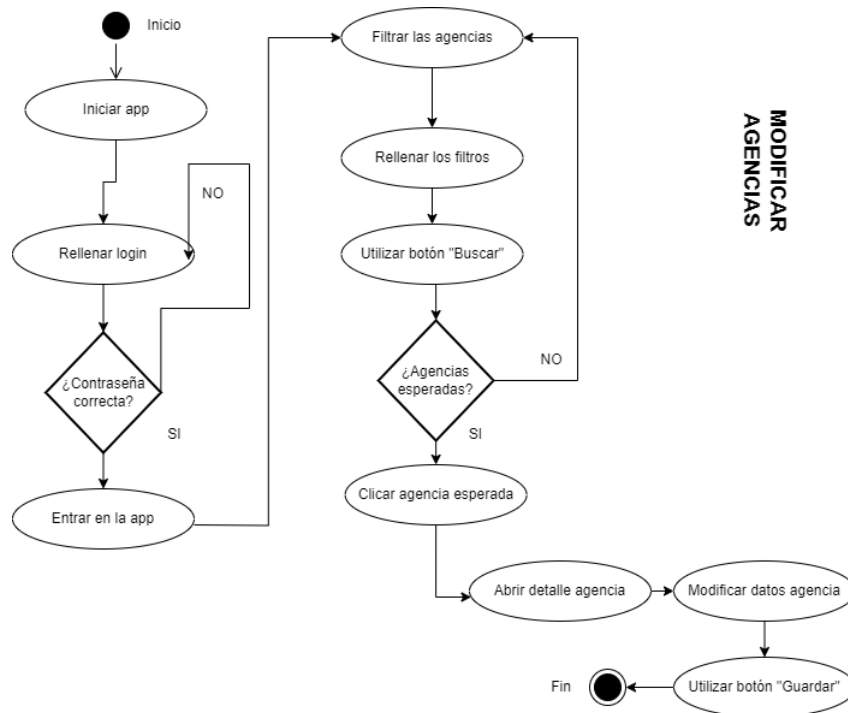
Anexo I : Flujos de interacción DCU

Los flujos de interacción son una herramienta valiosa para el diseño de interacción. Al comprender los flujos de interacción de las tareas que realizan los usuarios, los diseñadores pueden crear experiencias que sean fáciles de usar y efectivas. En nuestro desarrollo hemos encontrado los siguientes flujos e interacción:

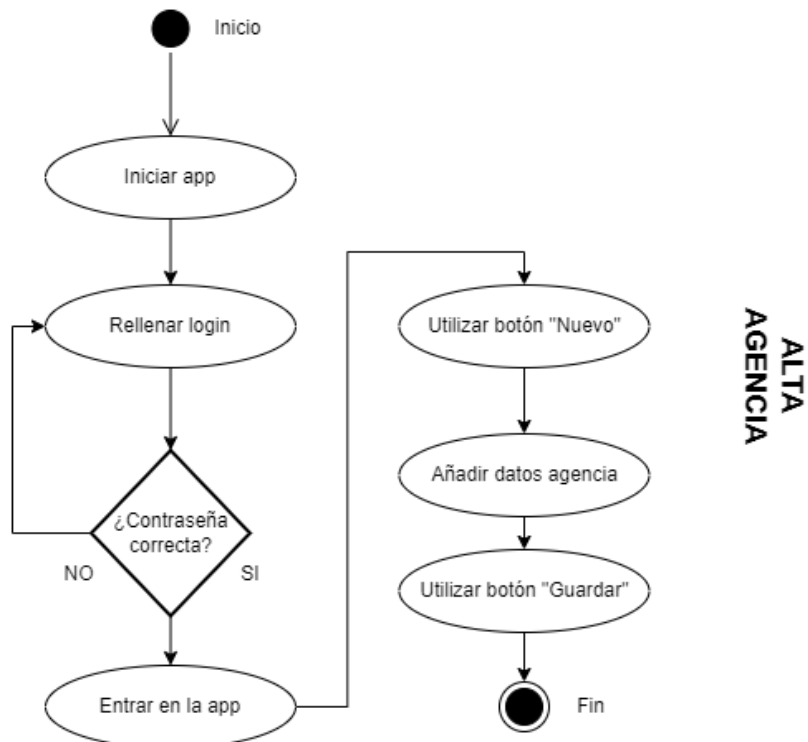
- Consultar agencias



- **Modificar agencias**

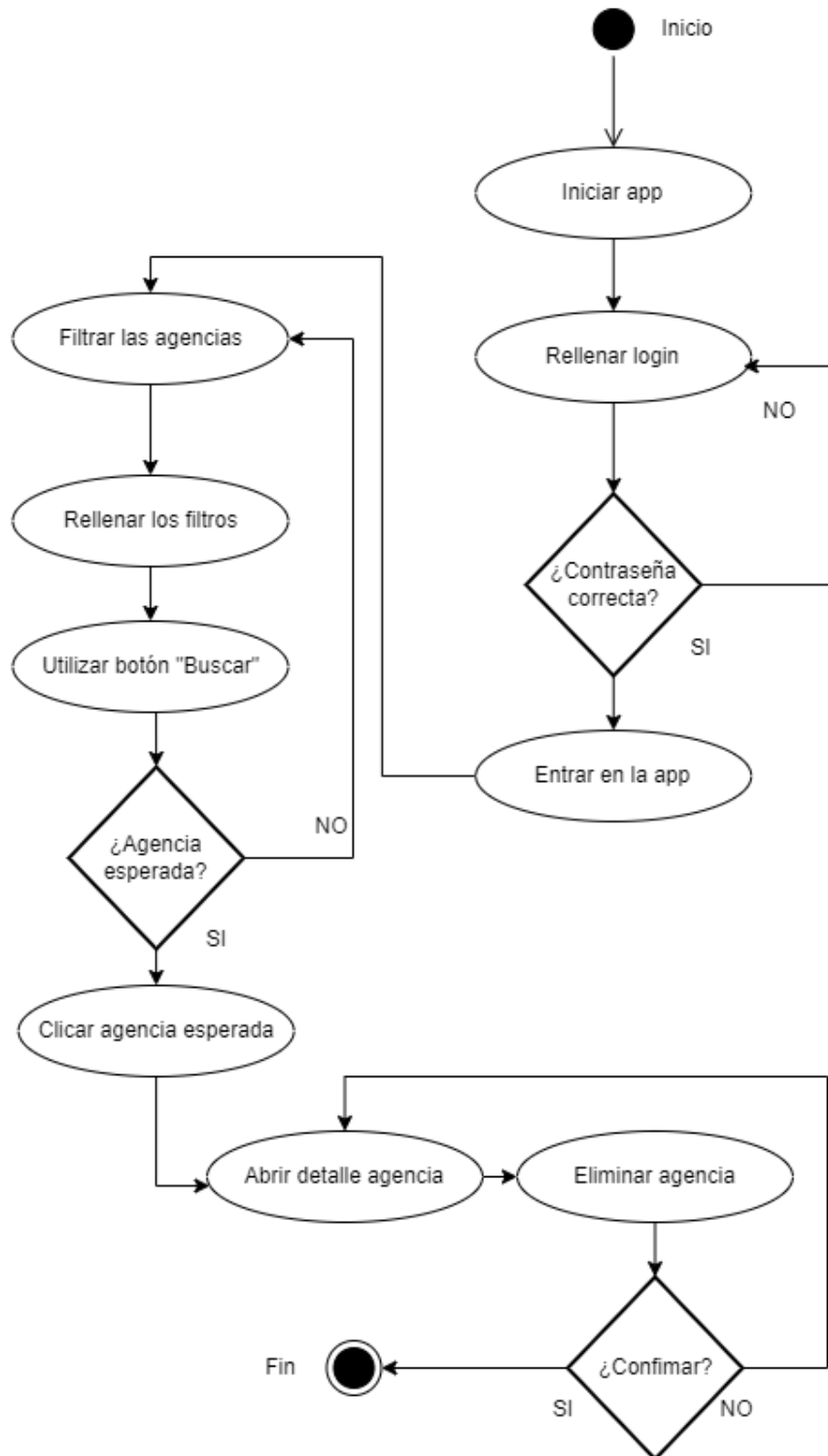


- **Crear agencia**



- Dar de baja una agencia

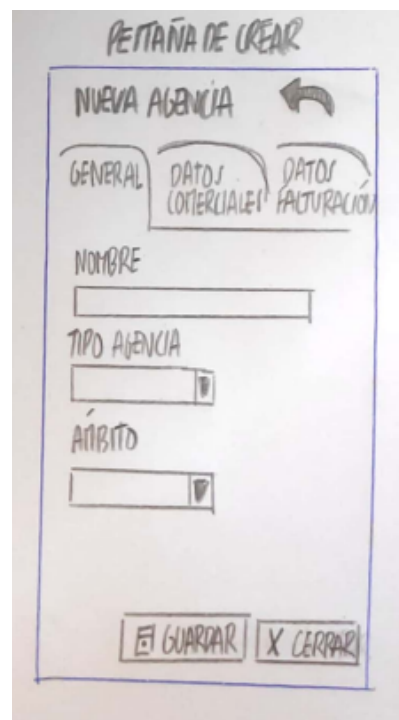
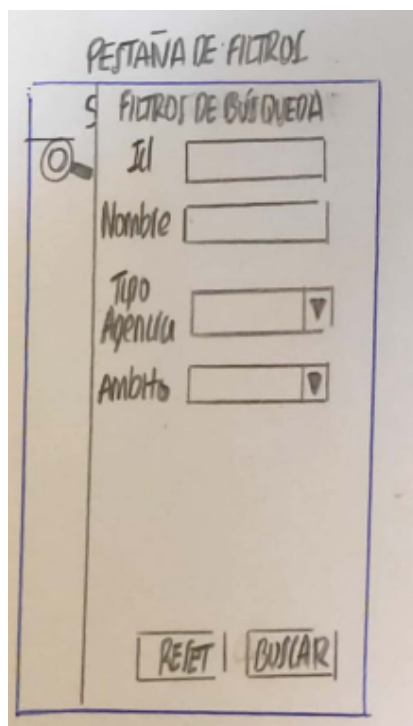
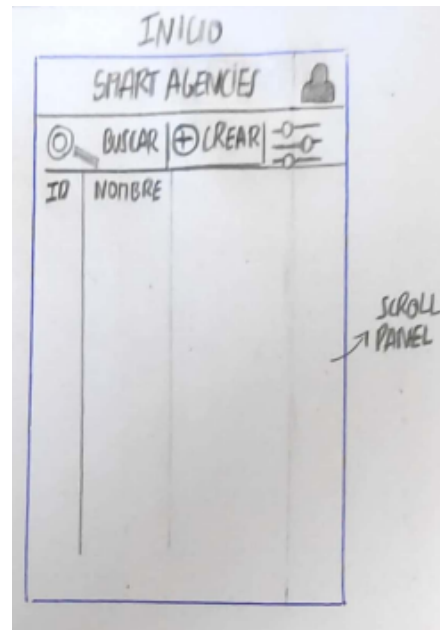
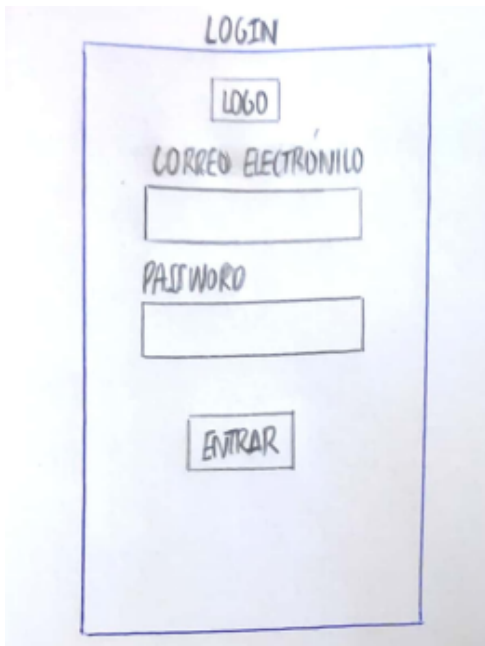
DAR DE BAJA UNA AGENCIA



Anexo II: Sketches escaneados DCU

Estos son los diseños de algunas pantallas hechos a mano alzada. Recogeremos feedback por parte de algunos usuarios que realizaron las entrevistas con nosotros para mejorar los diseños y la experiencia de usuario.

Se ha explicado que son prototipos para que se hagan una idea de cómo va a ser la aplicación a nivel visual.



CREAR AGENCIA

NUEVA AGENCIA ←

GENERAL
DATOS (COMERCIALES)
DATOS FACTURACIÓN

NOMBRE

TIP ¿DEBE CREAR LA AGENCIA?

SI
 NO

ÁMBITO

CUENTE

BÚSCA DE AGENCIA

SMART AGENCIES

BÚSQUEDA
 CREAR

ID	NOMBRE
10743	PRUEBA
1801	PRUEBA
2	PRUEBA
193	PRUEBA
⋮	

* SE PUEDE HAZER DOBLE CLICK

DETALLE AGENCIA

193 - PRUEBA

GENERAL
DATOS (COMERCIALES)
DATOS FISCALES

CALLE

NÚMERO

PAIS

COMERCIAL

⋮

DAR DE BAJA AGENCIA

193 - PRUEBA

GENERAL
DATOS (COMERCIALES)
DATOS FACTURACIÓN

NOMBRE

TIP ¿DEBE DAR DE BAJA LA AGENCIA?

SI
 NO

ÁMBITO

CUENTE

Muchos de ellos comentan que es una interfaz sencilla e intuitiva, algo positivo porque es lo que estaba buscando. Además, tiene bastante similitud a la aplicación de escritorio, por lo que hace que la nueva aplicación resulte familiar y no sea tan agresivo adaptarse a la aplicación

Anexo III: Explicaciones con soluciones DCU

Se ha tenido en cuenta los diseños que hay en la aplicación de Smart Agencies Desktop para que los usuarios no tengan un cambio tan brusco de una aplicación a otra. También se ha respetado la marca y el color que define a la empresa ya que era una de las principales preocupaciones que nos transmitieron al empezar el proyecto, es por eso que el color del prototipo es verde.

Se ha seguido el diseño occidental, es decir, los botones están colocados de arriba a abajo y de izquierda a derecha ya que es el nicho de mercado de donde son los trabajadores de nuestro cliente.

Se ha optado por mostrar un mensaje cada que vez que se hacia una acción de guardar en la aplicación como crear, modificar o dar de baja una agencia, para mantener al usuario informado y de que todo está funcionando correctamente. Esto es una funcionalidad que se ha implementado en la aplicación para dispositivos móviles ya que en la aplicación de escritorio no está disponible.

Anexo IV: Documento de preguntas DCU

Esta documentación proporciona una plantilla para realizar entrevistas a usuarios de Smart Agencies. Las entrevistas se utilizan para recopilar información valiosa sobre la experiencia del usuario, los puntos fuertes y débiles de la aplicación, y las posibles áreas de mejora.

Los objetivos de estas entrevistas son:

- Comprender las necesidades y expectativas de los usuarios.
- Identificar los problemas y desafíos que los usuarios encuentran al utilizar la aplicación.
- Obtener comentarios sobre las características y el diseño de la aplicación.

[Plantilla de entrevistas](#)

Anexo V: Encuesta y resultados sobre el MVP

Este apartado resume los resultados de una encuesta dirigida a usuarios finales de la aplicación Smart Agencies, diseñada para la gestión inteligente de agencias. Este análisis se enmarca en nuestro compromiso por mejorar continuamente nuestras soluciones tecnológicas. Agradecemos la participación activa de los usuarios, cuyas perspectivas y comentarios informarán las decisiones orientadas a optimizar Smart Agencies. Esta memoria presenta de manera confidencial los resultados y análisis obtenidos, con la convicción de que contribuirán a fortalecer nuestra capacidad para ofrecer soluciones tecnológicas avanzadas que impulsen el éxito y la eficiencia de las agencias.

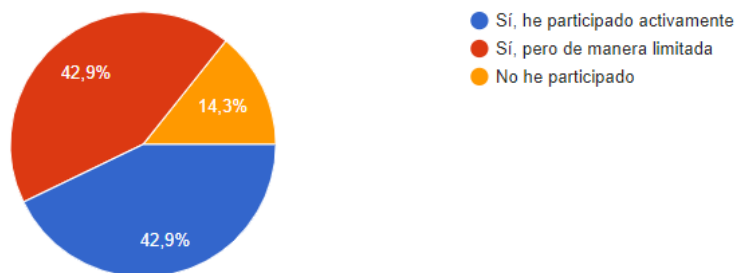
Estas han sido las preguntas y los resultados de la encuesta.

Participación en la Toma de Requisitos

 Copiar

¿Has participado activamente en el proceso de toma de requisitos durante el desarrollo de Smart Agencies App?

7 respuestas

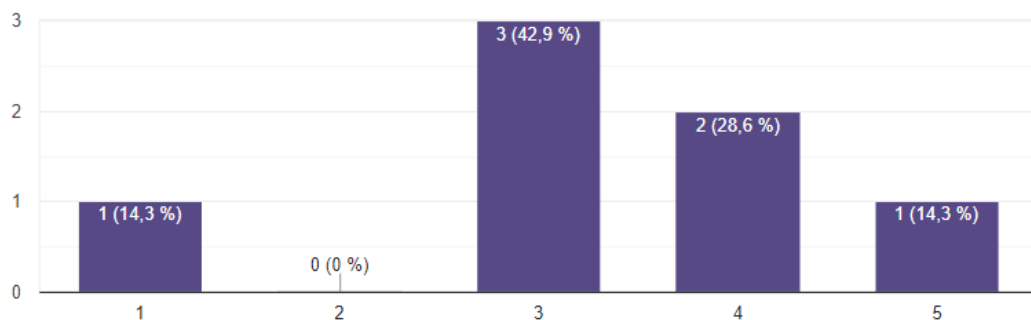


Cumplimiento de Expectativas

 Copiar

En comparación con tus expectativas iniciales, ¿consideras que Smart Agencies App ha cumplido adecuadamente con tus necesidades y requerimientos?

7 respuestas

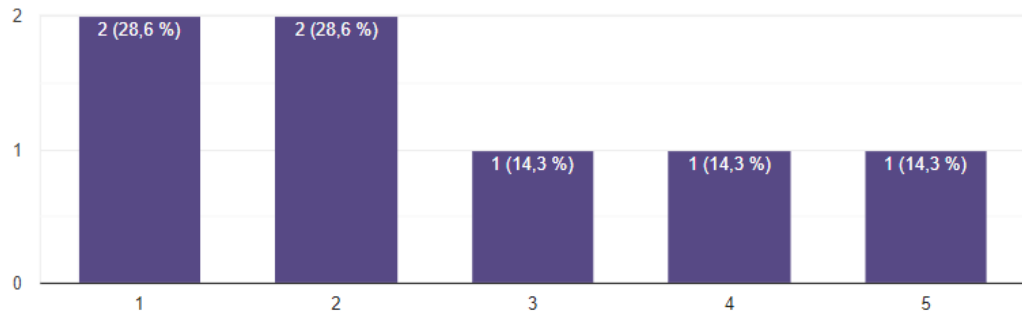


Facilidad de Uso

 Copiar

¿Qué tan fácil fue para ti aprender a usar Smart Agencies App?

7 respuestas

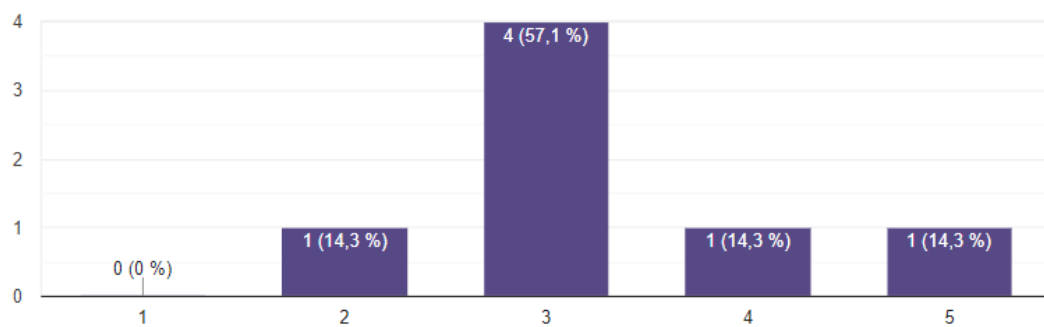


Rendimiento y Velocidad

 Copiar

En términos de rendimiento y velocidad, ¿Cómo calificarías Smart Agencies App?

7 respuestas

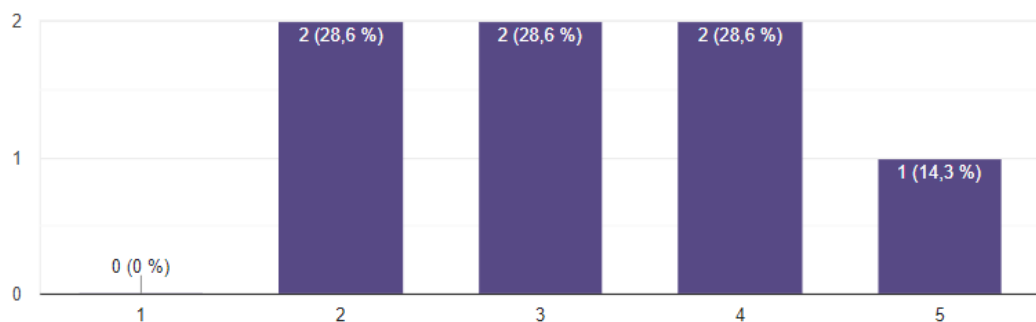


Funcionalidades

 Copiar

¿Estás satisfecho con las funcionalidades ofrecidas por nuestra aplicación?

7 respuestas

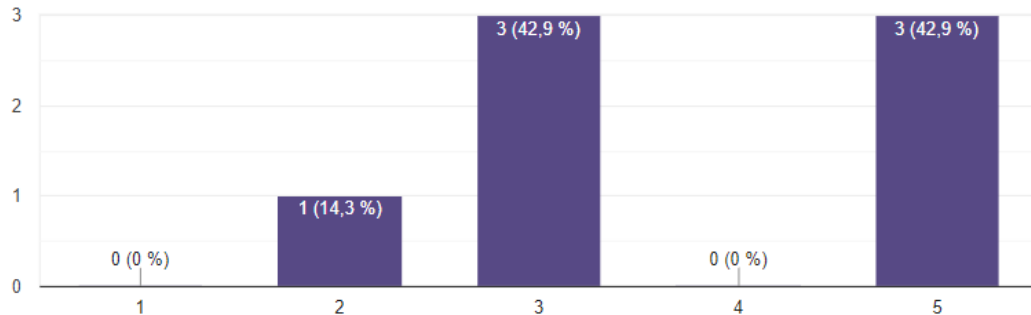


Documentación y Tutoriales

 Copiar

¿Cómo calificarías la disponibilidad y utilidad de la documentación y tutoriales para la versión de Smart Agencies App?

7 respuestas



¿Hay algún comentario adicional que te gustaría compartir sobre tu experiencia con nuestra aplicación?

2 respuestas

Cambiaría el color principal de la aplicación

Pienso que la aplicación podría ser un poco más moderna

Para finalizar, remarcar la positiva recepción de Smart Agencies es motivo de satisfacción, reflejando su eficacia en satisfacer las necesidades de las agencias. Aunque celebramos este éxito, estamos comprometidos con la mejora continua, abordando las áreas identificadas por los usuarios en futuros desarrollos. Agradecemos la valiosa retroalimentación y colaboración de los participantes, elementos clave para seguir elevando el estándar de Smart Agencies en el ámbito empresarial.