

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**USO DE DEVOPS Y SCRUM PARA EL DESARROLLO DE UNA
APLICACIÓN WEB PWA PARA CITAS Y REUNIONES DE LA
EMPRESA MANTICORE LABS**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS INFORMÁTICOS Y DE
COMPUTACIÓN**

Giler Salavarría Mario Javier

DIRECTORA:

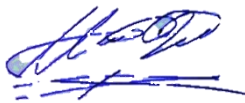
Dra. Pamela Catherine Flores Naranjo

Quito, enero 2023

DECLARACIÓN

Yo, Mario Javier Giler Salavarría declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cede mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.



Mario Javier Giler Salavarría

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Mario Javier Giler Salavarría bajo mi supervisión.

A handwritten signature in blue ink, appearing to read 'Pamela Flores', enclosed within a circular flourish.

PAMELA CATHERINE FLORES NARANJO, Ph.D.
DIRECTOR DE PROYECTO

DEDICATORIA

Este trabajo de titulación lo dedica primero a Dios por darme las fuerzas necesarias en cada paso de mi ciclo de aprendizaje.

De igual manera, todo el esfuerzo y la perseverancia a mis padres por ser el pilar fundamental de todo el crecimiento y la insistencia de continuar adelante ante cualquier adversidad.

A mí hermano por tener las palabras exactas ante los momentos duros y ser un ejemplo guía ante situaciones difíciles.

A mis profesores por brindarme los conocimientos que hoy en día aplico en mi vida profesional.

A mis tutores por el asesoramiento constante en la realización del proyecto.

AGRADECIMIENTOS

Estoy y estaré eternamente agradecido con mi Madre por todo el apoyo y amor incondicional que tiene hacia mí.

Agradezco a mi Padre por su sacrificio diario, para brindarme una buena educación.

Les agradezco a mí hermano por su apoyo y su amor cuando los he necesitado.

Agradezco a mis abuelitos por ser mis segundos padres y madres.

Agradezco a mis profesores por ejercer la profesión más inspiradora que es la educación y haberme proporcionado de los conocimientos necesarios para mi vida profesional.

Agradezco a mis tutores por el siempre estar disponibles para poder mejorar e identificar las mejores que se puede realizar en mis proyecto de titulación

Todos mis agradecimientos a la mejor institución, quien me ha nutrido de conocimientos y disciplina la Escuela Politécnica Nacional la mejor universidad del Ecuador, de la cual me siento muy orgulloso de ser estudiante.

ÍNDICES DE CONTENIDO

| | |
|---|----|
| RESUMEN..... | 1 |
| ABSTRACT | 3 |
| CAPÍTULO 1. INTRODUCCIÓN | 3 |
| 1.1 ANTECEDENTES..... | 3 |
| 1.2 PLANTEAMIENTO DEL PROBLEMA..... | 4 |
| 1.3 OBJETIVOS | 5 |
| OBJETIVO GENERAL | 5 |
| OBJETIVOS ESPECÍFICOS | 5 |
| CAPÍTULO 2. MARCO TEÓRICO..... | 6 |
| 2.1 Características de DevOps | 6 |
| • Componentes de DevOps..... | 7 |
| • Roles de DevOps..... | 7 |
| • Ciclo de desarrollo | 8 |
| • Métricas para DevOps | 8 |
| • Características de DevOps | 9 |
| 2.2 Scrum | 9 |
| • Aplicación web..... | 12 |
| • Aplicación Web Progresiva (PWA)..... | 14 |
| CAPÍTULO 3. FASE DE EXPLORACIÓN | 16 |
| 3.1 Planificación con los stakeholders | 16 |
| 3.2 Recolección de requerimientos..... | 16 |
| 3.2.1 Encuesta dirigida a los trabajadores de la empresa..... | 16 |
| 3.3 Roles de SCRUM | 21 |
| CAPÍTULO 4. FASE DE INICIALIZACIÓN | 22 |
| 4.1 Lenguajes de programación | 22 |
| 4.2 Herramientas de desarrollo..... | 22 |
| 4.3 Frameworks y Librerías | 23 |

| | | |
|--|------------------------------------|----|
| 4.4 | Configuración del proyecto | 24 |
| 4.4.1 | Formato épicas de usuario | 24 |
| 4.4.2 | Historias épicas de usuario..... | 26 |
| 4.5 | Sprint 0 | 30 |
| CAPÍTULO 5. FASE DE EJECUCIÓN | | 38 |
| 5.1 | Sprint 1 | 38 |
| | • Objetivo del Sprint | 38 |
| | • Sprint Backlog | 38 |
| | • Ejecución del Sprint..... | 40 |
| | • Revisión del Sprint..... | 49 |
| 5.2 | Sprint 2 | 50 |
| | • Objetivo del Sprint | 50 |
| | • Sprint Backlog | 50 |
| | • Ejecución del Sprint..... | 53 |
| | • Revisión del Sprint..... | 56 |
| 5.3 | Sprint 3..... | 58 |
| | • Objetivo de Sprint | 58 |
| | • Sprint Backlog | 58 |
| | • Ejecución del Sprint..... | 62 |
| | • Revisión del Sprint..... | 65 |
| CAPÍTULO 6. ANALISIS DE FUNCIONALIDAD Y USABILIDAD | | 67 |
| 6.1 | Prueba de usabilidad | 67 |
| 6.1.1 | Resultados de la encuesta SUS..... | 68 |
| 6.2 | Prueba de funcionalidad | 69 |
| CAPÍTULO 7. CONCLUSIONES Y RECOMENDACIONES..... | | 73 |
| | CONCLUSIONES | 73 |
| | RECOMENDACIONES | 74 |
| REFERENCIAS BIBLIOGRÁFICAS..... | | 75 |
| ANEXOS..... | | 78 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1: Roles de SCRUM | 21 |
| Tabla 2: Lenguajes..... | 22 |
| Tabla 3: Herramientas..... | 22 |
| Tabla 4: Frameworks y Librerías | 23 |
| Tabla 5: Modelo para historias de usuario..... | 25 |
| Tabla 6: Épicas de la aplicación | 26 |
| Tabla 7: Requerimientos funcionales | 26 |
| Tabla 8: Requerimientos no funcionales..... | 27 |
| Tabla 9: Tabla de prioridades..... | 28 |
| Tabla 10: Historias de usuario ordenado por épicas..... | 28 |
| Tabla 11: Validar Datos..... | 38 |
| Tabla 12: Validar fotografía | 39 |
| Tabla 13: Validar correo | 39 |
| Tabla 14: Creación de cuenta del prestador de servicios | 51 |
| Tabla 15: Verificar correo | 51 |
| Tabla 16: Iniciar sesión | 51 |
| Tabla 17: Consultar datos y reseñas de sus servicios..... | 52 |
| Tabla 18: Ingresara nuevos servicios | 52 |
| Tabla 19: Recepción de notificación..... | 52 |
| Tabla 20: Interactuar con el cliente..... | 52 |
| Tabla 21: Crear cuenta..... | 59 |
| Tabla 22: Verificar correo | 59 |
| Tabla 23: Iniciar sesión | 59 |
| Tabla 24: Calificar contratista | 59 |
| Tabla 25: Consultar sus datos personales..... | 60 |
| Tabla 26: Recepción de notificación..... | 60 |
| Tabla 27: Consultar servicios que existen en la aplicación | 61 |
| Tabla 28: Contratar servicio | 61 |
| Tabla 29: Preguntas de cuestionario de usabilidad | 67 |
| Tabla 30: Resultados de la Encuesta SUS..... | 68 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1: DevOps..... | 6 |
| Figura 2: Esquema de DevOps (Desarrollador - Operador). | 7 |
| Figura 3: Ciclo de desarrollo ágil..... | 11 |
| Figura 4: Pregunta 1 | 16 |
| Figura 5: Pregunta 2 | 17 |
| Figura 6: Pregunta 3 | 18 |
| Figura 7: Pregunta 4 | 19 |
| Figura 8: Pregunta 5 | 19 |
| Figura 9: Pregunta 6 | 20 |
| Figura 10: Pregunta 7 | 21 |
| Figura 11: Historias de usuario desde Trello | 25 |
| Figura 12: Diagrama de paquete UML | 31 |
| Figura 13: Arquitectura del proyecto | 31 |
| Figura 14: Entidad Componente Sistema..... | 32 |
| Figura 15: Fase de fuentes. | 33 |
| Figura 16: Configuración de conexión GitHub..... | 33 |
| Figura 17: Fase de actualización del commit. | 34 |
| Figura 18: Fase de aprobación del commit | 34 |
| Figura 19: Archivo de construcción de comandos para la fase del build. | 35 |
| Figura 20: Fase de construcción y de prueba. | 35 |
| Figura 21: Archivo de despliegue de comandos para la fase del build. | 36 |
| Figura 22: Fase de despliegue..... | 36 |
| Figura 23: Ejecución pipeline para el despliegue la aplicación..... | 37 |
| Figura 24: Sprint Backlog 1..... | 38 |
| Figura 25: Diseño de login. | 40 |
| Figura 26: Registro de usuarios. | 41 |
| Figura 27: Código de validación de cuenta por correo. | 41 |
| Figura 28: Validación de cuenta por correo..... | 42 |
| Figura 29: Servicios disponibles. | 42 |
| Figura 30: Solicitar servicio. | 43 |
| Figura 31: Reseñas de usuarios. | 43 |
| Figura 32: Valor de servicio. | 44 |
| Figura 33: Notificación de servicio. | 44 |
| Figura 34: Chat de la aplicación..... | 45 |
| Figura 35: Función que valida el login..... | 46 |

| | |
|---|----|
| Figura 36: Diseño del html | 46 |
| Figura 37: Validación de campos requeridos. | 47 |
| Figura 38: Código de Notificación. | 47 |
| Figura 39: Interfaz gráfica del login. | 48 |
| Figura 40: Validación por correo electrónico. | 48 |
| Figura 41: Revisión del sprint 1..... | 49 |
| Figura 42: Análisis de la finalización del Sprint 1. | 49 |
| Figura 43: Sprint Backlog 2..... | 50 |
| Figura 44: Ejecución de sprint..... | 53 |
| Figura 45: Interfaz gráfica de login y perfil. | 54 |
| Figura 46: Interfaz gráfica de notificaciones y google maps. | 55 |
| Figura 47: Interfaz gráfica del chat..... | 56 |
| Figura 48: Revisión del Sprint 2. | 57 |
| Figura 49: Análisis de la finalización del Sprint 2. | 57 |
| Figura 50: Sprint Backlog 3..... | 58 |
| Figura 51: Código para validar la imagen..... | 62 |
| Figura 52: Interfaz gráfica de la navegación por la aplicación móvil..... | 63 |
| Figura 53: Interfaz gráfica para solicitud que se ingresa para pedir el servicio..... | 64 |
| Figura 54: Interfaz gráfica del chat de la aplicación móvil. | 64 |
| Figura 55: Revisión del Sprint 3. | 65 |
| Figura 56: Análisis de la finalización del Sprint 3. | 66 |
| Figura 57: Prueba de funcionalidad 1.1 | 69 |
| Figura 58: Prueba de funcionalidad 1.2 | 70 |
| Figura 59: Prueba de funcionalidad 2 | 70 |
| Figura 60: Prueba de funcionalidad 3 | 71 |
| Figura 61: Resultado de pruebas realizadas..... | 71 |
| Figura 62: Evolución de pruebas. | 72 |

RESUMEN

Manticore-Labs es una empresa que tiene como aspiración expandir los servicios de tecnología de la información en el país, como una empresa competitiva, confiable, con la mejor capacitación y ofreciendo sus productos con los mayores estándares de calidad.

La empresa Manticore-Labs en la actualidad realiza la gestión de servicios de manera manual. Poseen un archivo donde registran la solicitud requerida por uno de sus clientes y coordinan los costos junto con los puntos de encuentro por mensajería de texto como whatsapp, telegram, entre otras. A la vez, la retroalimentación para sus clientes y los trabajadores que realizan la labor no existe para detectar puntos de mejora. De modo que, se propone para la empresa la elaboración de una aplicación web progresiva para la gestión de reuniones y citas en su catálogo de servicios. Esta aplicación ofrece administrar a sus empleados y clientes la información de los servicios que ejercen junto con la solicitud y búsqueda de cada uno. Además, de brindarles funcionalidades para realizar acciones de retroalimentación y observar los puntos de mejora.

Para el desarrollo de la aplicación se eligió el framework Ionic. Los entregables se realizaron mediante técnicas de despliegue continuo con DevOps con el propósito de que sean mantenibles en el tiempo, junto al marco de trabajo SCRUM permitiendo adaptabilidad, transparencia e inspección en el proyecto.

El presente trabajo fue realizado con metodología Scrum, conformada por 3 Sprints, evitando la sobrecarga de trabajo para el equipo de desarrollo. En el primer sprint se analizó la arquitectura del proyecto, se creó un pipeline, se instanció la base de datos, además de desarrollar los mockups. Para el segundo sprint se desarrolló del back-end junto con front-end. Finalmente, para el tercer sprint se culminó con el desarrollo del front-end y del back-end.

Finalmente, se realizó las pruebas de usabilidad para comprobar el correcto funcionamiento de la aplicación mediante una encuesta de SUS que fue aplicada a los 10 empleado de la empresa, obtenido un resultado de usabilidad de 75/100 el mismo que demuestra que la aplicación móvil es aceptable ya que el mínimo recomendado por los encuestadores SUS es de 70/100.

Palabras clave: Aplicación móvil, Frameworks, DevOps, Scrum, SUS, usabilidad

ABSTRACT

In the first part of the document for the development of this project a bibliographic and investigative research was conducted on the different topics such as mobile applications, features, advantages, disadvantages and utilities that will have in the company, and also investigated on the Frameworks for the development of mobile applications, choosing the Ionic framework to develop the mobile application since it was the one that best suited our needs, in this way the necessary information was collected on the topics to carry out the development of the project in the company Manticore-Labs.

This research work was carried out using the Scrum methodology, which consists of 5 phases that must be fully complied with so that the development of the project does not have complications, likewise for data collection a survey was applied where each of the answers obtained was analyzed in order to support this project.

For the development of the mobile application was divided into 3 Sprints which contained tasks that should be performed in each of them and thus carry a systematized and orderly work thus fulfilling all tasks that allow the proper functioning of the application, likewise with this division of the project by Sprints avoided the overload of work for the development team.

Finally, the usability tests were carried out to verify the correct functioning of the application through a SUS survey that was applied to the 10 employees of the company, obtaining a usability result of 75/100, which shows that the mobile application is acceptable since the minimum recommended by the SUS surveyors is 70/100.

Key words: Mobile application, Frameworks, DevOps, Scrum, SUS, usability.

CAPÍTULO 1. INTRODUCCIÓN

1.1 ANTECEDENTES

Manticore Labs una empresa dedicada a brindar capacitaciones, desarrollo de aplicaciones web, móviles, de escritorio, consultoría según el modelo de negocio. Sus productos son diseñados bajo estándares de calidad con últimas tecnologías. Además, sus procesos incluyen investigaciones tecnológicas para el software con la finalidad de que no queden obsoleto [1].

Entre sus ventajas ofrece seguridad y confianza a través de las estrategias usadas en los bancos, a partir de un Login hasta tokens con tiempo, dispositivos determinados, ip. A continuación, se presenta sus clientes potenciales [2]:

- Alphaside Alpha technologies (Desarrollo de aplicaciones en Java y Angular)
- Cooperativa Oscus (Levantamiento de información, prototipos y diseño de arquitectura)
- Oriente Seguros (Desarrollo de apps en Java y Angular para procesos de gestión y arquitectura)
- Espe Innovativa (Seguridad en software)
- Ejército ecuatoriano (Apps para inventarios, con niveles de seguridad y definición de arquitectura)
- Dental sonris (aplicaciones para servicios de odontologías que permiten llevar el inventario en tiempo real)
- Bagó (Servicio de outsourcing mediante sistemas privados)
- Fae (Configuración de la red VPN, gestión para el manejo de educación virtual)
- Sande Ecuador (Aplicación para ventas, buqueteria y control de inventarios)
- TVentas (Aplicaciones con botón de pagos y recepción de datos)
- EP Petroecuador (Aplicación para control de inventarios, así como para los vendedores desde diferentes puntos de venta)
- Diners Club (Capacitaciones de seguridad web con Owasp para talleres prácticos y chatbots)
- EPMAPS (Capacitaciones teóricas y prácticas paramseguridad web con Owasp)
- Ruta Iglesias (Aplicaciones para pagos e inscripción a la ruta de iglesias)
- Ans Arcotel (Software para gestionar los descuentos en pagos de socios)

- Escuela Politécnica Nacional (Capacitación a docentes en apps web y móviles, javascript, Python y otros)
- Alitas del Cadillac (Apps para pedidos y gestión de despacho)
- Facultad de Ingeniería en Sistemas (capacitaciones en desarrollo de software para doctorantes)
- Greenetics (Capacitaciones para seguridad en la web con Owasp)
- Campus sustentable EPN (Software que permite la reciclación sustentable)
- Universidad Andina (Ayuda en proceso de auditoría para el proceso de elecciones en la universidad)
- EPN Tech (Software para la gestión de proyectos)
- Uribe & Schwarzkopf (Software para la facturación electrónica)
- Servicio de Rentas Internas (Accesibilidad, mejoramiento de experiencia y reinención de las apps)

1.2 PLANTEAMIENTO DEL PROBLEMA

Según, la Agencia de Regulación y Control de las Telecomunicaciones (ARCOTEL), publicó una infografía con datos sobre los abonados del Servicio Móvil Avanzado – SMA y de acceso a internet, donde existen 8,1 millones de cuentas de internet móvil, lo que representa que un 6,4% de usuarios del Servicio Móvil Avanzado poseen un teléfono inteligente [3]. Por lo que, en la actualidad existe una alta demanda en el desarrollo de aplicaciones móviles que en la mayoría no son usadas por todos los usuarios, lo que representa un enorme problema que afrontan las aplicaciones móviles actualmente. De modo que, la preferencia de usar aplicaciones vinculadas más con las redes sociales, tales como Facebook, WhatsApp, Instagram, Twitter originan que la sociedad adquiera dispositivos móviles.

Manticore-Labs es una empresa dedicada al desarrollo web y móvil. Dentro de las aplicaciones que ellos ofertan al mercado, las empresas o clientes que deciden optar por sus servicios necesitan establecer servicios y ofrecerlos al público; para este tipo de propósitos, existen una gran variedad de aplicaciones que solucionan este servicio.

A pesar de esto, las empresas y clientes de Manticore-Labs prefieren una solución dentro del sistema que ya están utilizando. Algunas características especiales del aplicativo de gestión es que se instale la aplicación en los celulares con el uso de una aplicación web progresiva. Gestionar los estados de las reuniones, los costos en caso

de ser un servicio ofertado a clientes externos, y los recursos necesarios (infocus, hojas, esferos y demás) para los clientes internos, sin olvidar las fechas y horarios en tiempo real.

Por lo tanto, la solución para la empresa Manticore Labs es la elaboración de una aplicación híbrida cuyos entregables sean mantenibles en el tiempo y utilicen técnicas de despliegue continuo con DevOps. Además, sus funcionalidades tanto en web como en móvil deberían ser similares, gestionando eficazmente sus citas y reuniones con el catálogo de clientes que poseen.

1.3 OBJETIVOS

OBJETIVO GENERAL

Emplear DevOps y SCRUM para el desarrollo de una aplicación web PWA para citas y reuniones de la empresa Manticore Labs.

OBJETIVOS ESPECÍFICOS

- Definir requerimientos funcionales y no funcionales.
- Diseñar prototipos del aplicativo.
- Implementar los requerimientos en un número determinado de sprints.
- Ejecutar las tareas de cada sprint.
- Implementar la aplicación móvil y web en función de una PWA.
- Ejecutar pruebas de calidad de software y de uso del aplicativo.

CAPÍTULO 2. MARCO TEÓRICO

El marco teórico que fundamenta este documento proporcionará al lector una idea más clara de las herramientas, conceptos y tecnologías que se utilizaron para el desarrollo del aplicativo.

2.1 Características de DevOps

Se define como un marco de trabajo y filosofía en evolución para mejorar el desarrollo de aplicaciones en menor tiempo para nuevas publicaciones o funciones de productos para los clientes [4].

Además, es un movimiento ágil, el cual se centra en la comunicación y colaboración del equipo de desarrollo y de operación, con la meta que es colaborar.



Figura 1: DevOps.
Fuente: [5]

El objetivo de los DevOps es mejorar los tiempos de respuesta con relación a los requerimientos y necesidades del usuario a través de aplicaciones y medios que permitan alinear y finalizar las brechas entre los operadores y desarrolladores, usando prácticas ágiles en áreas de operativas [6].



Figura 2: Esquema de DevOps (Desarrollador - Operador).
Fuente: [7]

- **Componentes de DevOps**

Entre los componentes se menciona [8]:

- De gestión de desarrollo de software
- De gestión de data center
- Herramientas con ayuda de desarrollo y despliegue a la gestión de desarrollo de software.

- **Roles de DevOps**

Esta guía se basa en los roles que presenta la metodología Scrum, sin embargo, para realizar las actividades colaborativas se incluye a operadores que forman parte del equipo de desarrollo [9]:

- Scrum Máster
- Equipo de desarrollo
- Operadores
- Dueño del producto
- Backlog del producto

- **Ciclo de desarrollo**

DevOps se centra en el ciclo de vida que maneja Scrum, considerando lo siguiente [10]:

- Manejo de iteraciones cortas, es decir de 2 a 4 semanas por cada iteración.
- Busca la satisfacción del usuario, en base a los principios de DevOps.
- Entregas parciales según el producto automatizado y en colaboración con el equipo de desarrollo.

Respecto a las iteraciones, estas deben ser relacionadas con las métricas que maneja DevOps, en este caso sería las medidas cuantitativas según el atributo.

Los atributos de las métricas se dan en base al protocolo de los operadores y directivos con el fin de medir la gestión en un determinado tiempo.

Y, el marco colaborativo hace referencia al paradigma según la medida de rendimiento.

- **Métricas para DevOps**

Según Gartner define las métricas para ambiente:

- **Métrica de colaboración:** Referente al desarrollo y operaciones realizadas por el grupo de trabajo con el fin de entregar un producto en el tiempo establecido, el mismo que es corto cumpliendo la necesidad del usuario.
- **Métrica de tiempo de ciclo:** Permite medir el nivel de eficiencia desde el inicio de la iteración hasta el momento de su despliegue.
- **Métrica de frecuencia:** Muestra el tiempo repetitivo en el que se debe proporcionar la iteración.
- **Métrica de satisfacción al usuario:** Permite medir el nivel de satisfacción del usuario con referencia a la eficiencia de las aplicaciones.

Además, DevOps permite manejar los principios desde la estructura colaborativa, es decir de la etapa del desarrollo de las aplicaciones. Entre los principios que se basa se muestra [11]:

- **Integración continua:** Manera en la que el equipo de desarrollo maneja el trabajo, total o parcial en un tiempo establecido por los mismos. Además, en este principio

se usa herramientas que permitan automatizar el producto, mismas que integren continuamente las partes del código una vez que sean validadas a través de las pruebas automáticas.

- **Despliegue continuo:** Permite configurar el entorno de trabajo, de manera que sea efectivo por los usuarios. Se inicia con una configuración previa en el desarrollo, así como, una configuración antes que la versión sea terminada, de esta manera se comprueba los fallos existentes.
- **Entrega continua:** Hace referencia al despliegue de la versión del producto, en el cual se considera los requerimientos de la empresa para ser usados por los usuarios.

- **Características de DevOps**

Los DevOps se caracterizan por [12]:

- Trabajar con desarrolladores que estén en continua colaboración a través de sus herramientas compartidas.
- Permite corregir las desalineaciones sean de las personas u objetivos de manera que permita crear vínculos entre las operaciones y desarrolladores.
- Ayuda a entregar de manera continua el software debido a la ejecución de pruebas y supervisión de los entornos de trabajo.
- Con el uso de las herramientas la gestión de lanzamientos se torna fácil respecto a la virtualización, supervisión, inclusión, etc.

2.2 Scrum

Se define como un proceso para que el desarrollo de software sea creciente e iterativo, ya que, se considera como un framework estructurado por sprints. En los cuales, el equipo presenta una lista de requerimientos para los clientes llamados historias de usuario con el fin de dar valor a los clientes. Al finalizar el sprint se brinda un producto que sea distribuirle y comercializable [13].

Entre las características de scrum se mencionan [14]:

- ❖ Facilidad para adaptarse y flexible.
- ❖ Gestión regular frente a las necesidades del cliente.
- ❖ Resultados anticipados.
- ❖ Productividad y calidad.
- ❖ Retorno de inversión.
- ❖ Equipo motivado.
- ❖ Migración de riesgos.
- ❖ Alineamiento entre el equipo y el cliente.

Al aplicar la metodología ágil para desarrollar un software, es necesario considerar los roles asignados para el equipo de trabajo, entre ellos [15]:

- ✓ **Scrum Master:** Se encarga de coordinar a los integrantes del equipo y asignación de tareas a los mismos, con el fin de mantener al grupo organizado.
- ✓ **Product Owner:** Referente al grupo de interés, es decir a quienes va dirigido el proyecto desarrollado. Además, son aquellos que expresan lo que se desea hacer y los objetivos.
 - Existen casos en los cuales, se puede nombrar a representantes que no formen parte del equipo, mismo que debe encargarse de priorizar los intereses y diferentes puntos de vista.
- ✓ **Scrum Team:** Aquellos responsables a realizar las diferentes tareas. Se sugiere crear los equipos de trabajo menos de 10 integrantes con el fin de complementar de manera adecuada al grupo de forma que cada uno tenga los conocimientos necesarios para ejecutar las actividades asignadas.
- ✓ **Customers:** Denominados consumidores o usuarios quienes harán uso del producto, sin embargo, existen confusiones con los clientes, es decir la persona que paga y el consumidor quien usa el producto, aunque en ocasiones puede ser la misma persona.

➤ Fases

A continuación, se mencionan las fases de Scrum [16]:

1. **Concepto:** Se plantea las características del producto, así como la asignación del equipo que formará parte del desarrollo.
2. **Especulación:** En esta fase se coloca las disposiciones con la información recopilada, además se determinan los límites para el desarrollo del producto final, así como el tiempo y costos. El producto se realizará a partir de las ideas planteadas, partes interesadas e impacto.
Por otro lado, se considera observar los requisitos generales, lista de funcionalidades y planes de entrega.
3. **Exploración:** Se añaden las funcionalidades al producto.
4. **Revisión:** El producto es revisado por el equipo en base a lo que se tenga construido.
5. **Cierre:** El producto es entregado según la fecha acordada, como primera versión ya que, existe la posibilidad de modificar o realizar mejoras al mismo.

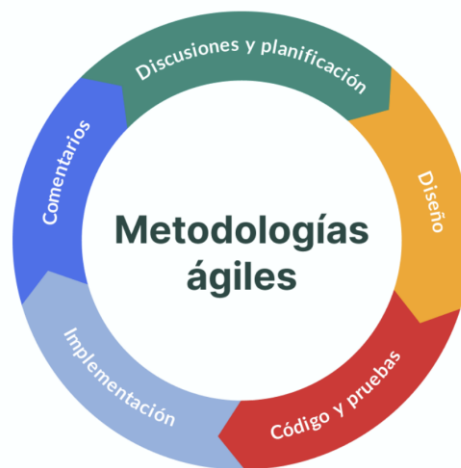


Figura 3: Ciclo de desarrollo ágil.
Fuente: [17]

➤ Componentes de Scrum

Entre los componentes que forman parte de la metodología Scrum se presenta [18]:

Reuniones: En este apartado se define 3 etapas importantes como la planificación del Backlog, seguimiento del sprint y revisión del sprint.

- ❖ **Planificación del Backlog:** Hace referencia al documento en el cual consta los requisitos del producto, se realizará desde la planificación del sprint 0.
- ❖ **Seguimiento del Sprint:** Se realizan reuniones diarias en las cuales se evalúa 3 preguntas principales: ¿Qué se realizó en la reunión pasada?, ¿Qué se realizará en la presente reunión?, ¿Qué inconvenientes existen y cómo se puede solucionar?
- ❖ **Revisión del Sprint:** Se observa el incremento generado por el Sprint, en el cual se presentarán todos los resultados con el fin de mejorar el feedback al cliente.

Elementos: Se presentan los elementos que forman parte de Scrum.

- ❖ **Producto Backlog:** Referente a las necesidades de los clientes.
- ❖ **Sprint Backlog:** Referente a las tareas que se hacen en los Sprint.
- ❖ **Incremento:** Se refiere al desarrollo del Sprint, es decir la parte finalizada y la operativa.

Historias de Usuario: De acuerdo con las funciones que va a tener el software. Estas serán el resultado de la comunicación y colaboración del equipo de trabajo y cliente, las mismas se componen de las 3 C:

- ❖ **Card:** Se describe las notas recordatorias.
- ❖ **Conversation:** Comunicación para entender y concretar lo planificado.
- ❖ **Confirmation:** Se establece un test funcional, en el cual se plantean los detalles y límites del producto [19].

- **Aplicación web**

Se definen como un programa informático que se ejecuta a través del internet sin ser instalada en el computador ya que, solo se lo hace mediante un navegador por ser desarrollada con un lenguaje de programación HTML. Además, brinda varias ventajas para los usuarios como el

acceso a la información de forma rápida y sencilla, así mismo se puede guardar la información en la base de datos [20].

De esta manera, las aplicaciones web al ser ejecutadas por el navegador los datos son procesados y guardados en una red por medio del navegador.

- **Tipos de aplicaciones web**

Las aplicaciones web se clasifican en 5 tipos [21]

- ❖ **Estáticas:** Aquellas que presentan poca información con el fin de no generar o crear contenidos nuevos.
- ❖ **Dinámica:** Utilizan bases de datos para almacenar la información y contenidos, mismos que se actualizan cuando el usuario acceda al aplicativo web.
- ❖ **Tienda virtual:** Su desarrollo tiene más complejidad ya que se crean pasarelas con pago a través de PayPal, tarjetas de crédito, entre otras.
- ❖ **Portal web:** Permite acceder a varias categorías o secciones.
- ❖ **Aplicaciones web que permiten gestionar contenidos:** Se usan en proyectos que van actualizando su contenido de manera frecuente.

- **Ventajas**

Las aplicaciones web brindan al usuario diferentes ventajas, entre ellas [22]:

- ❖ **Ahorro de tiempo:** Se realizan las actividades de manera sencilla sin necesidad de instalarla en los dispositivos.
- ❖ **Compatibilidad:** No tiene necesidad de crear clientes con relación al sistema operativo que maneje.
- ❖ **No usan espacio en el disco duro:** Al ser web, no ocupan espacio en el disco duro.
- ❖ **Actualizaciones rápidas:** Al ser gestionada por el desarrollador, cuando accedemos a la aplicación siempre se visualiza la última versión.
- ❖ **Multiplataforma:** Se usa desde cualquier S.O. ya que se accede a través del navegador.
- ❖ **Portables:** Es independiente del computador en el cual se use ya se inicia mediante una página web con acceso a internet.
- ❖ **Colaboración:** Permite compartir los datos por diferentes usuarios.

- **Aplicación Web Progresiva (PWA)**

Las aplicaciones progresivas nacen como una opción simplificada para los desarrolladores al desear implementar un aplicativo. Además, se considera como un estándar moderno para el desarrollo web ya que, se manejan como aplicaciones híbridas, es decir para diferentes sistemas operativos [23].

PWA une las tecnologías web y aplicaciones nativas, mismas que no necesitan ser instaladas ya que, son usadas a través de un navegador. Entre sus ventajas se menciona [21]

- ❖ Carga rápida
- ❖ Envío de notificaciones
- ❖ Acceso directo a la pantalla principal
- ❖ Presenta un modo offline

Además, las PWA utilizan las características de los navegadores para brindar una experiencia más apegada a las aplicaciones nativas, de modo que, facilitan a los usuarios finales obtener de manera fácil las PWA sin necesidad de visitar el playstore o una tienda de aplicaciones.

- **Objetivos y características de PWA**

Entre los objetivos que las PWA persiguen se mencionan [24]:

- ❖ Proporcionar mayor rendimiento en los dispositivos móviles y que se cargue de manera instantánea.
- ❖ Brindar una interfaz parecida a una nativa.
- ❖ Envío de notificaciones como una aplicación nativa

Así mismo, se menciona las características que posee [25]:

- ❖ Son progresivas, es decir que permiten usarlas en cualquier navegador para varios usuarios.
- ❖ Son responsive, se puede adaptar a diferentes pantallas sean de dispositivo móvil, pc de escritorio, laptops, Tablet.
- ❖ Su funcionalidad puede ser sin conexión.

- ❖ Tiene la posibilidad de instalarlas en el dispositivo móvil.
- ❖ Se direcciona a las apps mediante su URL.

- **Servicios en la nube**

Se definen como infraestructuras, sistemas o plataformas que alojan proveedores externos, y los mismos están disponibles para los usuarios mediante el internet. Además, facilitan la transmisión de datos de los usuarios mediante el internet, es decir desde los clientes hasta los proveedores y viceversa. De esta manera, para tener acceso a los servicios de la nube solo se requiere una computadora, conexión a internet y un sistema operativo [26].

Entre las ventajas de estos servicios se menciona:

- ❖ **Costo mensual:** El precio es mínimo a comparación si la empresa desea comprar el equipo para obtener el servicio.
- ❖ **Facilidad de administración y uso:** No se necesita de un administrador en seguridad y redes para brindar el servicio a la entidad.
- ❖ **No se requiere de licencias:** Únicamente se paga el mes por el servicio otorgado.

- **Service Worker**

Se define como una secuencia de comandos que el navegador logra ejecutar en segundo plano, el mismo es separado de la página web, de manera que no sea necesario la interacción del usuario. Actualmente, se añaden diferentes funciones en forma de notificaciones sincronizadas en segundo plano. Así mismo, se dice que a un futuro estos servicios podrán soportar las sincronizaciones periódicas o conocidas como perimetraje [23].

A continuación, se presenta el ciclo de vida de un servicio worker [27]:

- ❖ **Instalación para el sitio,** una vez que ya sea registrado, se lo hace mediante el lenguaje de programación JavaScript, este paso lo hará en segundo plano.
- ❖ **Activación,** en esta etapa se administra los cachés.
- ❖ **Control,** el service worker podrá controlar las páginas que tenga a su alcance, pese a que, su primera página registrada no será controlada sin antes cargarla.
- ❖ **Rescindir,** el service se rescindiré con el fin de ahorrar memoria, así mismo mantendrá el control de los eventos de mensaje.

CAPÍTULO 3. FASE DE EXPLORACIÓN

En este capítulo se detallará la planificación, acción e investigación junto con la organización del proyecto en flujos de trabajo para elaborar los requerimientos y roles para la aplicación.

3.1 Planificación con los stakeholders

En el plan de tesis se definió que el proyecto tendría tres reuniones mediante la plataforma ZOOM con el gerente de la empresa MANTICORE LABS, en las mismas que se logró obtener los requerimientos necesarios para la elaboración de la aplicación móvil mediante los problemas que se presentaban en dicha empresa, esta encuesta se encuentra en el apartado de Anexos. Asimismo, se diseñó los diagramas de procesos y prototipos de la aplicación móvil la mismas que confirmara y sustentara los requerimientos que se recolectaron durante las encuestas.

3.2 Recolección de requerimientos

Esta sección detalla las encuestas realizadas a la empresa para el compendio de requerimientos.

3.2.1 Encuesta dirigida a los trabajadores de la empresa

1. ¿La creación de una aplicación proporcionará una mejor ayuda para almacenar los servicios que ofrecen los integrantes de Manticore Labs?

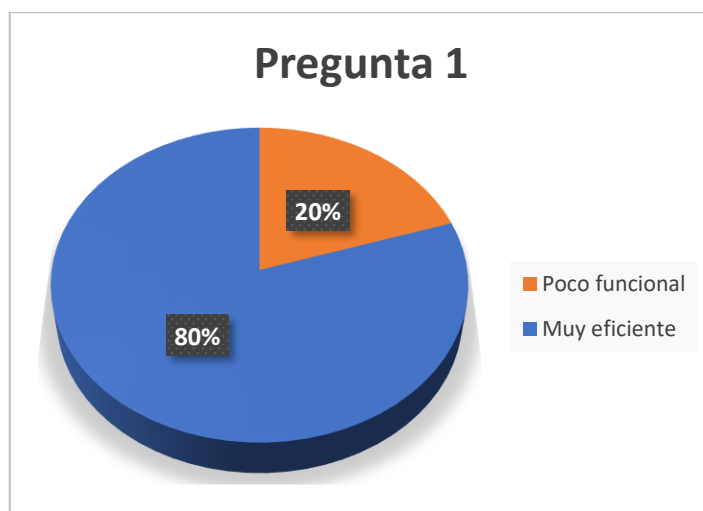


Figura 4: Pregunta 1
Fuente: La Encuesta

Análisis e interpretación

El 80% de los encuestados respondieron que la creación de la aplicación ayudara en almacenar los servicios que ofrece la empresa MANTICORE LABS, mientras que el 20% respondió que no sería funcional, debido a que están acostumbrados a llevar la información en una hoja de Excel y la aplicación móvil les causaría doble trabajo.

2. ¿La segmentación de los roles en el aplicativo proporcionará una mejor distribución de los usuarios que usarán la aplicación?

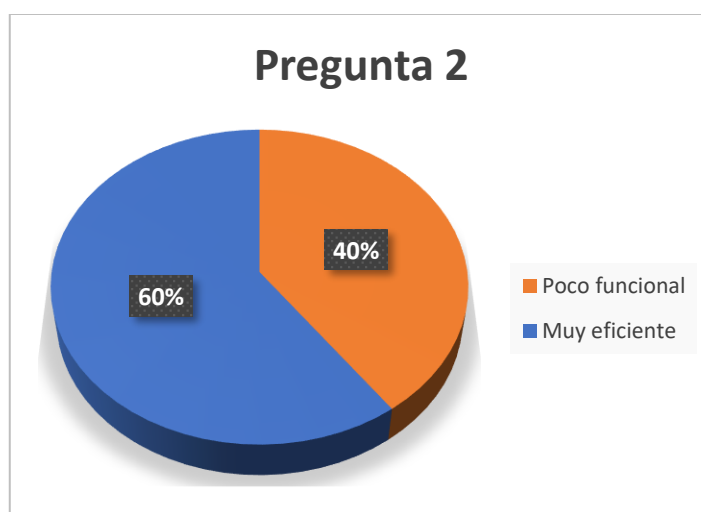


Figura 5: Pregunta 2
Fuente: La Encuesta

Análisis e interpretación

El 60% de los encuestados respondieron la aplicación roles en la aplicación proporcionará una mejor distribución de los usuarios en la empresa MANTICORE LABS, mientras que el 40% respondió que no sería funcional.

3. ¿Los empleados de Manticore Labs considerarán que la elaboración del aplicativo permitirá obtener un número importante de usuarios?

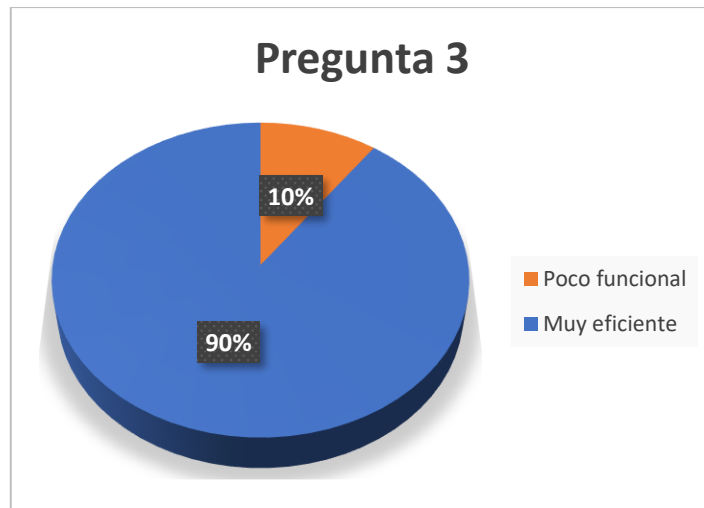


Figura 6: Pregunta 3
Fuente: La Encuesta

Análisis e interpretación

El 90% de los encuestados respondieron que con la elaboración de la aplicación ayudara a obtener mayor cantidad de usuarios para la empresa MANTICORE LABS, mientras que el 10% respondió que no sería funcional.

4. ¿Consideran que la negociación de los precios ante los servicios que se ofrecerán en el aplicativo es un plus ante las aplicaciones que existen en el mercado?

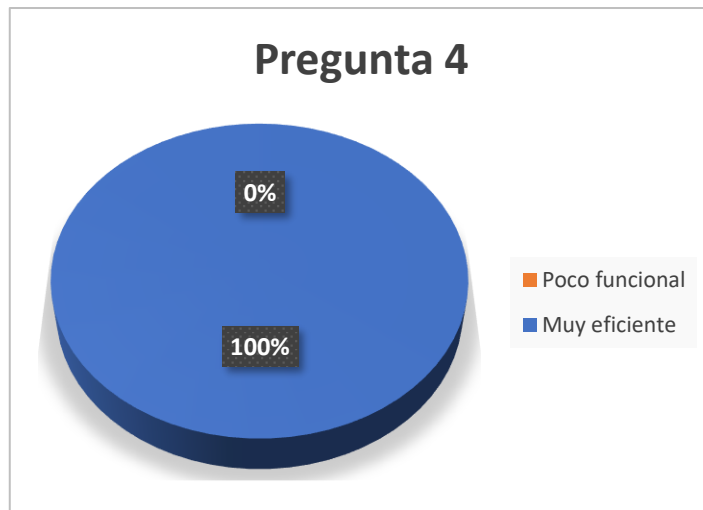


Figura 7: Pregunta 4
Fuente: La Encuesta

Análisis e interpretación

El 100% consideran que la negociación de los precios ante los servicios que se ofrecerán en el aplicativo es un plus ante las aplicaciones que existen en el mercado y que es muy eficiente la creación de la aplicación.

- ¿La implementación de la funcionalidad de geolocalización de los usuarios es de utilidad para identificar por sectores a los usuarios del aplicativo de Manticore Labs?

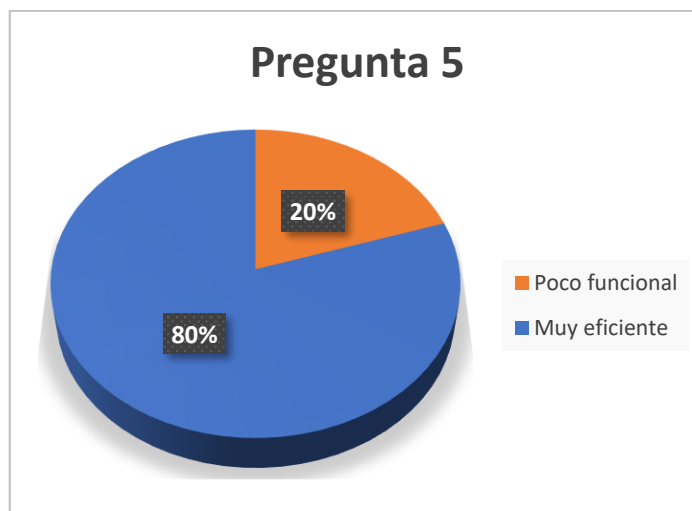


Figura 8: Pregunta 5
Fuente: La Encuesta

Análisis e interpretación

El 80% de los encuestados manifestó que la implementación de la funcionalidad de geolocalización de los usuarios es de utilidad para identificar por sectores a los usuarios que utilizan la aplicación de la empresa MANTICORE LABS y así poder realizar un estudio de mercado para las estrategias, mientras que el 20% respondió que no sería funcional.

6. ¿Consideran que la aplicación alcanza los estándares que esperan los inversionistas de la empresa Manticore Labs?

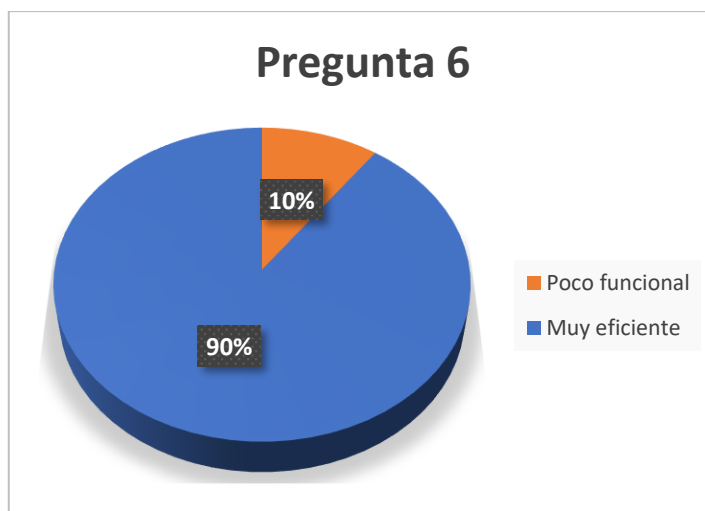


Figura 9: Pregunta 6
Fuente: La Encuesta

Análisis e interpretación

El 90% de los encuestados manifestaron que la aplicación alcanzara los estándares que esperan los inversionistas de la empresa MANTICORE LABS, mientras que el 10% respondió que no sería funcional.

7. ¿Consideran que la aplicación pueda crecer, así como crezca la demanda de sus usuarios?

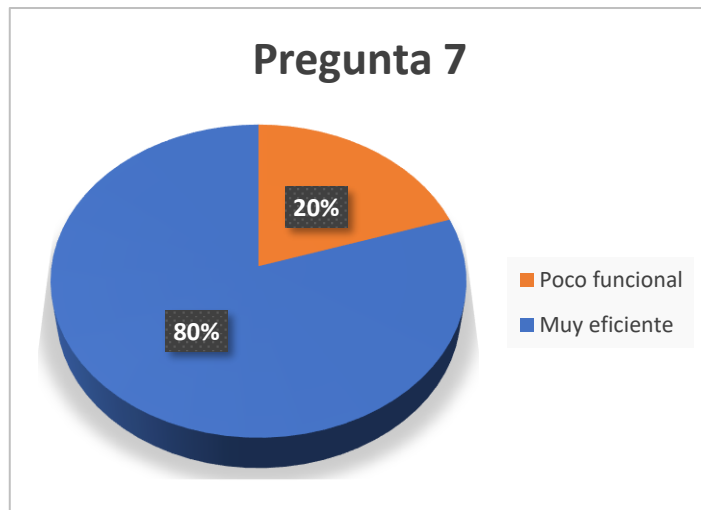


Figura 10: Pregunta 7
Fuente: La Encuesta

Análisis e interpretación

El 80% de los encuestados manifestaron que la aplicación ayudara que crezca la demanda de sus usuarios en la empresa MANTICORE LABS, mientras que el 20% respondió que no sería funcional.

Según la información recolectada por medio de la encuesta aplicada a los trabajadores de la empresa MANTICORE LABS, se obtuvo los siguientes requerimientos funcionales y no funcionales los mismos que son descritos a continuación:

3.3 Roles de SCRUM

La Tabla 1 describe las funcionalidades y roles de cada individuo que participaron en el proyecto.

Tabla 1: Roles de SCRUM

| Partes Interesadas (SCRUM) | | | |
|----------------------------|---------------------------------------|----------------|--|
| ID | Roles | Representantes | Descripción |
| S1 | Scrum Master | Cristian Lara | Profesional responsable de que todos eventos SCRUM se cumplan. |
| S2 | Product Owner (Dueño del Producto) | Adrián Egüés | Profesional que representa al stakeholders o parte interesada |

| | | | |
|-----------|--------------------------------------|-------------|---|
| S3 | Development Team (Equipo Desarrollo) | Mario Giler | Profesional responsable de desarrollar el producto. |
|-----------|--------------------------------------|-------------|---|

CAPÍTULO 4. FASE DE INICIALIZACIÓN

Dentro del contenido del presente capítulo se especificó la configuración del entorno de desarrollo, las herramientas escogidas y la arquitectura implementada para la elaboración de la aplicación.

4.1 Lenguajes de programación

La Tabla 2 lista los lenguajes de programación y marcas utilizados en el proyecto.

Tabla 2: Lenguajes

| Nombre | Descripción | Aplicación en el proyecto |
|-------------------|---|---|
| JavaScript | Lenguaje de programación que permite interactividad dinámica. | Utilizado para desplegar el compilado del aplicativo PWA. |
| CSS | Lenguaje que permite personalizar el estilo y fuentes de los documentos HTML. | Utilizado en el diseño de la PWA. |
| HTML | Lenguaje para estructurar y elaborar una pagina web o PWA. | Utilizado en la estructura de la PWA. |
| NoSQL | Lenguaje de programación para base datos de no relacional | Utilizado para la base de datos del proyecto. |

4.2 Herramientas de desarrollo

La Tabla 3 lista las herramientas utilizadas en el proyecto.

Tabla 3: Herramientas

| Nombre | Descripción | Aplicación en el proyecto |
|--------|-------------|---------------------------|
|--------|-------------|---------------------------|

| | | |
|--------------------------------|---|---|
| WebStorm | Editor de texto creado por JetBrains. | Utilizado para escribir el código del proyecto. |
| Git | Sistema de versionamiento de código. | Utilizado para el control de versiones del proyecto. |
| GitHub | Repositorio de códigos de proyecto de programación y control de versiones. | Utilizado para el versionamiento de código del proyecto. |
| Cloud Firestore | Base de datos NoSQL que permite almacenar e indexarlas fácilmente. | Se utilizó para almacenar las sesiones de los usuarios. |
| Firebase Authentication | Servicio de autenticación de Firebase. | Se utilizó para la autenticación de usuarios. |
| MongoDB | Base de datos no relacional de Apache. | Se utilizó para almacenar la información de las citas y reuniones de la aplicación. |
| Heroku | Servicio que proporciona una plataforma para alojar diferentes aplicativos. | Se utilizó para alojar la base de datos junto con el frontend y backend del aplicativo. |

4.3 Frameworks y Librerías

La Tabla 4 lista todos los frameworks y librerías que se ha utilizado en el proyecto.

Tabla 4: Frameworks y Librerías

| Nombre | Descripción | Aplicación en el proyecto |
|----------------|---|--|
| Node.js | Entorno de ejecución basado en el Lenguaje JavaScript y el motor V8 de Google. | Utilizado para la ejecución del sistema web. |
| NPM | Gestor de paquetes que permite obtener librerías, agregar dependencias y administrar módulos. | Utilizado para instalar los diferentes paquetes usados en el proyecto. |

| | | |
|--------------------|---|---|
| Angular | Framework que permite diseñar aplicaciones y plataformas de manera eficiente escalable y sofisticada. | Utilizado para el front-end del sistema web. |
| Material-UI | Librería que permite crear una interfaz de usuario, con los principios de diseño Google. | Utilizado para la interfaz del sistema web. |
| Ionic | Framework que permite elaborar aplicaciones híbrididad basadas en tecnologías web. | Utilizado para compilar y desplegar el aplicativo en plataformas móviles. |

4.4 Configuración del proyecto

4.4.1 Formato épicas de usuario

En la elaboración del presente proyecto se definió 3 Sprints los mismos que estarían divididos en los roles de administrador, prestador de servicios y cliente.

A continuación, se detalla las especificaciones que contendrá el modelo de las historias de usuario:

- ✓ Número: identificador para la historia de usuario.
- ✓ Usuario: responsable de describir la historia de usuario.
- ✓ Nombre: título de la historia de usuario acorde a su descripción.
- ✓ Prioridad en negocio: necesidad de negocio: alta, media o baja.
- ✓ Riesgo en desarrollo: impacto a desarrollar en caso de que se presenten fallos: alto, medio o bajo.
- ✓ Puntos estimados: tiempo para finalizar la historia de usuario.
- ✓ Iteración asignada: representa al número que se designa la historia de usuario.
- ✓ Programador responsable: quien realiza la historia de usuario.
- ✓ Descripción: se detalla el ¿qué hacer? de la historia de usuario.
- ✓ Observación: especifica los procesos respecto a la historia de usuario.

De esta manera, se presenta el modelo que se usará para las historias de usuario:

Tabla 5: Modelo para historias de usuario

| Historia de usuario | |
|--------------------------|------------------------------|
| Número: | Usuario: |
| Nombre historia: | |
| Prioridad en Negocio: | Riesgo en desarrollo: |
| Puntos estimados: | Iteración asignada: |
| Programador responsable: | |
| Descripción: | |
| Observación: | |

Así mismo, en la Figura 11 se presenta el diseño de las historias en el software Trello.

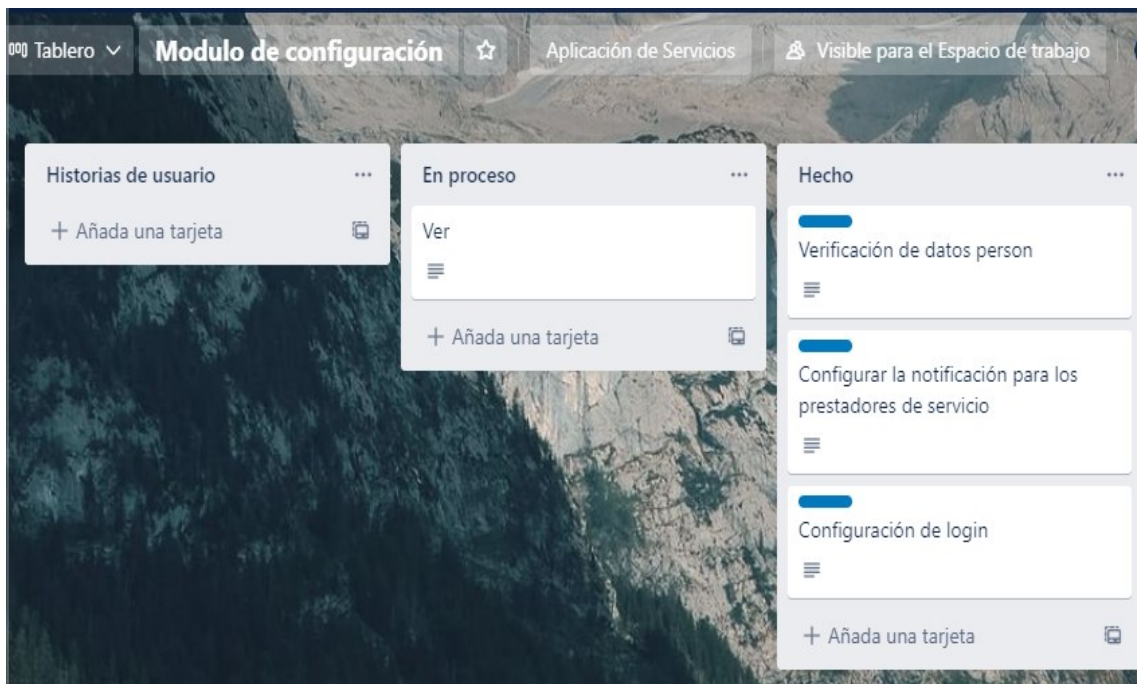


Figura 11: Historias de usuario desde Trello

4.4.2 Historias épicas de usuario

Para la elaboración de las épicas se mantuvo varias reuniones con el Gerente de la empresa, en las cuales supo manifestar la necesidad que tenían en elaborar una aplicación móvil que permita agendar citas y reuniones. En el diálogo de cada reunión se identificaron varios flujos de trabajo y se pudieron segmentar para elaborar un conjunto de épicas, las mismas que se detallan en la Tabla 7.

Tabla 6: Épicas de la aplicación

| Código | Nombre | Descripción |
|--------|--------------------------|--|
| TI01 | Administrador | Módulo de Administrador del Sistema para agendar citas y reuniones |
| TI02 | Gestión de profesionales | Módulo de Gestión de profesionales |
| TI03 | Gestión de cliente | Módulo de Gestión de clientes |

Una vez definidas las épicas, se definió y se obtuvo la lista de los requerimientos funcionales y no funcionales que tendrá la aplicación móvil mediante las reuniones que se mantuvo con el gerente de la empresa.

Tabla 7: Requerimientos funcionales

| Código | Descripción |
|--------|--|
| RF01 | El sistema validará los datos ingresados en la creación de los usuarios |
| RF02 | El sistema validará que se suban imágenes o fotografías en el registro de las citas |
| RF03 | El sistema validará que el formato del correo sea el correcto |
| RF04 | El sistema permitirá crear una cuenta para agendar una cita |
| RF05 | El sistema enviará un correo de verificación para el registro de las citas |
| RF06 | El sistema permitirá iniciar sesión con la cuenta creada en el aplicativo |
| RF07 | El sistema consultará datos y reseñas de las citas agendadas |
| RF08 | El sistema permitirá ingresar nuevos servicios y citas |
| RF09 | El sistema notificará las citas y servicios que hayan sido agendados |
| RF10 | El sistema permitirá comunicarse internamente en el aplicativo |
| RF11 | El sistema permitirá calificar la cita agendada |
| RF12 | El sistema permitirá confirmar el horario establecida para la cita |
| RF13 | El sistema permitirá validar que los campos obligatorios en el registro de citas se ingresen correctamente |
| RF14 | El sistema permitirá evaluar la cita y el servicio agendado |

| | |
|-------------|---|
| RF15 | El sistema permitirá realizar filtros de búsqueda por medio de fechas de agendamiento |
| RF16 | El sistema permitirá solicitar agendar y reservar una cita |
| RF17 | El sistema permitirá guardar la ubicación del cliente |
| RF18 | El sistema permitirá comentar acerca de una cita agendada |

Tabla 8: Requerimientos no funcionales

| Código | Descripción |
|--------------|---|
| RNF01 | El sistema debe ser competente para permitir 1000 citas por segundo |
| RNF02 | La información almacenada en la base de datos debe ser respaldada cada día en horario nocturno |
| RNF03 | Las funcionalidades de espera en el sistema tienen que tener un tiempo de respuesta máximo de 30 segundos |
| RNF04 | Las comunicaciones entre el servidor y el aplicativo no pueden llegar en plano sino encriptadas |
| RNF05 | El sistema debe tener un manual de usuario estructurado correctamente |
| RNF06 | En caso de que exista un error y se reinicie la aplicación debe tener un tiempo de inicialización de máximo 5 minutos |
| RNF07 | Las credenciales de acceso al sistema podrán ser cambiadas solamente por el administrador de base de datos |
| RNF08 | El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final. |
| RNF09 | La probabilidad de falla del Sistema no podrá ser mayor a 0,5%. |
| RNF10 | La aplicación no debe pesar más de 30 MB |

Para definir la complejidad de cada historia de usuario se utilizó “Planning Poker”, también conocido como “Scrum Poker”, es una técnica utilizada para calcular el esfuerzo de las tareas de proyectos de desarrollo de software. Esta herramienta nos permite medir la cantidad de puntos de la historia para las tareas relevantes del equipo de desarrollo, se necesita calcular el esfuerzo que tomaría completar cada tarea en una escala exponencial. De modo que, brinda si una tarea es el doble de compleja y tomaría el doble del tiempo en ser completada. Primeramente, se escogió la tarea que tomaría el menor esfuerzo para ser usada como guía base, en este caso HI08 (ver Tabla 11).

La dificultad asignada para las demás historias es una representación del tamaño de cada historia de usuario en comparación con la historia base.

De la misma forma por medio de los requerimientos funcionales se obtuvo las historias de usuario presentes en la aplicación móvil, como también se definió una tabla donde se sacó la complejidad y la prioridad de cada una de las historias. Para la prioridad se usaron los valores mostrados en la siguiente TABLA 6.

Tabla 9: Tabla de prioridades

| Prioridad | Valoración |
|--------------|------------|
| Baja | 1 |
| Media | 2 |
| Alta | 3 |

A continuación, se muestra las historias de usuario agrupadas por épicas

Tabla 10: Historias de usuario ordenado por épicas

| Épica | Código | Nombre | Prioridad |
|-------|--------|--|-----------|
| TI01 | HI01 | Como usuario administrador quiero recibir toda la información de los usuarios que están creando cuentas para verificar que sean usuarios legítimos y no robots. | 2 |
| | HI02 | Como usuario administrador quiero recibir las fotografías ingresadas en la creación de cuentas para añadir la fotografía al portafolio interno de los usuarios registrados en la aplicación. | 2 |
| | HI03 | Como usuario administrador quiero que se valide el ingreso del correo para que una vez sea ingresado con éxito pueda enviarse un correo de activación de la cuenta. | 2 |
| TI02 | HI04 | Como usuario prestador de servicios quiero crear una cuenta para registrar los oficios que puedo ofertar. | 1 |
| | HI05 | Como usuario prestador de servicios quiero que al registrarme en el aplicativo se validen la información ingresada, en particular el correo para que una vez se envíe la información me pueda llegar correctamente las credenciales al correo ingresado en el sistema. | 2 |
| | HI06 | Como usuario prestador de servicios quiero loguearme en la aplicación mediante mi correo para | 1 |

| | | | |
|------|------|---|---|
| | | administrar la información de las diversas asistencias registradas. | |
| | HI07 | Como usuario prestador de servicios quiero consultar el feedback de mis servicios que han sido evaluados por el cliente para tener retroalimentación del trabajo proporcionado. | 3 |
| | HI08 | Como usuario prestador de servicios quiero modificar descripciones o archivos de fotos de la información creada anteriormente para corregir errores de ortografía o mejorar las fotos para que se visualice mejor el servicio brindado. | 3 |
| | HI09 | Como usuario prestador de servicios quiero requiere recibir notificaciones de los servicios que son solicitados para que pueda estar pendiente de los nuevos pedidos de los servicios solicitados. | 3 |
| | HI10 | Como usuario prestador de servicios quiero tener una comunicación con el cliente a través de un servicio de chat para intercambiar mensajes de acuerdos de negociación de la cita. | 3 |
| TI03 | HI11 | Como usuario cliente quiero crear una cuenta para poder adquirir una cita de un servicio que necesito. | 1 |
| | HI12 | Como usuario cliente quiero que al instante de registrarme en el aplicativo se validen la información ingresada para que no pueda ingresar correos o información distinta a lo que se piden en los datos de ingreso. | 2 |
| | HI13 | Como usuario cliente quiero iniciar sesión en la aplicación mediante mi correo para solicitar y agendar una cita de los servicios que requiero. | 1 |
| | HI14 | Como usuario cliente quiero calificar el servicio prestado por parte del profesional para que otros usuarios puedan tener una referencia del tipo de servicio que ofrece ya sea bueno o malo. | 2 |
| | HI15 | Como usuario cliente quiero consultar los datos del usuario que ofrece los servicios profesionales para verificar la identidad del profesional. | 2 |

| | | |
|------|---|---|
| HI16 | Como usuario cliente quiero solicitar un servicio profesional que requiero para solventar una incidencia que poseo como cliente. | 3 |
| HI17 | Como usuario cliente quiero consultar servicios que existen en la aplicación para solicitar alguno que cumpla con mis necesidades de cliente. | 3 |
| HI18 | Como usuario cliente quiero registrar mi posición actual del lugar de residencia para que el usuario prestador de servicios pueda observar la ubicación en el mapa. | 3 |

4.5 Sprint 0

- **Planificación**

Dentro de este sprint se planteó la arquitectura del aplicativo junto con la creación de un pipeline para poder ejecutar parcialmente un ciclo de devops pero la fase principal es la integración y entrega continua. A la vez, se creó un repositorio junto con las ramas de desarrollo y master para diferenciar los ambientes.

- **Ejecución**

- Arquitectura del proyecto
- Pipeline

- **Arquitectura del proyecto**

La arquitectura a bajo nivel utilizada para este aplicativo se representa en la Figura 12. El equipo Scrum instanció una arquitectura MEAN, porque contiene las siguientes herramientas, las cuales son: MongoDB, Express.js, Angular.js y Node.js.

La Figura 12 muestra la arquitectura Cliente - Servidor, donde se la comunicación que existe entre componentes son métodos de petición HTTP junto con proxys para la conexión ante servicios externos, en este caso MongoDB.

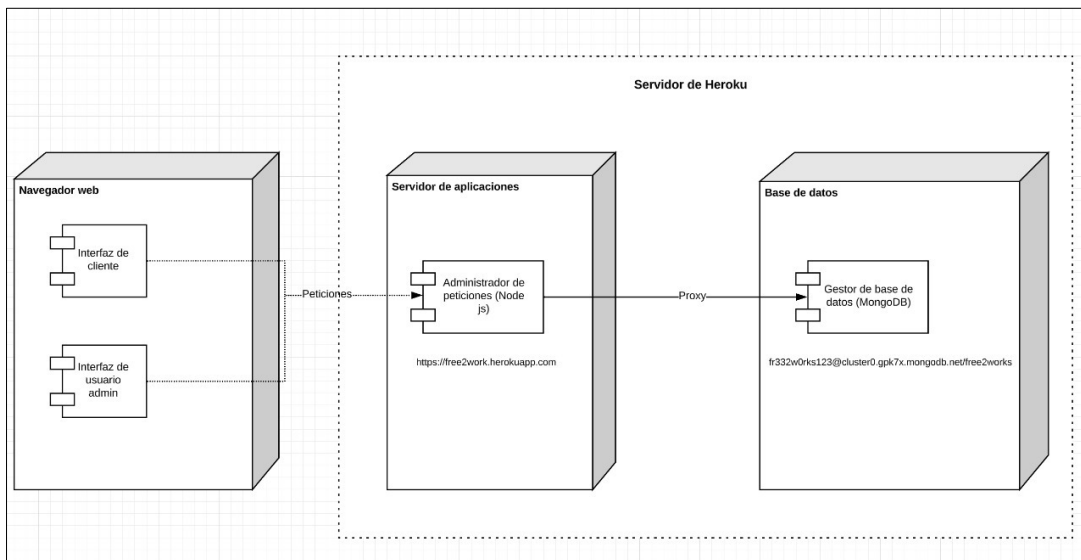


Figura 12: Diagrama de paquete UML

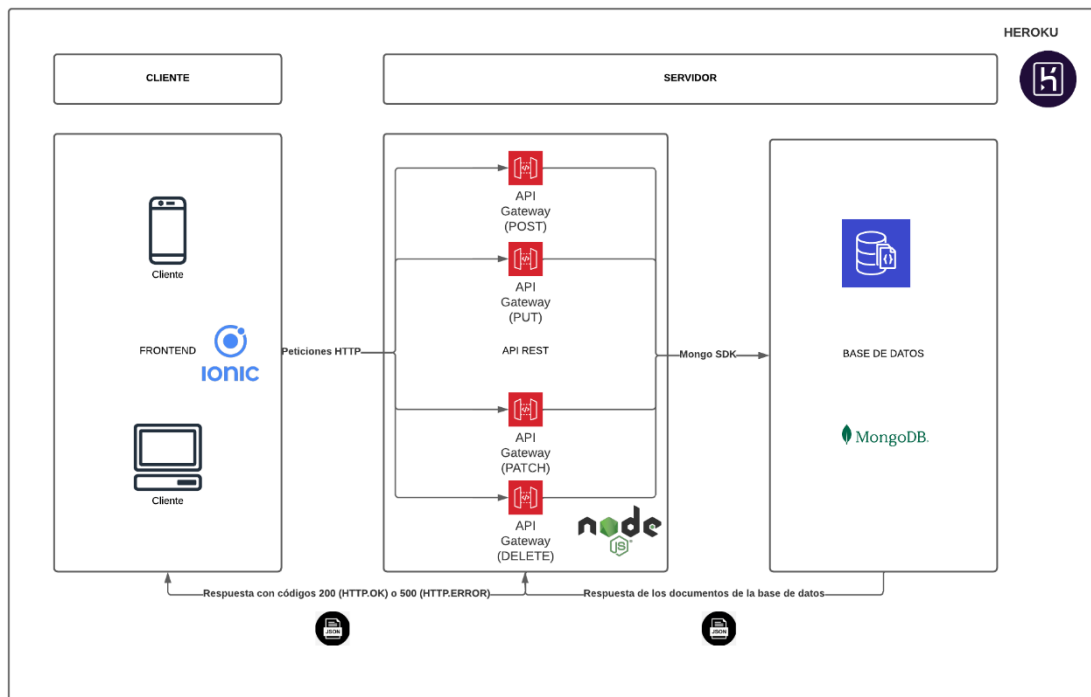


Figura 13: Arquitectura del proyecto

El servicio de base de datos es MongoDB donde se almacenará los datos de los usuarios y la información que vayan ingresando al aplicativo. Además, MongoDB posee un servicio con interfaz gráfica de CLI (Command Line Interface), en la que se puede realizar búsquedas de los documentos almacenados.

El servicio web hace referencia a Node.js, donde nos permite nos permite alojar nuestro servicio de la aplicación para las peticiones HTTP (Backend) , ya que es un entorno de tiempo de ejecución de Javascript.

Finalmente, cliente se refiere a la capa funcional en donde el usuario navega directamente con el aplicativo para realizar un flujo funcional de las funcionalidades que existen.

A nivel de organizacional del proyecto se usó el patrón de arquitectura de software Entidad Componente Sistema (ECS) Figura 13.

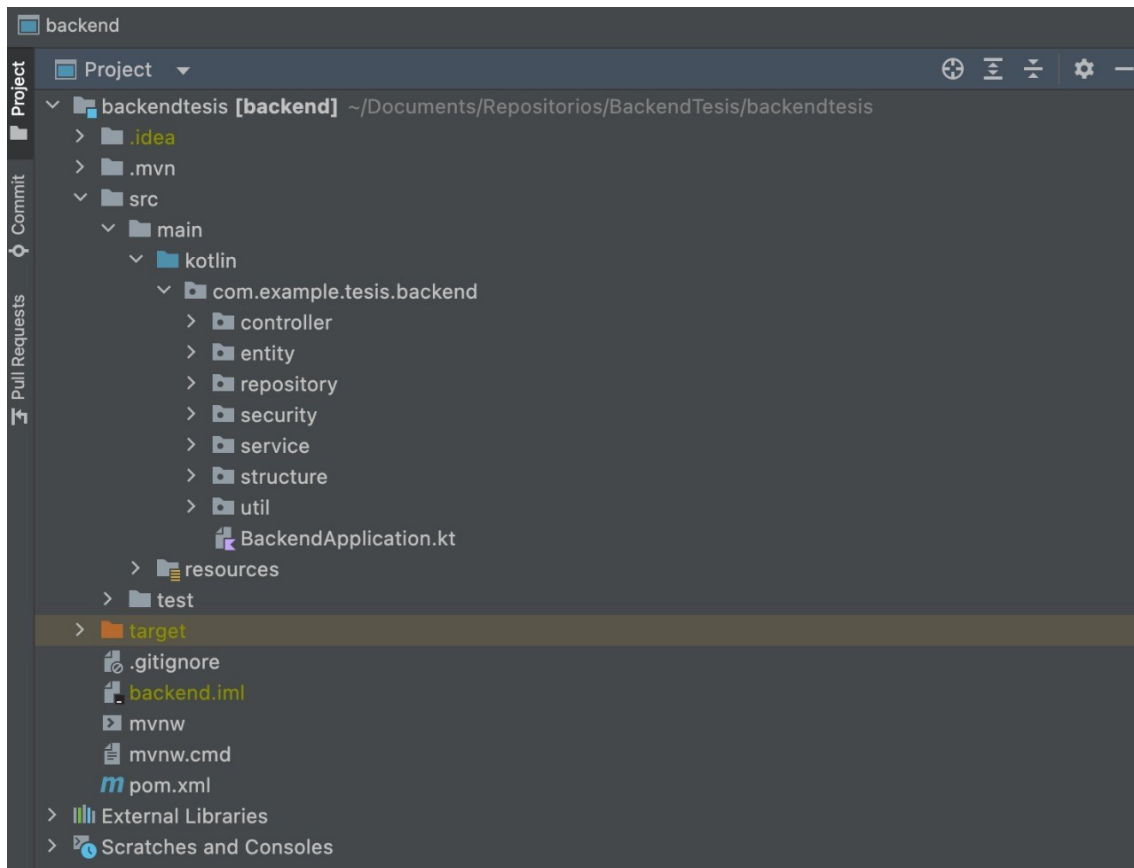


Figura 14: Entidad Componente Sistema

- **Pipeline**

El equipo de desarrollo ha implementado un pipeline para realizar integración y despliegue continuo. De modo que, se elaboró un pipeline en Amazon Web Service, el cual permite desplegar en los ambientes productivos y de desarrollo cada cambio que se realice a la rama. Esta funcionalidad nos proporciona evitar lanzar comandos locales para conectarnos al servidor y depositar el compilado.

El pipeline contiene diferentes fases al momento de realizar el CI/CD, a continuación, se detallan estas fases:

- Fuente. - Es aquel que se conecta con el repositorio en este caso github y toma las fuentes para poder alojarlos en la máquina de aws. En la Figura 14 se muestra la fase Fuente y en la Figura 15 se muestra la conexión hacia el repositorio de la aplicación.

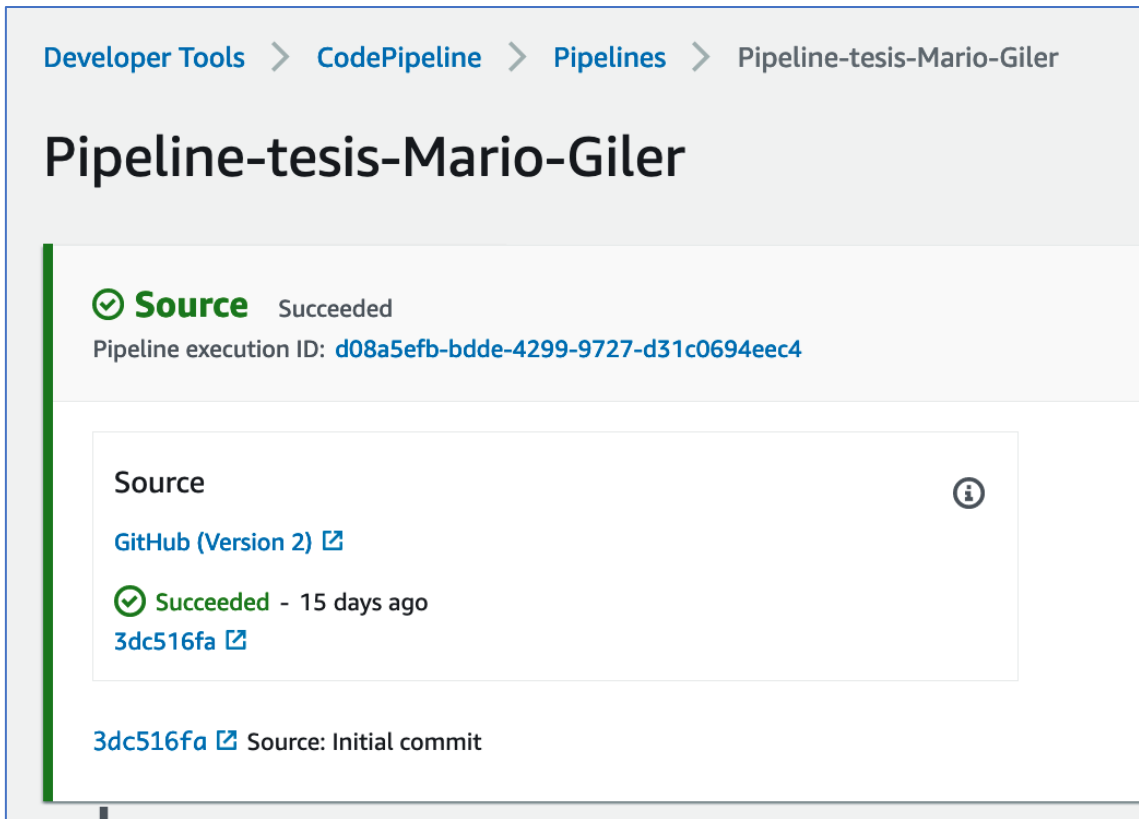


Figura 15: Fase de fuentes.

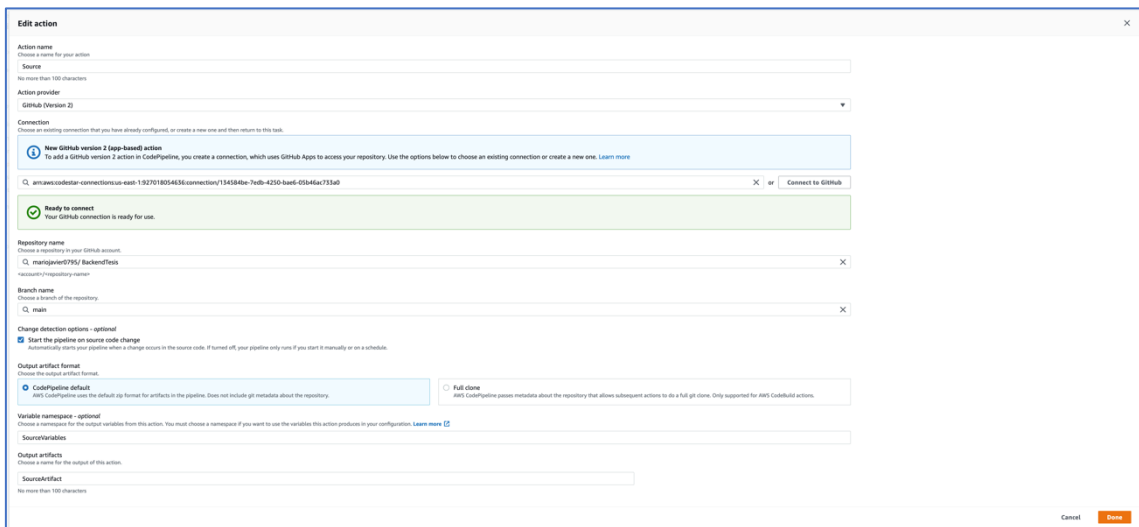


Figura 16: Configuración de conexión GitHub.

- **Actualizar Commit ID.** - Permite renovar cada commit que se realiza a la rama para no chocar con los diferentes commits que pueden existir en el repositorio. En la Figura 16 se detalla la fase.

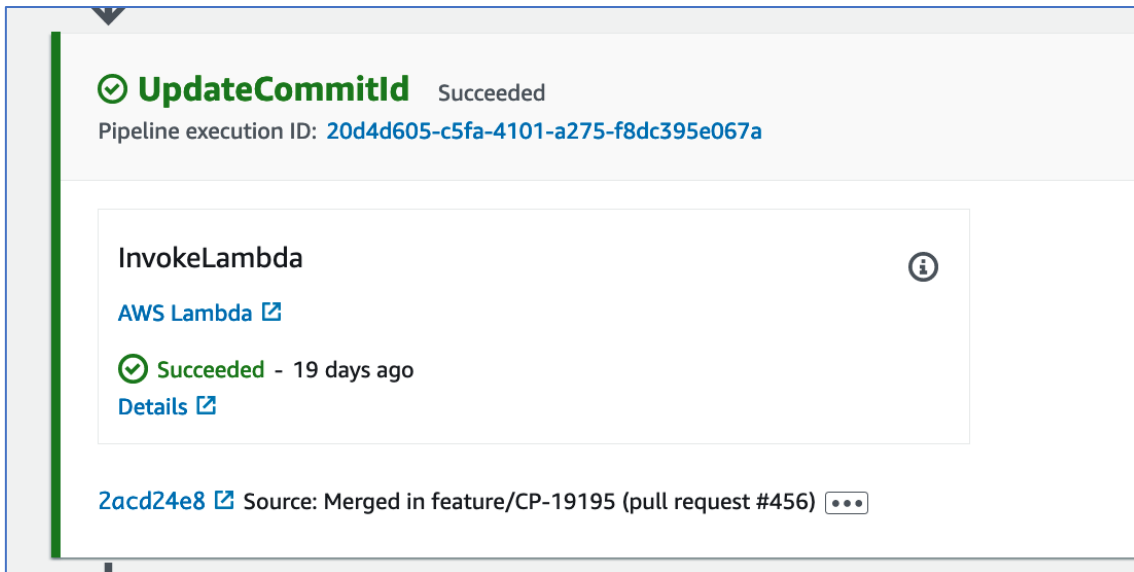


Figura 17: Fase de actualización del commit.

- Aprobación Manual. - Esta fase es solo para poder autorizar que el commit que está siendo ingresado continúe en los ciclos de deploy o se interrumpa. En la [Figura 17](#) se puede observar la aprobación.

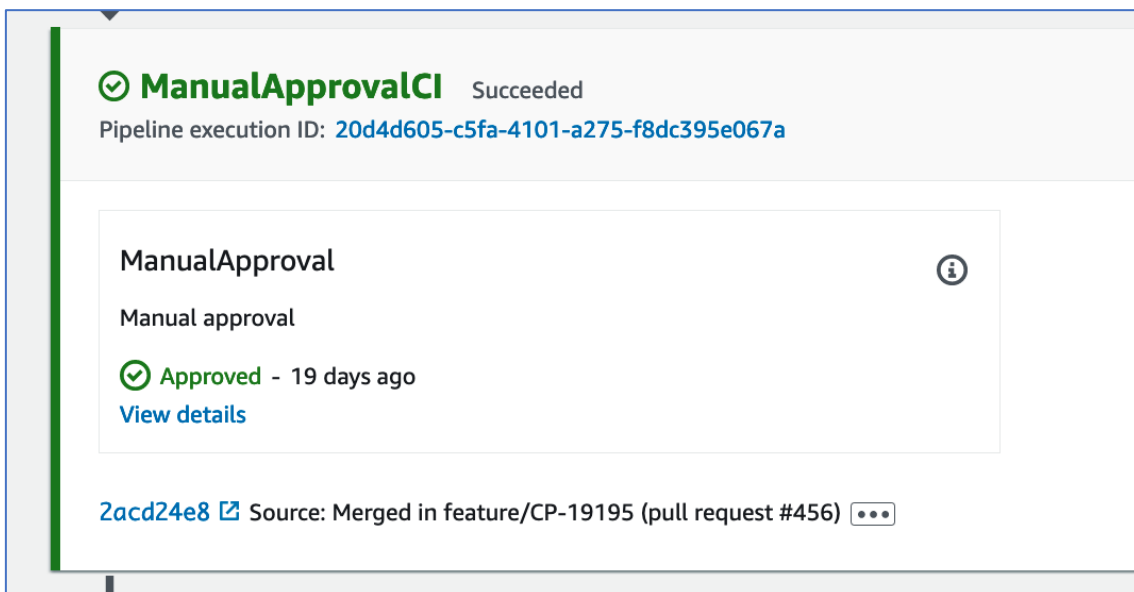


Figura 18: Fase de aprobación del commit

- Construcción y Pruebas. - Esta fase es la más importante porque contiene los comandos que se ejecutarán en un archivo denominado buildspec y permitirá correr las pruebas unitarias de la aplicación. Se mencionó que los comandos se agregan manualmente junto con credenciales de npm para que puedan ser leídas y ejecuten las acciones de construcción como se detalla en la Figura 18. En la Figura 19 se expone las dos fases de construcción y de prueba.

```

Buildspec

1 version: 0.2
2
3 env:
4   variables:
5     USRV_STAGE: "UNSET"
6     USRV_COMMIT: "UNSET"
7     USRV_BRANCH: "UNSET"
8     SLS_DEBUG: "*"
9   parameter-store:
10    NPM_TOKEN: /global/NPM_TOKEN
11  phases:
12
13    install:
14      runtime-versions:
15        nodejs: 12
16      commands:
17        - echo Installing pre-requirements
18    pre_build:
19      commands:
20        - echo Install... @.@
21        - touch .npmrc
22        - printf "\n//registry.npmjs.org/:_authToken=\"${NPM_TOKEN}\"" >> .npmrc
23        - npm ci
24        # - zip -r node_modules.zip node_modules/. node_modules
25        # - ls node_modules
26        # - npm list
27        - npm run tsc:interface
28        # - npm i -g serverless
29    build:
30      commands:
31        - echo the artifact will be ready soon
32        # - export NODE_OPTIONS=--max_old_space_size=6144 #6144
33        # - serverless package --verbose --package my-artifacts
34        - node --max-old-space-size=11264 /usr/local/bin/npx serverless package --verbose --package my-artifacts
35        # - serverless package --verbose --package my-artifacts
36        - ls
37        - ls ../..
38    post_build:
39      commands:
40        - echo Build completed on `date`
41  artifacts:
42    files:
43      - ./my-artifacts/*
44      - ./serverless.yml
45      - ./package.json
46      - ./package-lock.json
47      - ./npmrc
48      #- ./node_modules/**/**
49      #- ./node_modules.zip
50      - ./RetentionOption.js
51      - ./ServerlessScripts.js
52

```

Figura 19: Archivo de construcción de comandos para la fase del build.

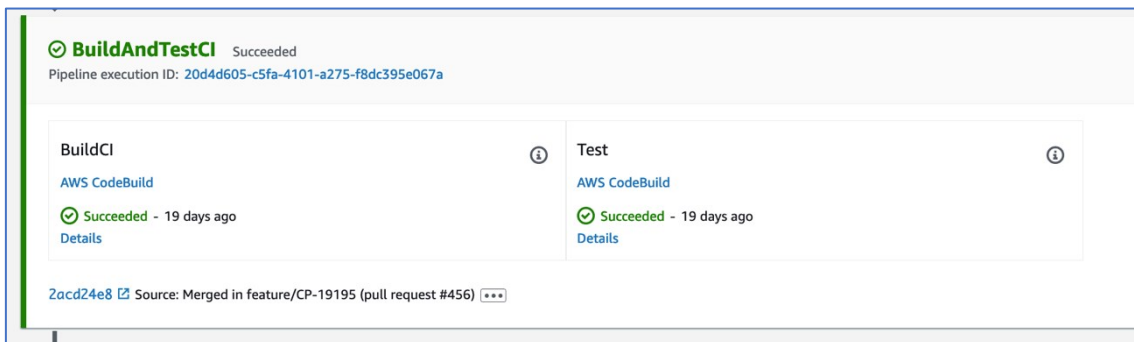


Figura 20: Fase de construcción y de prueba.

- **Despliegue.** - Esta fase es la encargada de hacer el despliegue en el respectivo servidor. Así mismo, como con la fase de construcción contiene un archivo buildspec para conectarse al servidor y realizar la entrega del compilado del

proyecto como se lo realizó en la Figura 20. Para la Figura 21 expone la fase de despliegue.

```
Buildspec
1 version: 0.2
2
3 env:
4 variables:
5   ARTIFACT: s3://artifact-build-cotopaxi/usrv_back_build
6   SLS_DEBUG: "*"
7 parameter-store:
8   NPM_TOKEN: /global/NPM_TOKEN
9 phases:
10 install:
11   runtime-versions:
12     nodejs: 16
13   commands:
14     - echo Installing pre-requirements...
15     #- npm install -g npm ts-node tslint typescript serverless@1.26.1
16
17 pre_build:
18   commands:
19     - echo Download $ARTIFACT ... @.@
20     - ls
21     - aws --debug s3 cp $ARTIFACT ./artifact.zip
22     - if [ -z $REGION ]; then REGION=us-east-1; fi
23 build:
24   commands:
25     - ls
26     - echo the artifact will be ready soon
27     - unzip -o artifact.zip
28     #- unzip node_modules.zip
29     #- touch .npmrc
30     #- printf "\n//registry.npmjs.org/:_authToken=\"${NPM_TOKEN}\"" >> .npmrc
31     - npm ci
32
33     #- npm uninstall serverless
34     #- npm i serverless --force
35
36     - npx serverless deploy --force --package my-artifacts --region $REGION
37     #- npm run deploy
38     - ls
39 post_build:
40   commands:
41     - npx serverless prune -n 2 --stage $USRV_STAGE --region $REGION
42     # - echo Optimizing Lambda
43     # - echo $file > test.txt
44     # - sed -i "s/$USRV_NAME/$USRV_NAME/g" ./test.txt
45     # - sed -i "s/$USRV_ENV/$USRV_STAGE/g" ./test.txt
46     # - cat test.txt
47     # - aws lambda invoke --function-name usrv-lambda-analytics-dev-analyzeLambdaConfig --payload file://test.txt response.json
48     # - cat response.json
49     - echo Build completed on `date` |
```

Figura 21: Archivo de despliegue de comandos para la fase del build.

The screenshot shows a successful pipeline execution in AWS CodeBuild. At the top, a green checkmark icon is followed by the text "DeployCI Succeeded". Below this, the "Pipeline execution ID" is displayed as "20d4d605-c5fa-4101-a275-f8dc395e067a". A central card displays "DeployCI" with an information icon, "AWS CodeBuild", and "Succeeded - 19 days ago" with a "Details" link. At the bottom, the source is identified as "2acd24e8" with a link icon, and the commit message is "Source: Merged in feature/CP-19195 (pull request #456)" with a three-dot menu icon.

Figura 22: Fase de despliegue.

Para finalizar, en la Figura 22 se visualiza todo el pipeline conectado de manera secuencial para un ciclo devops en la aplicación.

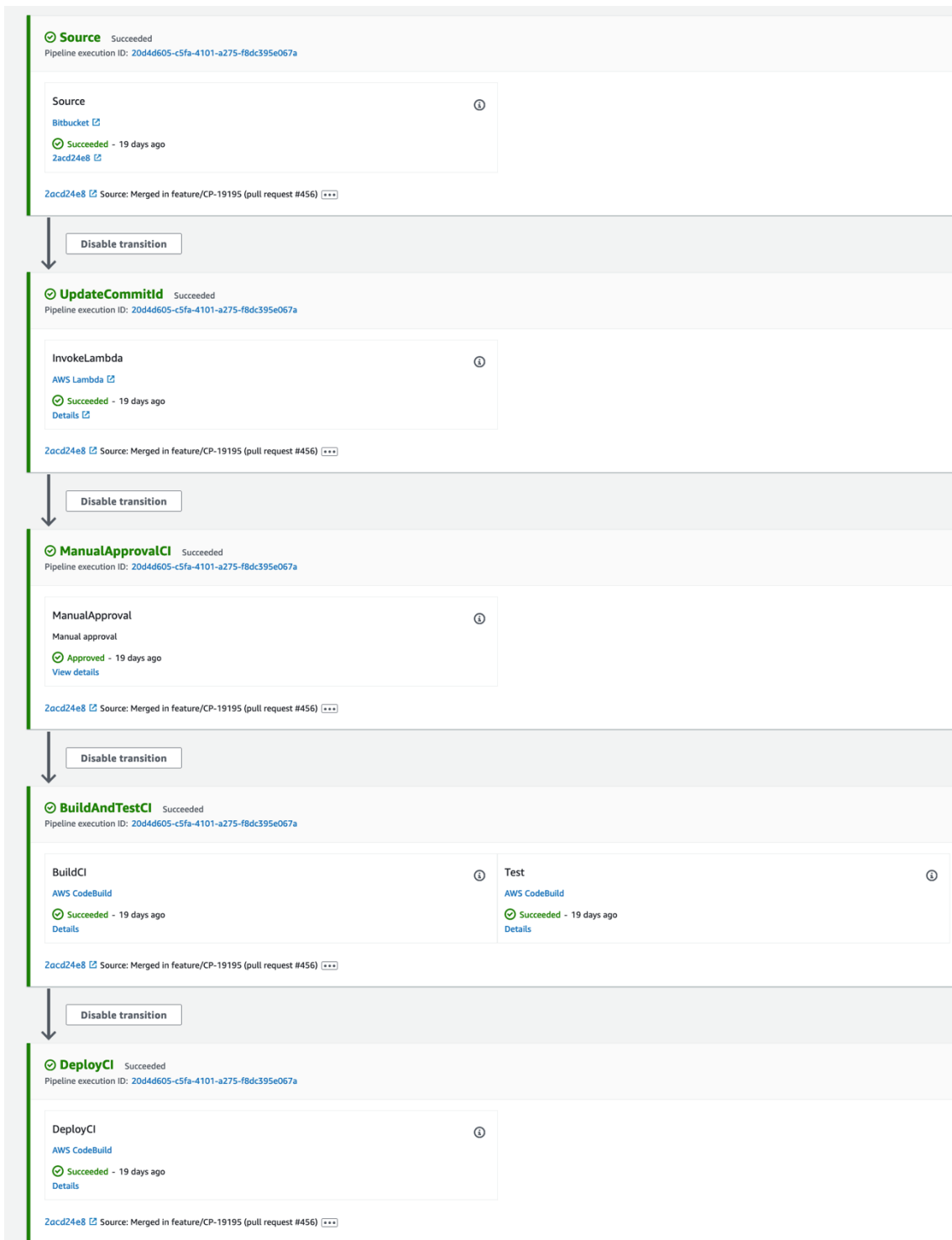


Figura 23: Ejecución pipeline para el despliegue la aplicación.

CAPÍTULO 5. FASE DE EJECUCIÓN

En este capítulo se detalla los diferentes sprint en el proceso de desarrollo.

5.1 Sprint 1

En este Sprint se realizó la historia de usuario por parte del administrador donde se pudo crear el acceso a la aplicación y la validación de datos ingresados por el usuario.

- **Objetivo del Sprint**

- Validar datos de ingresos
- Validar fotografía
- Validar correo

- **Sprint Backlog**

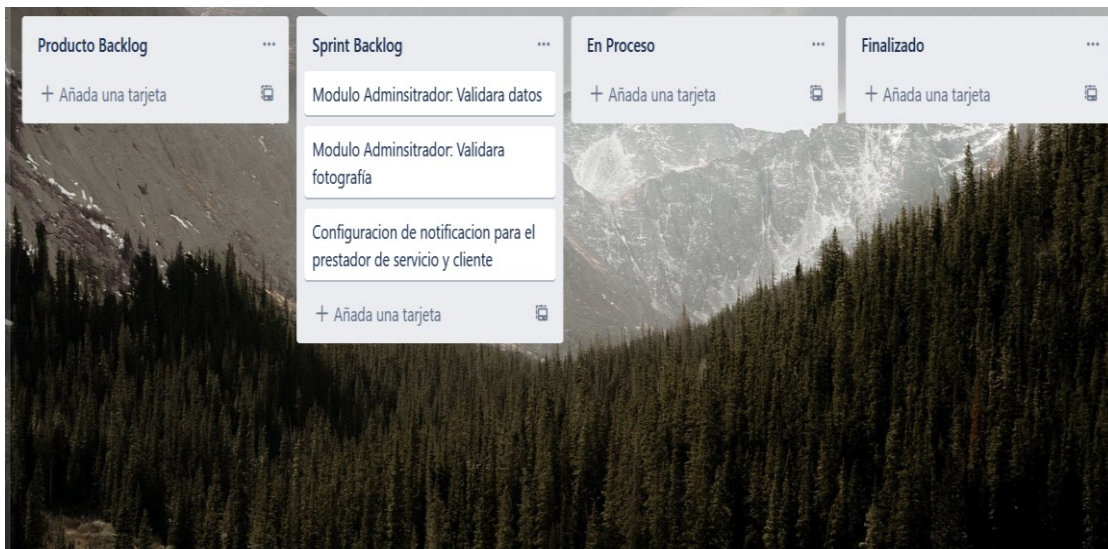


Figura 24: Sprint Backlog 1.

Tabla 11: Validar Datos

| Historia de usuario | |
|---|-----------------------------------|
| Número: 1 | Usuario: Administrador |
| Nombre historia: Validar datos de ingresos | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Alto |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario administrador quiero recibir toda la información de los usuarios que están creando cuentas para verificar que sean usuarios legítimos y no robots. | |
| Observación: El administrador solo trabaja desde el back-end. | |

Tabla 12: Validar fotografía

| Historia de usuario | |
|--|-----------------------------------|
| Número: 2 | Usuario: Administrador |
| Nombre historia: Validar fotografía | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Alto |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario administrador quiero recibir las fotografías ingresadas en la creación de cuentas para añadir la fotografía al portafolio interno de los usuarios registrados en la aplicación. | |
| Observación: El administrador solo trabaja desde el back-end. | |

Tabla 13: Validar correo

| Historia de usuario | |
|---|-----------------------------------|
| Número: 3 | Usuario: Administrador |
| Nombre historia: Validar correo | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Alto |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario administrador quiero que se valide el ingreso del correo para que una vez sea ingresado con éxito pueda enviarse un correo de activación de la cuenta. | |
| Observación: El administrador solo trabaja desde el back-end. | |

- **Ejecución del Sprint**

Se realizó los prototipos de las pantallas de la aplicación ya que el mismo fue solicitado por el Product Owner de la aplicación con la finalidad de revisar que los flujos estén correctos y no gastar tiempo de desarrollo hasta que todo esté aprobado a continuación en las siguientes figuras se muestra los mockups.



Figura 25: Diseño de login.

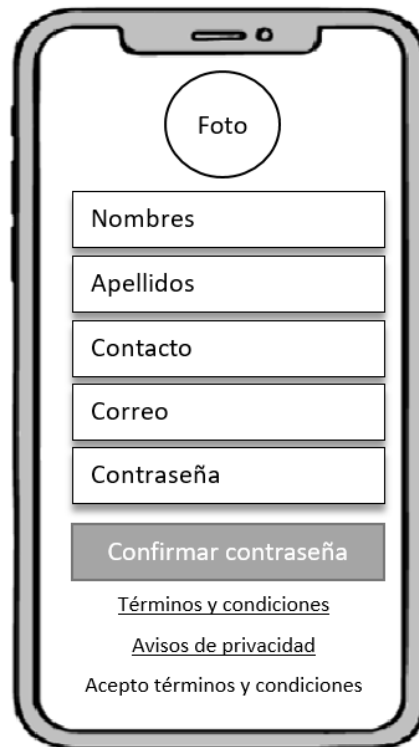


Figura 26: Registro de usuarios.



Figura 27: Código de validación de cuenta por correo.



Figura 28: Validación de cuenta por correo.



Figura 29: Servicios disponibles.

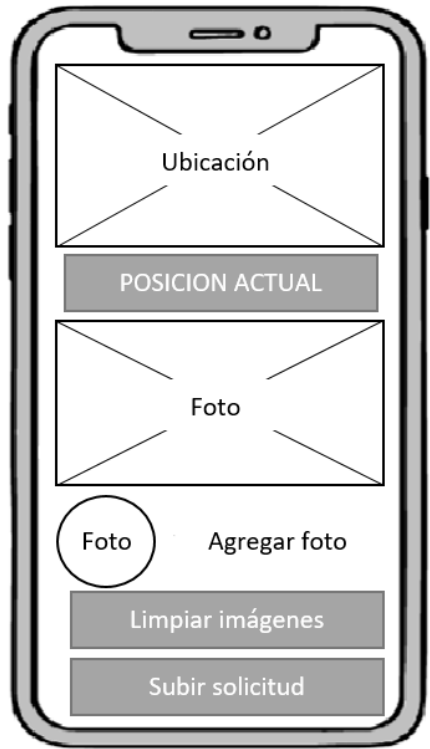


Figura 30: Solicitar servicio.



Figura 31: Reseñas de usuarios.

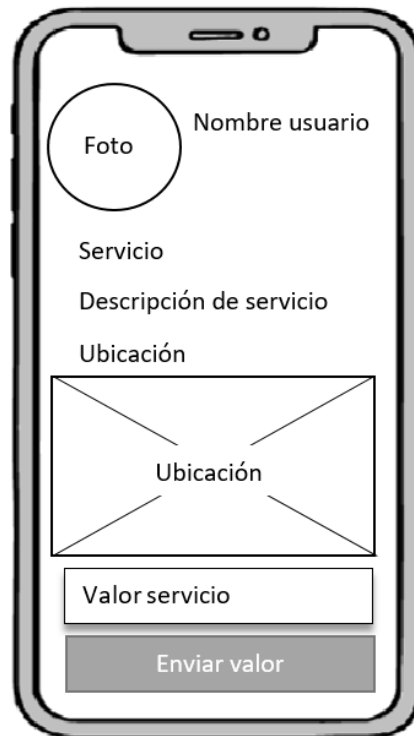


Figura 32: Valor de servicio.

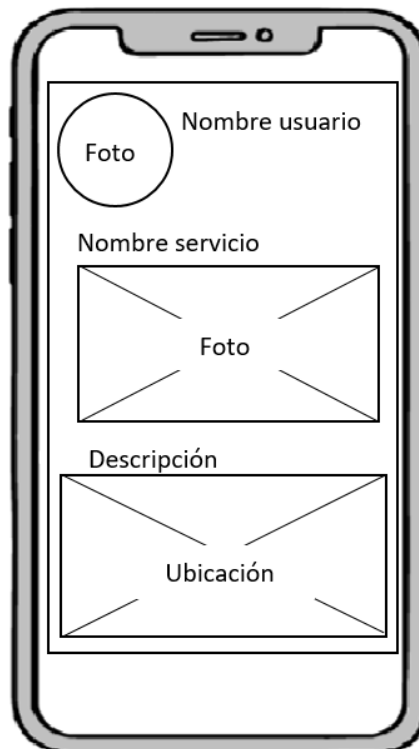


Figura 33: Notificación de servicio.



Figura 34: Chat de la aplicación.

Se diseñó la interfaz de inicio de sesión y registro del usuario de forma amigable para el usuario y así realizar la mínima cantidad de pasos para completar su inscripción. Esta pantalla está enfocada tanto para el prestador de servicios como el contratante de servicios la misma que consta de una serie de controles los cuales han sido programados previamente para que todos sus campos completados de forma obligatoria como también se valida el correo electrónico ingresado mediante un código que es enviado al mismo.

Función que valida el login sea correcto, no este repetido el usuario y puede ser verificado.

```
60 }
61
62 login(): void {
63   if (this.loginForm.valid) {
64     this.authService.loginUser(this.loginForm.value).subscribe({
65       next: (userInfo) => {
66         if (userInfo.codeValidation.expiresAt && userInfo.codeValidation.expiresAt?.toLowerCase() === 'd
67           this.widget.presentToast('La cuenta no ha sido verificada');
68         } else if (userInfo.activated) {
69           this.widget.presentToast('La cuenta no ha sido verificada');
70           this.router.navigate(['login/validate']);
71         } else if (userInfo) {
72           this.loadingService.create("Cargando información");
73           //this.showAlert();
74           const userData = userInfo;
75           sessionStorage.setItem('token', userInfo.token);
76           delete userData.token;
77           sessionStorage.setItem('ud', JSON.stringify(userData));
78           this.chatService.verifyUser();
79           this.worksService.getTypeOfUser();
80           this.navCtrl.navigateRoot('user/home');
81         }
82       },
83     },
84     error: (ERROR) => {
85       this.widget.presentToast(ERROR.error.message);
86     }
87   }
88 }
```

Figura 35: Función que valida el login

Diseño del html del login y registro de nuevos usuarios

```
loginpage.html - otros - visual studio code
main.component.html login.page.html x
src > app > pages > login > login.page.html > div.login
1 <div class="login">
2   <div class="login-container-logo">
3     <br>
4     
5   </div>
6   <form [formGroup]="loginForm" class="login-container">
7     <ion-item>
8       <ion-input required formControlName="email" placeholder="Email" type="email"></ion-input>
9       <ion-icon name="person" color="danger" slot="start"></ion-icon>
10    </ion-item>
11    <ion-item>
12      <ion-input #passwordEyeRegister required formControlName="password" placeholder="password" [type]-
13      <ion-icon slot="start" color="danger" name="lock-closed-outline"></ion-icon>
14      <button item-end class="btn_eye_icon" (click)="togglePasswordMode()"
15        <ion-icon slot="end" [name]="iconpassword"></ion-icon>
16    </button>
17  </ion-item>
18  <ion-button (click)="login()" color="danger" class="btn-expand">Iniciar Sesión</ion-button>
19  <div class="login-sign">
20    <a (click)="navigate('login/recoveryPassword')" class="white underline margin-bottom">¿Olvidaste tu
21    <ion-label class="white">¿Aun no tienes cuenta?</ion-label>
22    <a [routerLink]="['/register']" class="underline">Crear cuenta</a>
23  </div>
24 </form>
25 </div>
```

Figura 36: Diseño del html

Validación de campos requeridos

```
    this.registerForm = fb.group({
      name: new FormControl('', [Validators.required, Validators.minLength(3)]),
      lastName: new FormControl('', [Validators.required, Validators.minLength(3)]),
      contact: new FormControl(null, [Validators.required, Validators.minLength(10), Validators.maxLength(10)]),
      email: new FormControl('', [Validators.required, Validators.pattern(/^[\s@]+@[^\s@]+\.[^\s@]+$/)]),
      password: new FormControl('', [Validators.required,
        Validators.compose([
          CustomValidators.patternValidator(/\d/, { hasNumber: true }),
          CustomValidators.patternValidator(/[A-Z]/, { hasCapitalCase: true }),
          CustomValidators.patternValidator(/[\ `!@#%&*()_+~\-\[\]\{\};':"\\|,.<>\/?~]/, { hasSpecialCharacter: true })
        ])
      ]),
      passwordConfirmation: new FormControl('', Validators.required),
      termsAndConditions: new FormControl(false, Validators.requiredTrue),
      experience: this.fb.array([]),
    }, {
      validators: [
        matchValidate('password', 'passwordConfirmation')
      ]
    })
  }

  async selectImageSource() {
    const buttons = [
      {
        text: 'Camara',
        icon: 'camera',
        handler: () => {

```

Figura 37: Validación de campos requeridos.

Configuración de Notificaciones

```
    this.registerForm = fb.group({
      name: new FormControl('', [Validators.required, Validators.minLength(3)]),
      lastName: new FormControl('', [Validators.required, Validators.minLength(3)]),
      contact: new FormControl(null, [Validators.required, Validators.minLength(10), Validators.maxLength(10)]),
      email: new FormControl('', [Validators.required, Validators.pattern(/^[\s@]+@[^\s@]+\.[^\s@]+$/)]),
      password: new FormControl('', [Validators.required,
        Validators.compose([
          CustomValidators.patternValidator(/\d/, { hasNumber: true }),
          CustomValidators.patternValidator(/[A-Z]/, { hasCapitalCase: true }),
          CustomValidators.patternValidator(/[\ `!@#%&*()_+~\-\[\]\{\};':"\\|,.<>\/?~]/, { hasSpecialCharacter: true })
        ])
      ]),
      passwordConfirmation: new FormControl('', Validators.required),
      termsAndConditions: new FormControl(false, Validators.requiredTrue),
      experience: this.fb.array([]),
    }, {
      validators: [
        matchValidate('password', 'passwordConfirmation')
      ]
    })
  }

  async selectImageSource() {
    const buttons = [
      {
        text: 'Camara',
        icon: 'camera',
        handler: () => {

```

Figura 38: Código de Notificación.

Interfaz del login y registro de usuarios en la aplicación móvil la misma que verificará que todos los campos estén completos. También la fotografía es validada para que tome desde el dispositivo móvil o a su vez se suba una desde su galería.

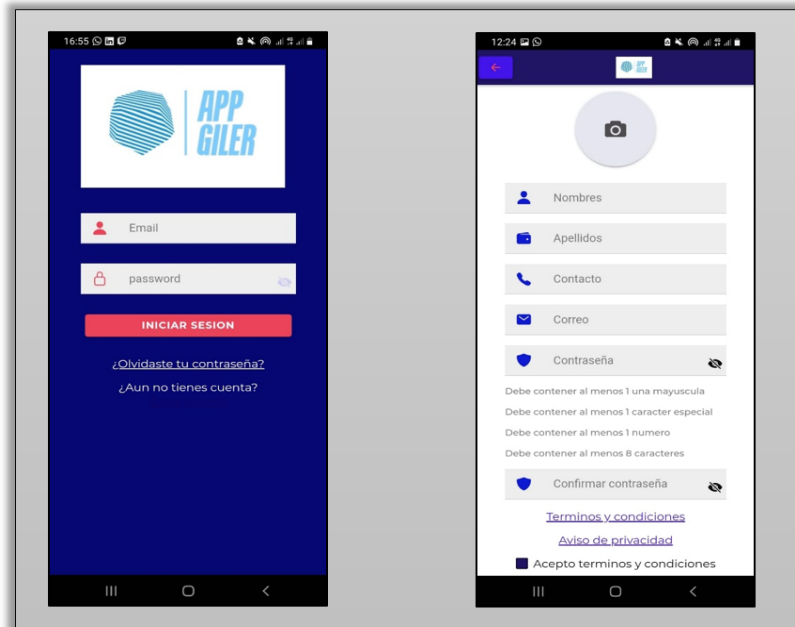


Figura 39: Interfaz gráfica del login.

Validación de correo electrónico el mismo que es verificado mediante un código que se le envía al correo electrónico ingresado, una vez que el correo es validado se le muestra en pantalla al usuario que se ha registrado de forma satisfactoria.

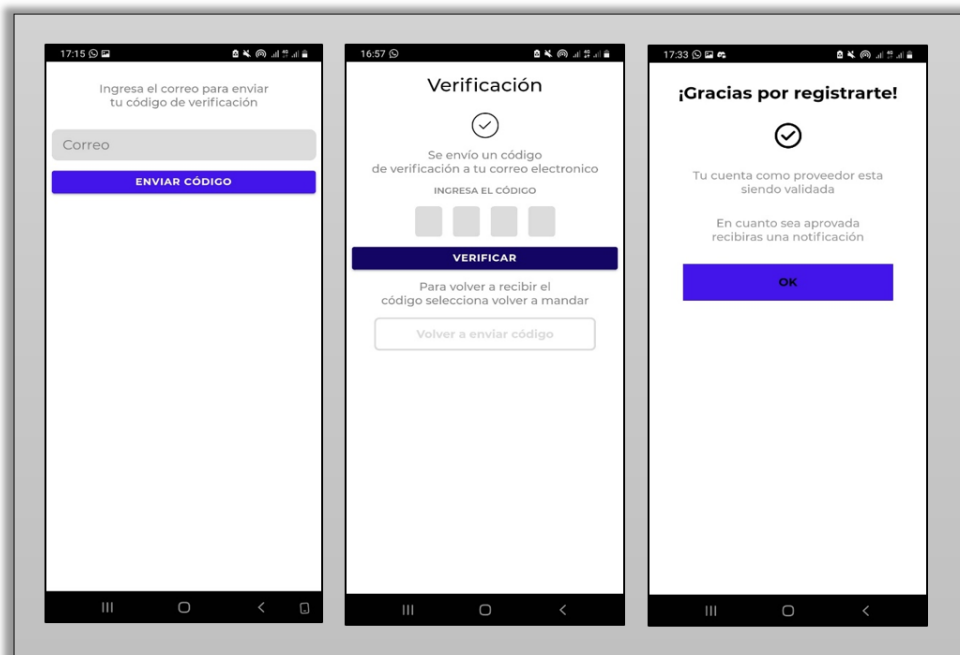


Figura 40: Validación por correo electrónico.

- **Revisión del Sprint**

Al final del sprint 1 se logró cumplir con los objetivos planteados, se diseñó la interfaz de inicio de sesión y registro de usuarios, validación de correo electrónico por código, validación de los datos que sean correctos, como también se configuro las notificaciones que le llegara al usuario prestador de servicios.

En la Figura 40 se muestra la captura de la revisión del sprint 1 en la herramienta trello:

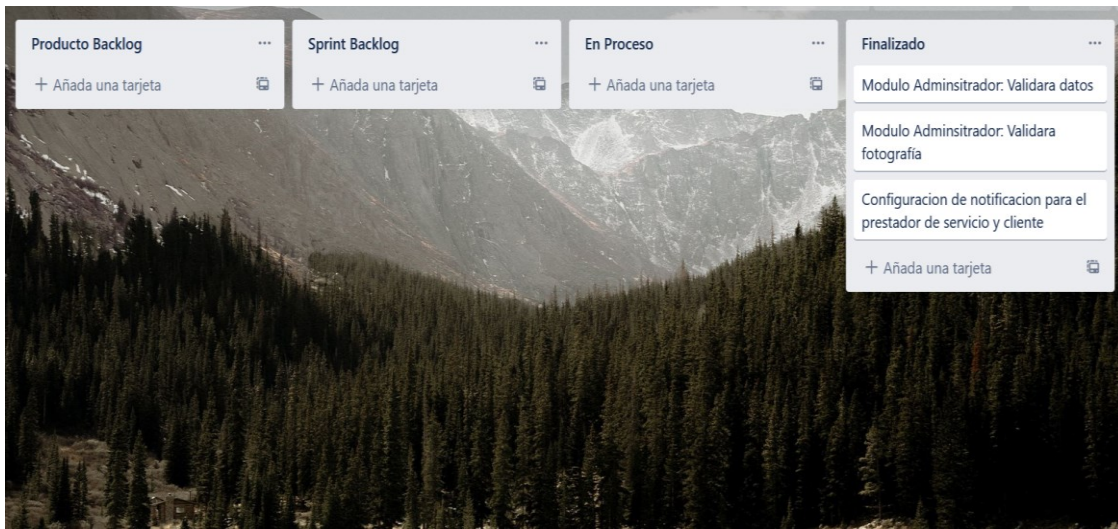


Figura 41: Revisión del sprint 1.

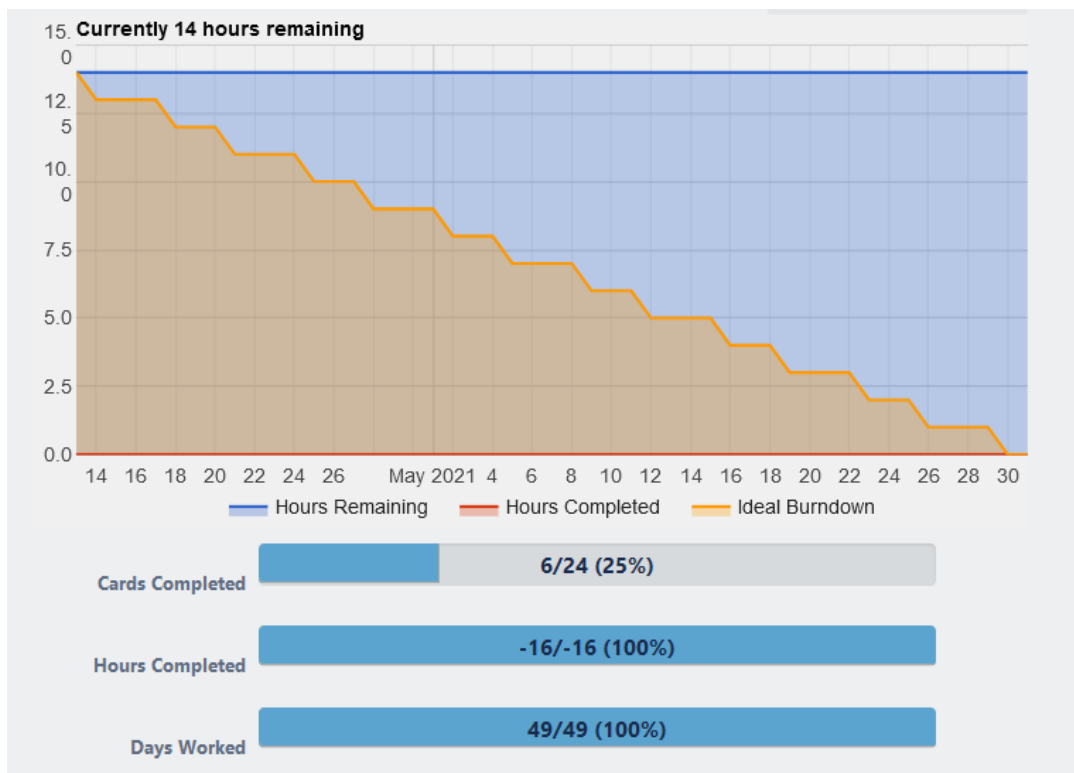


Figura 42: Análisis de la finalización del Sprint 1.

Con la finalización del Sprint 1 se culminaron las tareas planteadas en el desarrollo según las historias de usuario planificadas en este sprint, pero se tuvo un inconveniente en la validación de la fotografía ya que por ser una imagen se tuvo que indagar como almacenar en la base de datos, pero con una correcta búsqueda se pudo solucionar el problema encontrado en este Sprint.

5.2 Sprint 2

En este Sprint se realizó la historia de usuario por parte del prestador de servicios donde se configura la creación de la cuenta, se muestra los servicios que presenta y le llega una notificación cuando requieren de su servicio el mismo que al ser contactado por un cliente se muestra el chat donde interactúa con el mismo.

- **Objetivo del Sprint**

- Crear cuenta
- Verificar correo
- Iniciar sesión
- Consultar datos y reseñas de un servicio
- Crear nuevos servicios
- Envío de notificaciones de los servicios requeridos
- Chat entre los usuarios

- **Sprint Backlog**

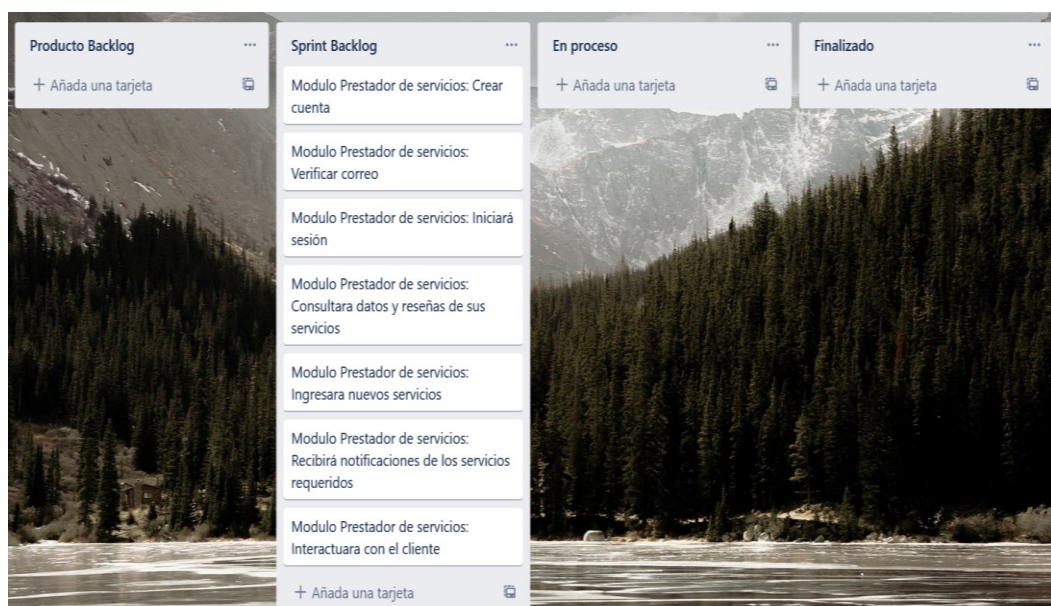


Figura 43: Sprint Backlog 2.

Tabla 14: Creación de cuenta del prestador de servicios

| Historia de usuario | |
|---|--|
| Número: 4 | Usuario: Prestador de servicios |
| Nombre historia: Crear cuenta | |
| Prioridad en Negocio: Baja | Riesgo en desarrollo: Medio |
| Puntos estimados: 1 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero crear una cuenta para registrar los oficios que puedo ofertar. | |
| Observación: Los datos deben ser reales para verificación de estos. | |

Tabla 15: Verificar correo

| Historia de usuario | |
|--|--|
| Número: 5 | Usuario: Prestador de servicios |
| Nombre historia: Verificar correo | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Medio |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero que al registrarme en el aplicativo se validen la información ingresada, en particular el correo para que una vez se envíe la información me pueda llegar correctamente las credenciales al correo ingresado en el sistema. | |
| Observación: El correo debe ser válido para la verificación mediante un código. | |

Tabla 16: Iniciar sesión

| Historia de usuario | |
|---|--|
| Número: 6 | Usuario: Prestador de servicios |
| Nombre historia: Iniciar sesión | |
| Prioridad en Negocio: Baja | Riesgo en desarrollo: Medio |
| Puntos estimados: 1 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero loguearme en la aplicación mediante mi correo para administrar la información de las diversas asistencias registradas. | |
| Observación: Iniciara sesión mediante el correo y clave ingresadas en el registro. | |

Tabla 17: Consultar datos y reseñas de sus servicios

| Historia de usuario | |
|---|--|
| Número: 7 | Usuario: Prestador de servicios |
| Nombre historia: Consultar datos y reseñas de un servicio | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero consultar el feedback de mis servicios que han sido evaluados por el cliente para tener retroalimentación del trabajo proporcionado. | |
| Observación: El usuario podrá verificar que sus datos personales sean los correctos. | |

Tabla 18: Ingresara nuevos servicios

| Historia de usuario | |
|--|--|
| Número: 8 | Usuario: Prestador de servicios |
| Nombre historia: Crear nuevos servicios | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero modificar descripciones o archivos de fotos de la información creada anteriormente para corregir errores de ortografía o mejorar las fotos para que se visualice mejor el servicio brindado | |
| Observación: El usuario podrá ingresar todos los servicios que pueda ofrecer sin limitante alguno. | |

Tabla 19: Recepción de notificación

| Historia de usuario | |
|--|--|
| Número: 9 | Usuario: Prestador de servicios |
| Nombre historia: Envío de notificaciones de los servicios requeridos | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero requiere recibir notificaciones de los servicios que son solicitados para que pueda estar pendiente de los nuevos pedidos de los servicios solicitados. | |
| Observación: Se notificara de los servicios pendientes mediante la aplicación. | |

Tabla 20: Interactuar con el cliente

| Historia de usuario | |
|--|--|
| Número: 10 | Usuario: Prestador de servicios |
| Nombre historia: Chat entre los usuarios | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario prestador de servicios quiero tener una comunicación con el cliente a través de un servicio de chat para intercambiar mensajes de acuerdos de negociación de la cita. | |
| Observación: Entablarán una comunicación por chat en la app con la persona que desea contratar su servicio. | |

- **Ejecución del Sprint**

Configuración del chat para la aplicación donde recibe los mensajes y procesa para mostrarse en la aplicación

```

54   buildLastPartOfChat(idUser, idWork){
55       this.chats.push({ idUser, idUser2: this.id, idWork, messages: [] });
56       this.buidDeepChat(idWork, idUser).map().once(conversation => this.addChat(conversation));
57   }
58
59   async verifyUser(){
60       const session = JSON.parse(sessionStorage.getItem('ud'));
61       if(session._id){
62           this.id = session._id;
63           this.messages$ = new BehaviorSubject<any>([]);
64           this.detectPages();
65           this.buidAllChats();
66       }
67   }
68
69   addChat(conversation): void {
70       if(!conversation.read && this.showMessageNotification){
71           this.widget.presentToast("Tienes un nuevo mensaje");
72           this.single_notification(conversation.message, conversation.nameUser);
73       }
74
75       this.chats[this.getIndex(conversation?.userId, conversation?.idWork)]?.messages.push(conversation);
76       this.messages$.next(this.chats);

```

Figura 44: Ejecución de sprint.

Interfaz gráfica donde se puede visualizar el login y perfil del usuario que presta los servicios de igual forma tiene la valoración del servicio presentado como también tenemos unos tabs en la parte inferior de la aplicación en la cual visualizara las notificaciones, los mensajes de los clientes y ver los servicios que son solicitados.

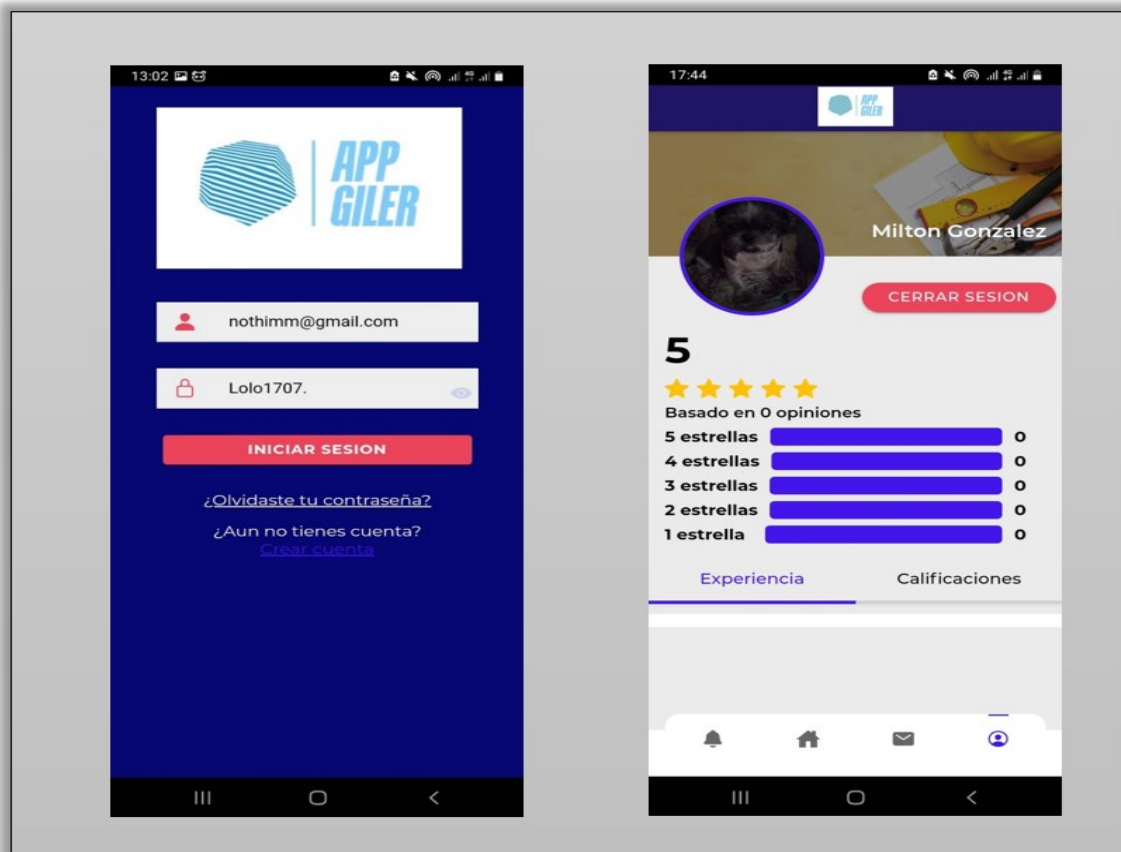


Figura 45: Interfaz gráfica de login y perfil.

Interfaz gráfica donde el usuario observa las notificaciones que tiene pendientes de los clientes que requieren el servicio que ofrece.

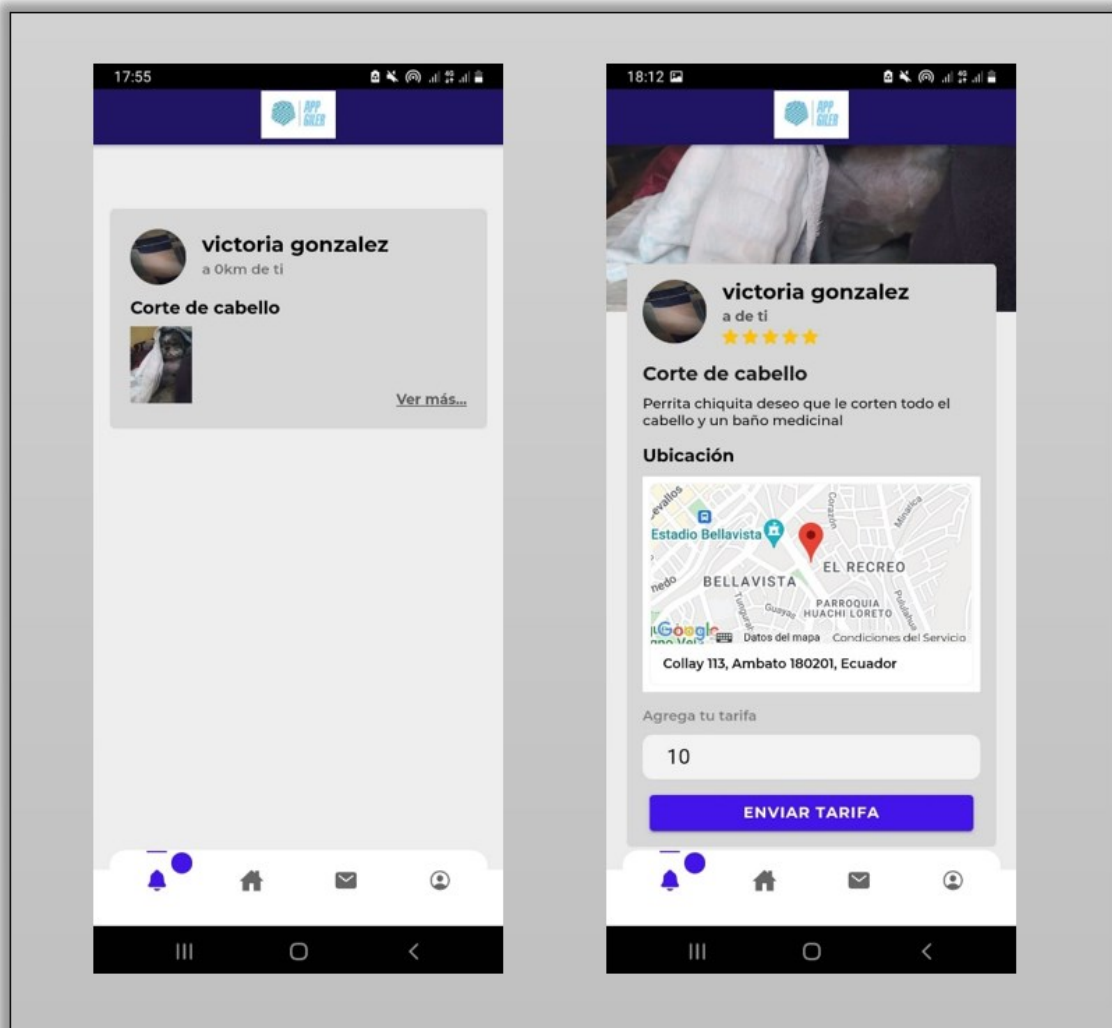


Figura 46: Interfaz gráfica de notificaciones y google maps.

Interfaz gráfica del chat donde interactúan con el cliente y arreglan el valor de sus servicios donde el cliente acepta contratar o no en caso de que sea contratado le llega una notificación de la ubicación y la foto donde se encuentra la persona.

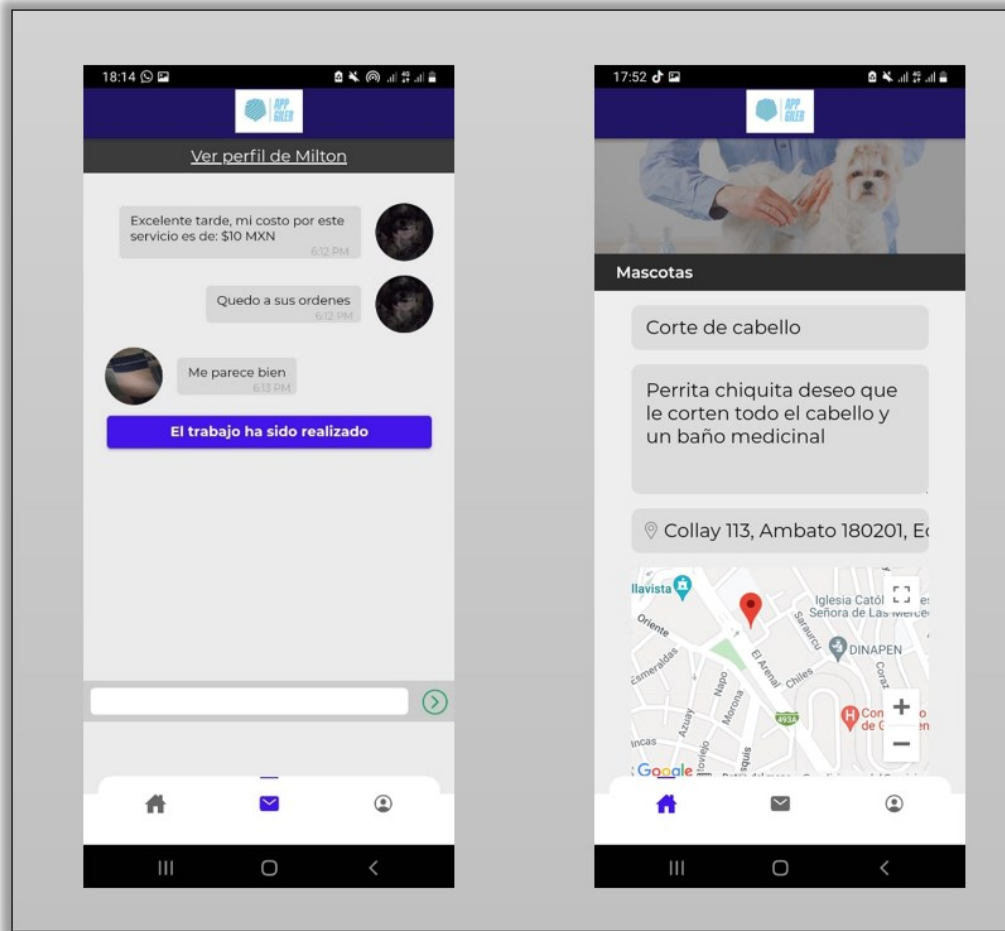


Figura 47: Interfaz gráfica del chat.

- **Revisión del Sprint**

Al final del Sprint 2 se logró cumplir con los objetivos planteados, donde se configuro la creación de la cuenta por parte del usuario prestador de servicios, en la misma que al momento de ingresar sus datos le llega una notificación que su usuario ha sido validado y se configuro la cuenta donde muestra los servicios que presta que al momento que un cliente requiera contratarlo le llega una notificación a la app donde se contactaran por medio del chat de la aplicación donde se llegaran a un acuerdo por el valor del servicio.

En caso de llegar a un acuerdo y el cliente lo contrate la app le enviará una foto y la ubicación donde se encuentra el cliente de esta manera cuando ya se realiza la contratación el servicio se desactiva de la aplicación, finalmente el cliente podrá calificarlo de una forma positiva o negativa según el servicio presentado.

En la Figura 47 se muestra la captura de la revisión del sprint 2 en la herramienta trello:

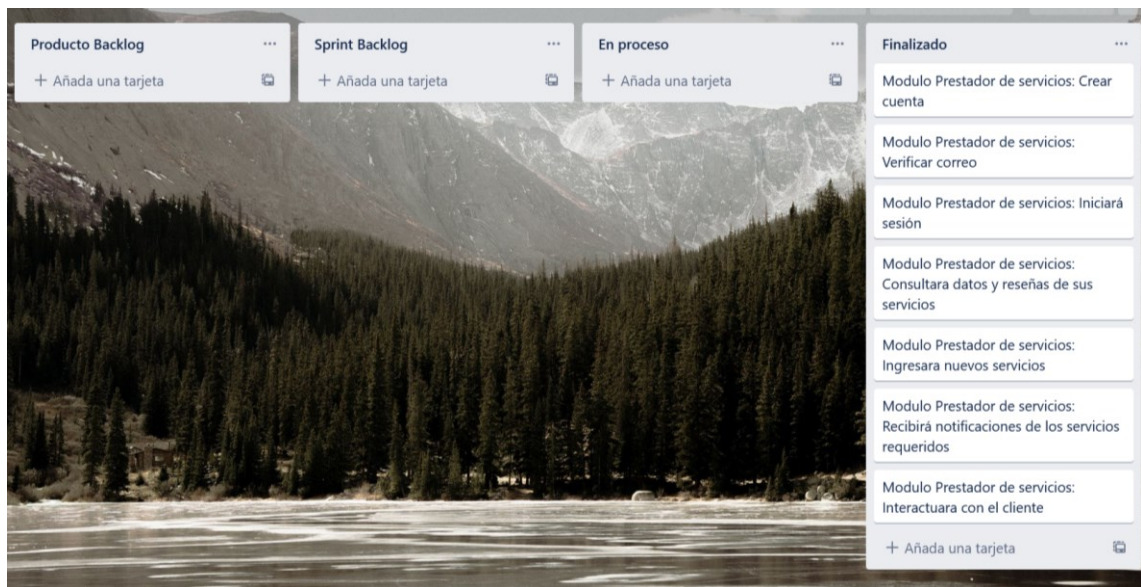


Figura 48: Revisión del Sprint 2.

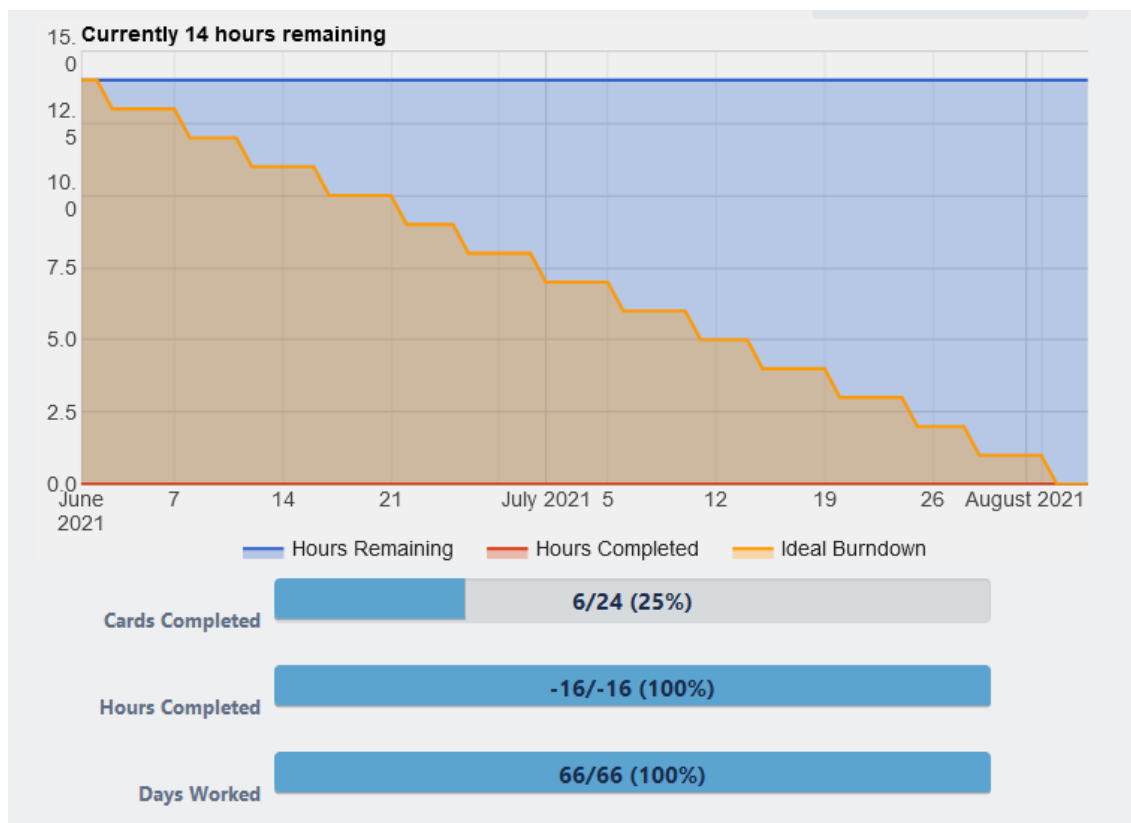


Figura 49: Análisis de la finalización del Sprint 2.

Se desarrollaron las tareas planteadas para el Sprint 2 con normalidad, pero se tuvo que realizar una investigación sobre la implementación del chat para interactuar con los clientes como también se presentó el inconveniente de como adquirir la ubicación de los contratistas

como de profesionales que prestan los servicios. Pero se solucionó los inconvenientes presentados en esta Sprint para seguir avanzando en el desarrollo de la aplicación móvil. Finalmente se realizaba pruebas de funcionamiento del código para poder llevar una buena calidad del sistema mientras se iba avanzando con el desarrollo del software.

5.3 Sprint 3

En este Sprint se realizó la historia de usuario por parte del contratante de servicios donde se configura la creación de la cuenta el mismo que podrá solicitar un servicio el mismo que tendrá que subir una fotografía y la ubicación donde se encuentra localizado de la misma manera podrá buscar en la aplicación, de igual forma podrá revisar el perfil de la persona que presta el servicio y puede interactuar con el profesional por medio de un chat de la aplicación.

- **Objetivo de Sprint**

- ❖ Crear cuenta
- ❖ Verificar información
- ❖ Iniciar sesión
- ❖ Calificar servicio
- ❖ Consultar sus datos
- ❖ Solicitar servicio
- ❖ Consultar servicios que existentes
- ❖ Contratar servicio

- **Sprint Backlog**

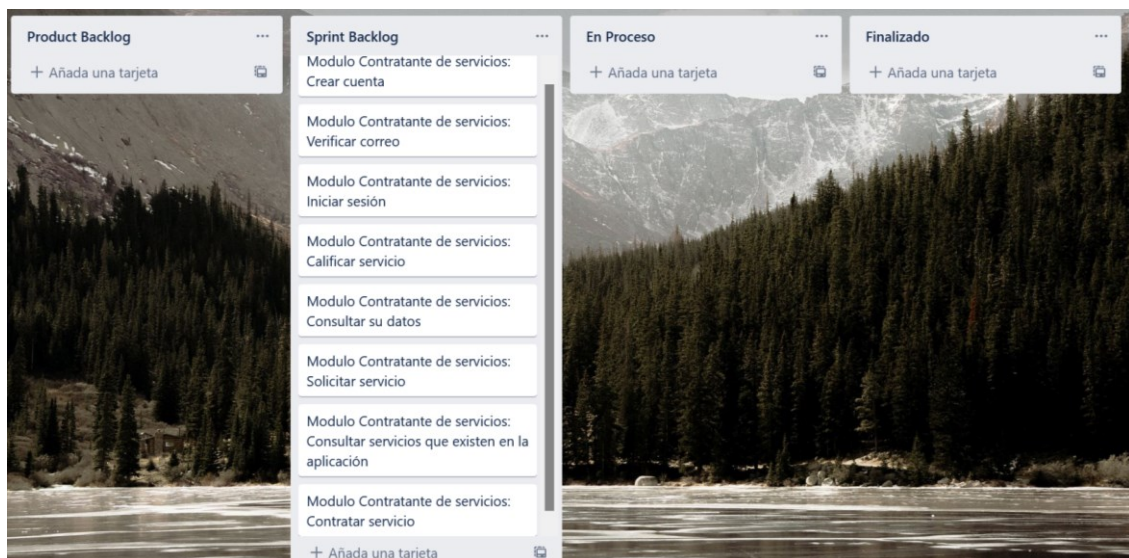


Figura 50: Sprint Backlog 3.

Tabla 21: Crear cuenta

| Historia de usuario | |
|--|--|
| Número: 11 | Usuario: Contratante de servicios |
| Nombre historia: Crear cuenta | |
| Prioridad en Negocio: Baja | Riesgo en desarrollo: Medio |
| Puntos estimados: 1 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero crear una cuenta para poder adquirir una cita de un servicio que necesito. | |
| Observación: Los datos deben ser reales para verificación de los mismos. | |

Tabla 22: Verificar correo

| Historia de usuario | |
|--|--|
| Número: 12 | Usuario: Contratante de servicios |
| Nombre historia: Verificar información | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Medio |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero que al instante de registrarme en el aplicativo se validen la información ingresada para que no pueda ingresar correos o información distinta a lo que se piden en los datos de ingreso. | |
| Observación: El correo debe ser válido para que se pueda autenticar la cuenta. | |

Tabla 23: Iniciar sesión

| Historia de usuario | |
|---|--|
| Número: 13 | Usuario: Contratante de servicios |
| Nombre historia: Iniciar sesión | |
| Prioridad en Negocio: Baja | Riesgo en desarrollo: Medio |
| Puntos estimados: 1 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero iniciar sesión en la aplicación mediante mi correo para solicitar y agendar una cita de los servicios que requiero. | |
| Observación: Deberá iniciar sesión con el correo y contraseña ingresadas en su registro. | |

Tabla 24: Calificar contratista

| Historia de usuario |
|---------------------|
|---------------------|

| | |
|---|--|
| Número: 14 | Usuario: Contratante de servicios |
| Nombre historia: Calificar servicio | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Medio |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero calificar el servicio prestado por parte del profesional para que otros usuarios puedan tener una referencia del tipo de servicio que ofrece ya sea bueno o malo. | |
| Observación: La calificación y reseña se dará dependiendo el servicio prestado. | |

Tabla 25: Consultar sus datos personales

| | |
|---|--|
| Historia de usuario | |
| Número: 15 | Usuario: Contratante de servicios |
| Nombre historia: Consultar sus datos personales | |
| Prioridad en Negocio: Media | Riesgo en desarrollo: Medio |
| Puntos estimados: 2 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero consultar los datos del usuario que ofrece los servicios profesionales para verificar la identidad del profesional. | |
| Observación: Se consultara sus datos personales mediante una pantalla en la aplicación móvil. | |

Tabla 26: Recepción de notificación

| | |
|--|--|
| Historia de usuario | |
| Número: 16 | Usuario: Contratante de servicios |
| Nombre historia: Solicitar servicio | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero solicitar un servicio profesional que requiero para solventar una incidencia que poseo como cliente. | |
| Observación: Se ingresará sus datos y el tipo de servicio que requiere contratar. | |

Tabla 27: Consultar servicios que existen en la aplicación

| Historia de usuario | |
|---|--|
| Número: 17 | Usuario: Contratante de servicios |
| Nombre historia: Consultar servicios que existen en la aplicación | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero consultar servicios que existen en la aplicación para solicitar alguno que cumpla con mis necesidades de cliente. | |
| Observación: Navegara por la aplicación buscando el servicio que requiere contratar. | |

Tabla 28: Contratar servicio

| Historia de usuario | |
|---|--|
| Número: 18 | Usuario: Contratante de servicios |
| Nombre historia: Contratar servicio | |
| Prioridad en Negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 3 | Iteración asignada: 1 |
| Programador responsable: Mario Giler | |
| Descripción: Como usuario cliente quiero registrar mi posición actual del lugar de residencia para que el usuario prestador de servicios pueda observar la ubicación en el mapa. | |
| Observación: Contratará un servicio una vez que haya llegado un acuerdo con el profesional. | |

- **Ejecución del Sprint**

En esta parte del código se valida la imagen que el contratante suba ya que es un campo requerido para la prestación del servicio como también se debe ingresar la ubicación donde se encuentra.

```
src > app > pages > register > TS register.page.ts > RegisterPage > verifyForm > forEach() callback
155 tryregister(){
156   if(this.registerForm.valid && this.base64Image){
157     this.router.navigate(['/register1']);
158     this.userCreateProvider.page1Data(this.registerForm, this.base64Image);
159   }else{
160     this.verifyForm(this.registerForm);
161     if(!this.base64Image){
162       this.widget.presentToast('Debes de subir una imagen');
163     }
164   }
165 }
166
167 verifyForm(form: FormGroup): void {
168   let countErrors = 0;
169   let flagTerms = false;
170   if (form.valid) {
171     this.widget.presentToast('Debes de subir una imagen');
172   } else {
173     Object.keys(form.value).forEach(name => {
174       if (name === 'termsAndConditions' && form.get(name).invalid){
175         flagTerms = true;
176       }else if (form.get(name).invalid) {
177         countErrors++;
178         form.get(name).markAsDirty();
179       }
180     });
181     this.widget.presentToast((countErrors > 0) ? 'Completa los campos' : flagTerms ? 'Debes de aceptar los terminos' : '');
182   }
183 }
```

Figura 51: Código para validar la imagen.

Interfaz gráfica donde se muestra la navegación por la aplicación móvil la cual tiene varios servicios que ofrece como también se muestra el perfil del cliente que requiere contratar un servicio, el perfil de contratante cuenta con tres tabs ubicados en la parte inferior de la aplicación los mismo que tienen la función de mostrar los servicios presentes, los mensajes que tiene y su perfil de usuario.

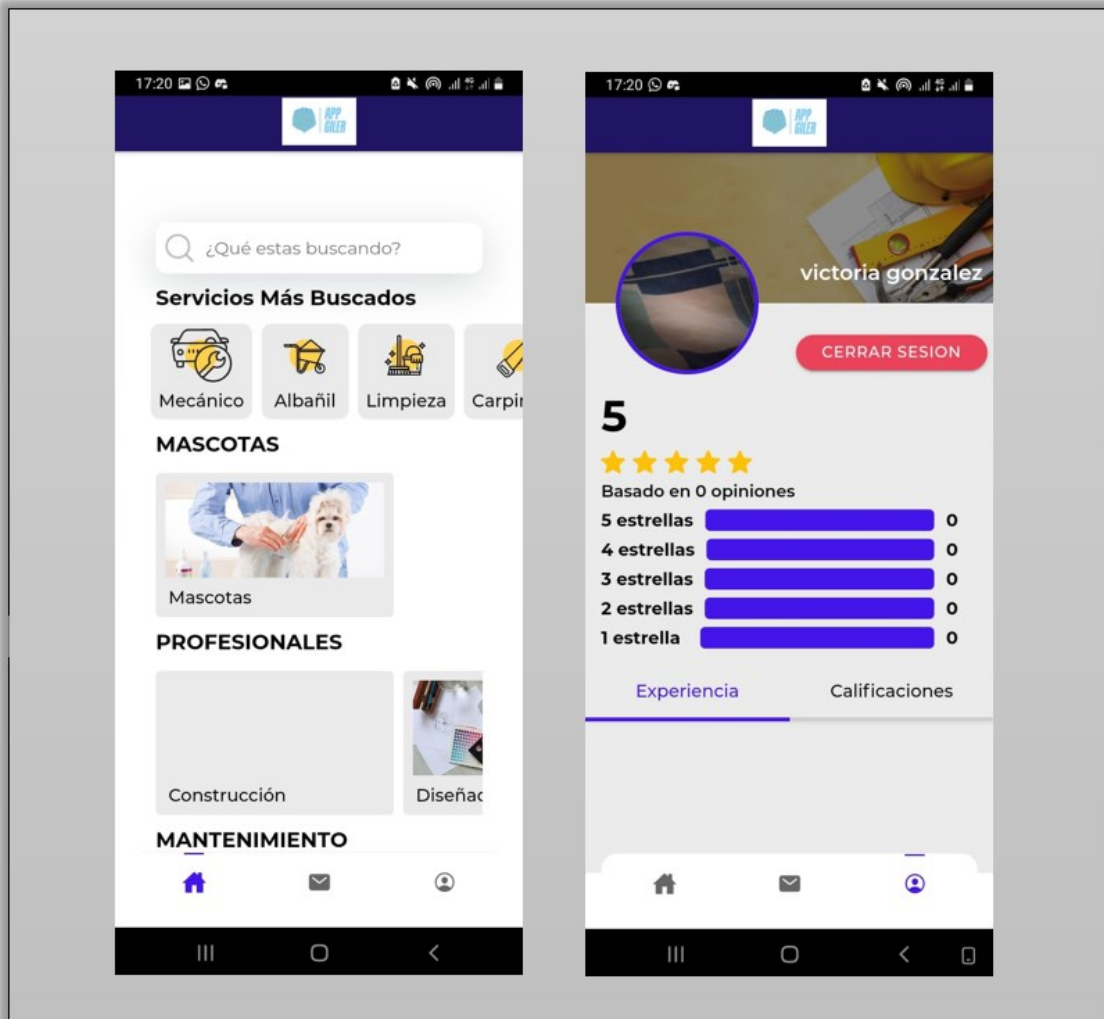


Figura 52: Interfaz gráfica de la navegación por la aplicación móvil.

Interfaz gráfica donde se muestra la solicitud que se ingresa para pedir el servicio donde ingresamos la foto, la ubicación la misma que sera vista con un profesional y se contactara mediante el chat de la aplicación móvil donde se llegará a un acuerdo y se procederá a contratarlo de igual forma se puede revisar el perfil del profesional donde se podrá ver las reseñas de sus anteriores trabajos.

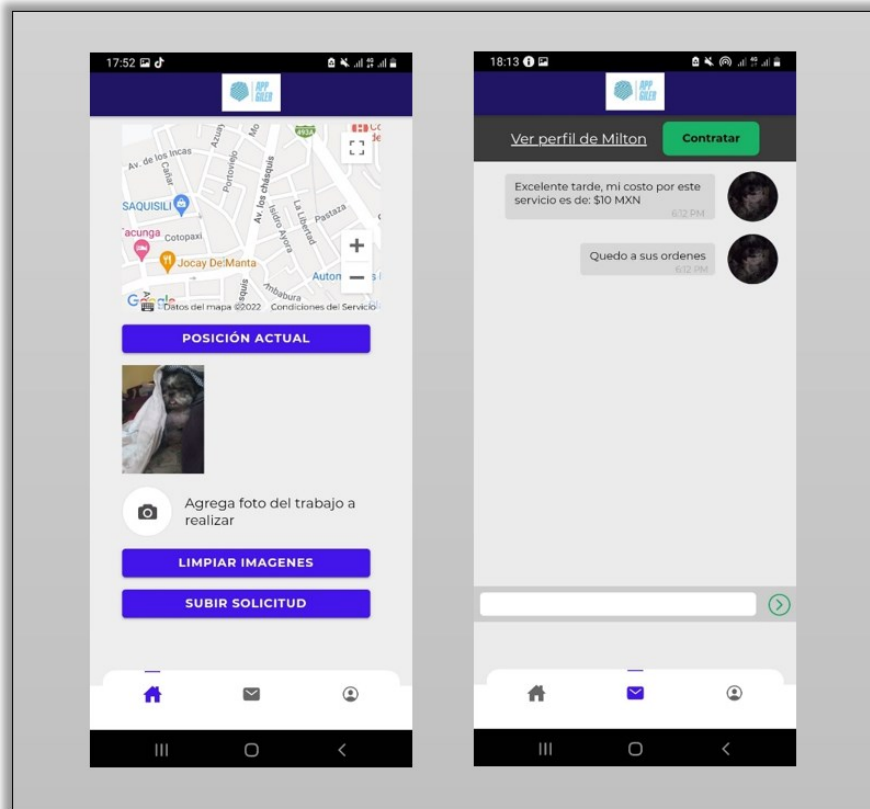


Figura 53: Interfaz gráfica para solicitud que se ingresa para pedir el servicio.

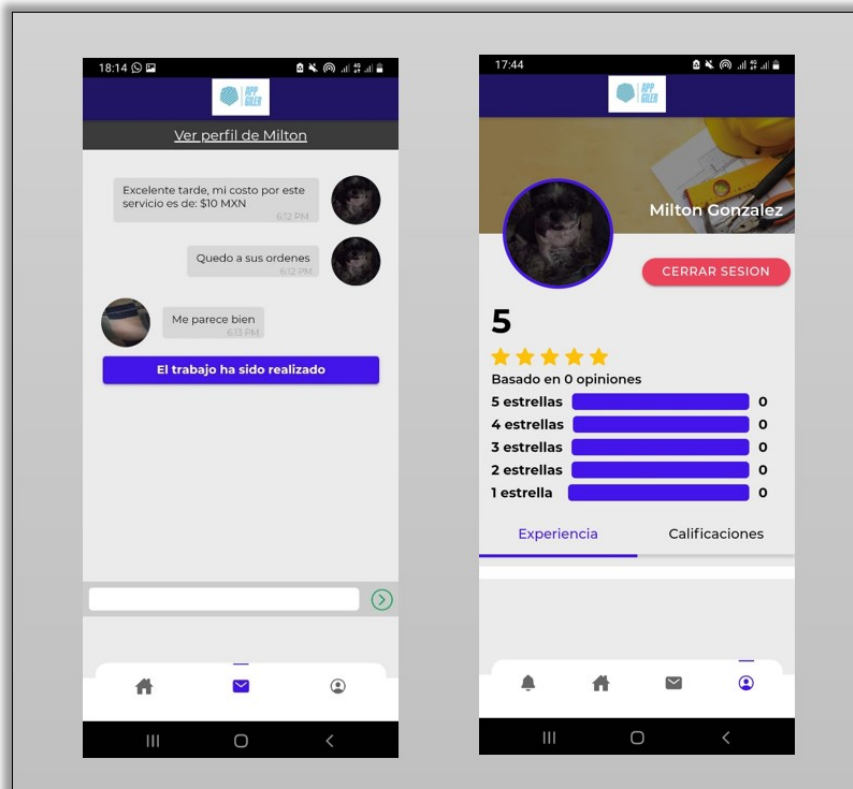


Figura 54: Interfaz gráfica del chat de la aplicación móvil.

- **Revisión del Sprint**

Al final del Sprint 3 se logró cumplir con los objetivos planteados, donde se configuro la creación de la cuenta por parte del usuario contratante de servicios, el mismo que se registrara en la aplicación móvil donde podrá navegar por la aplicación donde se mostraran los distintos tipos de servicios que existe donde podrá realizar un pedido de un servicio que requiere con la fotografía y su ubicación donde un profesional se contactará con él por medio del chat interno de la aplicación y se pondrán de acuerdo con el servicio en caso de llegar a contratarlo se dará por finalizado el servicio, de igual forma el contratante podrá acceder al perfil del profesional donde podrá ver las reseñas de anteriores trabajos realizados como también podrá calificar del mismo.

En la Figura 54 se muestra la captura de la revisión del sprint 3 en la herramienta trello:

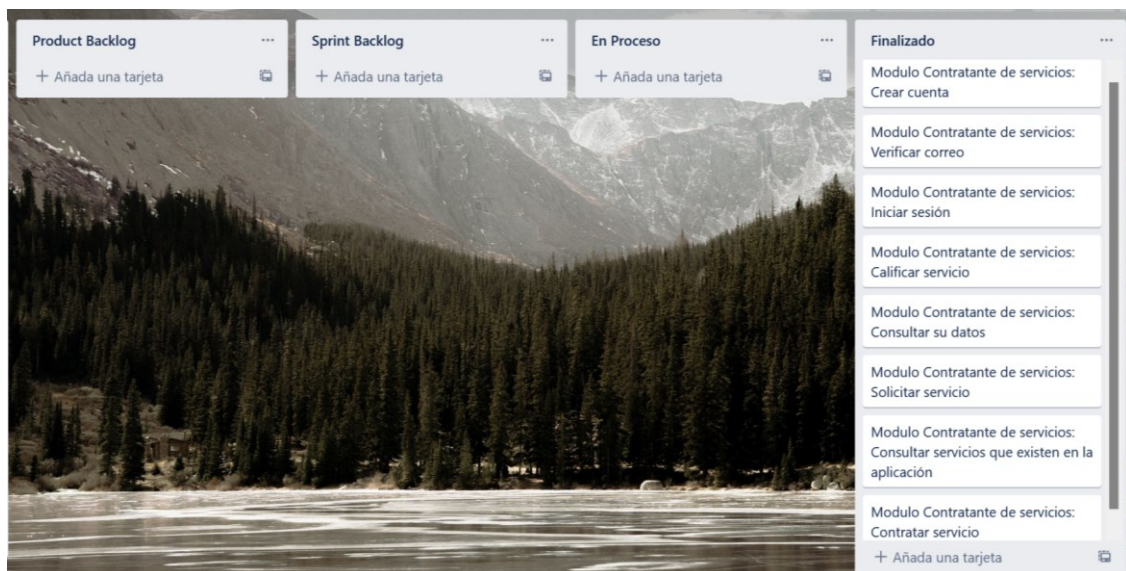


Figura 55: Revisión del Sprint 3.

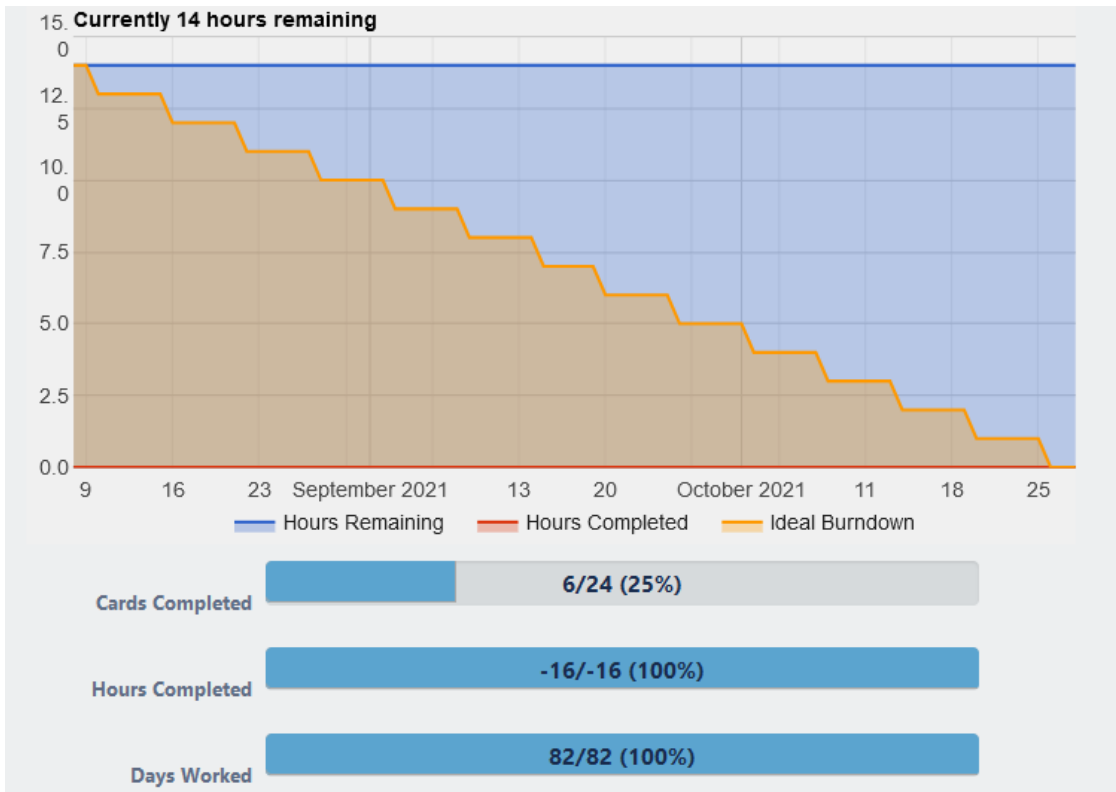


Figura 56: Análisis de la finalización del Sprint 3.

Finalmente se culminó las tareas del Sprint 3 con normalidad dando por finalizado todas las tareas que se plantearon para el desarrollo de la aplicación móvil, como también se desarrollaron todas las pruebas de la aplicación para poder validar que los servicios estén funcionando con normalidad y pasar a la siguiente etapa la cual es realizar la prueba de usabilidad donde se podrá probar la aplicación móvil a fondo y observar que tan intuitiva es su manejo por parte del usuario.

CAPÍTULO 6. ANALISIS DE FUNCIONALIDAD Y USABILIDAD

En este capítulo se detalla el análisis tanto de la funcionalidad como de la usabilidad de la aplicación.

6.1 Prueba de usabilidad

Para la realización de usabilidad de la aplicación móvil se realizó mediante la encuesta de SUS (System Usability Scale) [28].

System Usability Scale (SUS), es una herramienta independiente de la tecnología la cual sirve para probar la usabilidad de hardware, software, sitio web, aplicaciones móviles, teléfonos, etc.

La prueba de usabilidad se realizar con un cuestionario de 10 preguntas cada una con 5 opciones de respuesta. [28], como muestra la tabla 26

Tabla 29: Preguntas de cuestionario de usabilidad

| Pregunta | Siendo 1 difícil y 10 fácil | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| ¿Qué tan fácil es registrarse en la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil es iniciar sesión en la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil fue registrar un servicio en la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil fue buscar un servicio requerido? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil fue calificar un servicio en la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil fue usar el chat de la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ¿Qué tan fácil fue publicar la experiencia de un servicio contratado en la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Pregunta | Siendo 1 poco probable y 10 muy probable | | | | | | | | | |
| ¿Utilizaría esta aplicación para contratar servicios requeridos para una siguiente ocasión? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Pregunta | Siendo 1 difícil de usar y 10 fácil de usar | | | | | | | | | |
| En términos generales, ¿Cómo califica la experiencia de uso de la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Pregunta | Siendo 1 mala y 10 buena | | | | | | | | | |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| En términos generales, ¿Cómo califica la experiencia visual de la aplicación? | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|----|

Para obtener los resultados de usabilidad se tiene que sumar los resultados promediados obtenidos de los cuestionarios realizados a los clientes, tomando en cuenta las siguientes recomendaciones.

- Las preguntas impares tomaran el valor asignado por el usuario y se le restara 1
- Las preguntas pares serán de menos 5 el valor asignado por los usuarios
- Una vez obtenido los resultados, será multiplicado por 2,5

6.1.1 Resultados de la encuesta SUS

La encuesta de SUS fue realizado a un total de 10 empleados de la empresa, los mismo que fueron los voluntarios para realizar los casos de prueba en la sección anterior.

En la Tabla 47, se puede ver el detalle de los resultados del cálculo de cada pregunta por cada encuestado. Las respuestas de las encuestas realizadas a los usuarios se encuentran en el Anexo.

Una vez calculado el promedio de cada pregunta, se suma los valores para luego multiplicar por 2,5. Dando el resultado de usabilidad de 75/100 el mismo que demuestra que la usabilidad del sistema es muy aceptable ya que el mínimo recomendado por los encuestadores de SUS es de 70/100.

Tabla 30: Resultados de la Encuesta SUS

| Encuestados | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
|-----------------|-----------|------------|-----------|------------|-----------|------------|------------|------------|------------|------------|
| Autor 1 | 10 | 10 | 10 | 9 | 10 | 9 | 10 | 10 | 10 | 7 |
| Autor 2 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 9 | 10 | 9 |
| Autor 3 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 8 |
| Autor 4 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 9 | 10 | 8 |
| Autor 5 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 9 |
| Autor 6 | 10 | 10 | 10 | 10 | 10 | 10 | 9 | 10 | 8 | 8 |
| Autor 7 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 9 | 9 |
| Autor 8 | 10 | 10 | 10 | 10 | 10 | 10 | 8 | 10 | 8 | 8 |
| Autor 9 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| Autor 10 | 10 | 9 | 10 | 9 | 10 | 10 | 9 | 9 | 9 | 7 |
| Promedio | 10 | 9,9 | 10 | 9,8 | 10 | 9,9 | 9,6 | 9,7 | 9,4 | 8,3 |

Al finalizar las pruebas de usabilidad por medio de una encuesta aplicada a los empleados de la empresa MANTICORE LABS, se visualizó que las 10 preguntas tuvieron una puntuación buena por parte de los empleados sobre la usabilidad de la app, demostrando que es una aplicación intuitiva y fácil de utilizar.

6.2 Prueba de funcionalidad

Para las pruebas de funcionalidades se usó la herramienta de automatización de pruebas RUNSCOPE que facilita la resolución rápida de problemas de la API mediante un conjunto de pruebas automatizadas, monitoreo y depuración.

A continuación, se detalla los casos de prueba que fueron ejecutados por el pipeline creado.

Prueba de funcionalidad 1.- API inicialización del agendamiento de una cita.

En la Figura 56 y 57 se insertaron datos pruebas para corroborar la respuesta del API

| Available Variables | |
|-------------------------|----------------------------------|
| {{bank_id}} | 1022 |
| {{base_url}} | |
| {{currency}} | MXN |
| {{document_number}} | 123456789 |
| {{document_type}} | NIT |
| {{env}} | qa |
| {{mail}} | |
| {{name}} | test name |
| {{payment_description}} | Pago prueba |
| {{phone}} | 0946382736 |
| {{private_merchant_id}} | d77a08bdfc7a47d898ab2e1f571f6fa4 |
| {{public merchant id}} | a5108cab472f498592b2f38b44c2d35c |

Figura 57: Prueba de funcionalidad 1.1

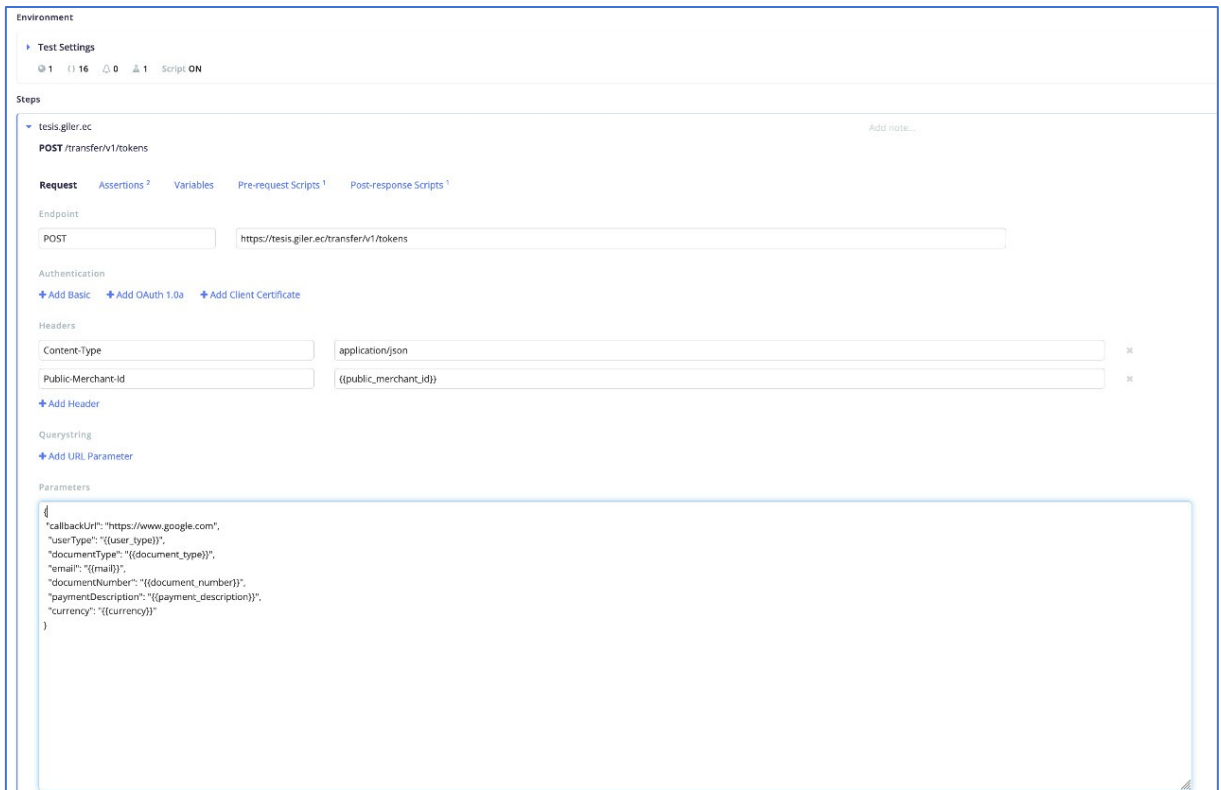


Figura 58: Prueba de funcionalidad 1.2

Prueba de funcionalidad 2.- API consulta del agendamiento de una cita.

En la Figura 58 se ingresaron datos pruebas para validar que la API responde correctamente al consultar.

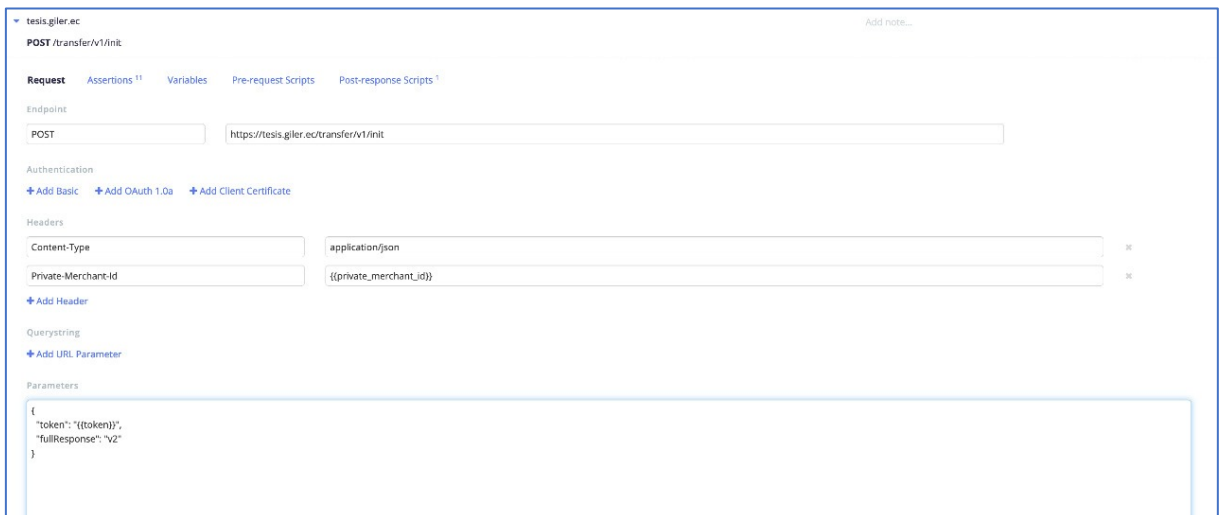


Figura 59: Prueba de funcionalidad 2

Prueba de funcionalidad 3.- Prueba de navegación de la aplicación.

En la Figura 59 se consume el endpoint donde se encuentra la aplicación para evaluar su navegación y su diseño funcional.

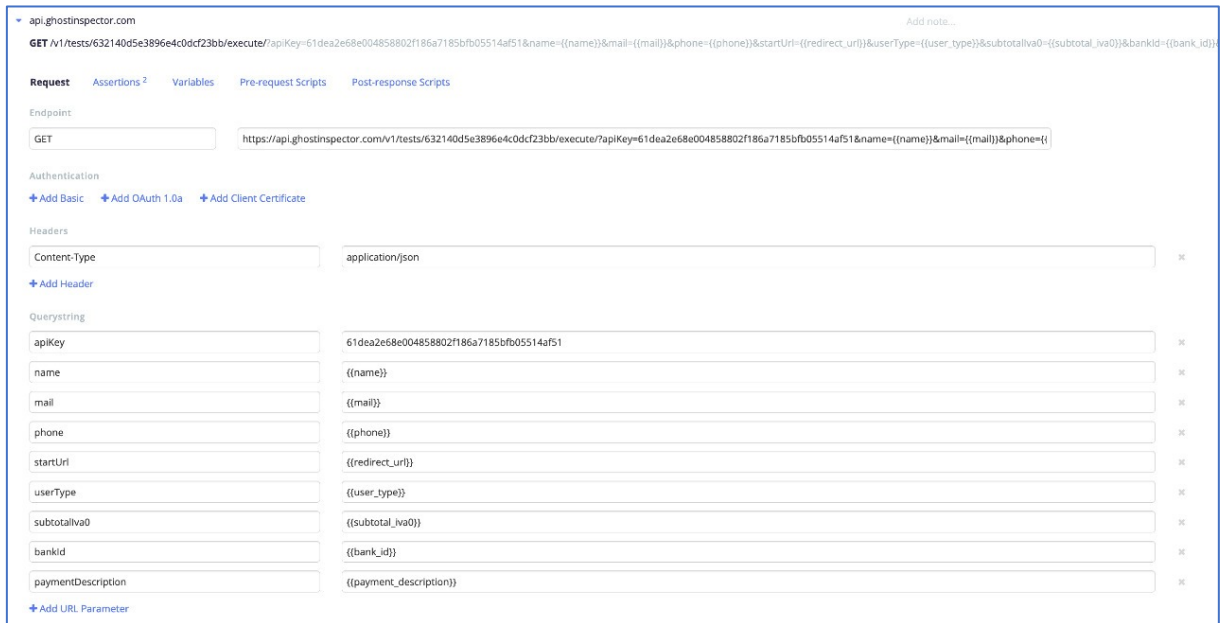


Figura 60: Prueba de funcionalidad 3

Resultados del conjunto de pruebas.

En la Figura 60 exponemos los distintos resultados de los conjuntos de pruebas realizados.



Figura 61: Resultado de pruebas realizadas

Evolución de las pruebas.

En la Figura 61 exponemos la evolución de los distintos casos de pruebas realizados.

| Recent Test Runs | |
|------------------|--|
| ✔ Passed | Sep 14 9:13pm US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 12:44pm US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:42am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:33am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:16am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:13am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:10am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 9:07am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 8:51am US California via Dashbo... ⌵ Test Settings |
| ✔ Passed | Sep 14 8:47am US California via Dashbo... ⌵ Test Settings |

Figura 62: Evolución de pruebas.

CAPÍTULO 7. CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- El uso de webhook, que es un sistema de comunicación automático para el intercambio de datos entre aplicaciones, permitió la entrega continua. Cada vez que el equipo de desarrollo efectuaba un commit, se ejecutaban las pruebas unitarias, para desplegar el software en los respectivos ambientes de forma ágil, segura y confiable.
- Emplear un pipeline mediante diferentes fases tanto de construcción, como de pruebas permitió alojar un conjunto de pruebas alojadas en un repositorio para ser ejecutadas como la fase final y así limitar que nuevos cambios al aplicativo se desplieguen con errores al ambiente de producción. De tal manera, bloquea entregas de cambios y se revisaba que algún cambio no afectara flujos principales de la aplicación.
- El desarrollo de PWA permitió que el software funcione correctamente tanto en aplicaciones web como en aplicaciones móviles, mediante el uso de los service worker, que son secuencias de comandos ejecutados por un navegador en segundo plano. Dado que la aplicación no es grande se logró estabilidad y robustez semejante a una aplicación nativa.
- Mediante herramientas de CI/CD como archivos de comandos se pudo mejorar las configuraciones de los repositorios y las credenciales para unir la fuente de los proyectos y la compilación de cada uno de manera cíclica.
- Se desarrolló una aplicación móvil híbrida en el Framework de desarrollo IONIC, misma que facilitará al usuario en sus tareas manuales, como la búsqueda de servicios que necesiten realizar cotidianamente. La aplicación tiene geolocalización que permite al usuario buscar servicios de acuerdo con su ubicación, para agendar la cita de acuerdo a su preferencia.
- Las encuestas de usabilidad determinan que la aplicación móvil es intuitiva, fácil de utilizar, tiene un correcto funcionamiento y cumple con todas las tareas planteadas.
- La aplicación móvil facilitará a los usuarios que prestan servicios, publicar el tipo de trabajo que realizan por medio de la aplicación, con la finalidad de crecer su popularidad, así como también el número de clientes. Los usuarios podrán comunicarse por un chat integrado en la aplicación e interactuar con sus clientes antes de agendar el servicio.

RECOMENDACIONES

Para el correcto funcionamiento de la aplicación móvil para la gestión de servicios se recomienda:

- Al momento de plantear la arquitectura es transcendental analizar que la base de datos con la que trabaja el sistema no es escalable, por tanto, si la aplicación crecer en número de usuarios se tendrán problemas de rendimiento.
- El aplicativo tiene un diseño responsive, para los sistemas operativos Android e IOS, por tanto, las ventanas se despliegan correctamente en los dispositivos que manejan estos sistemas operativos, sin embargo, no se ha analizado en sistemas Android e IOS con versiones inferiores del año 2010, por ello se recomienda probarlos y de ser el caso adaptar las pantallas para estos dispositivos.
- Para realizar un escalamiento, mejoras al rendimiento o nuevas funcionalidades al aplicativo, se recomienda usar frameworks, que permitan crear y administrar recursos escalables a nivel de código.
- Para administrar las incidencias de manera ordenada se recomienda usar una nomenclatura clara para diferenciar una corrección o mejora al aplicativo.
- En caso de que se necesite alterar la información en la base de datos como un nuevo campo integrar el script de modificación al pipeline para que se ejecute dentro del ciclo de integración continua.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Manticore-Labs, «Facebook,» 28 Junio 2018. [En línea]. Available: <https://es-la.facebook.com/manticorelabsec/>.
- [2] Manticore Labs, «Manticore-labs,» 28 Junio 2018. [En línea]. Available: <https://manticore-labs.com/>.
- [3] Agencia de Regulación y control de telecomunicaciones, «Arcotel.gob.ec,» Junio 2017. [En línea]. Available: <http://www.arcotel.gob.ec/464-de-usuarios-del-servicio-movil-avanzado-poseen-un-smartphone/>.
- [4] D. Garayar, «Desarrollo de una aplicación web basado en la metodología Scrum - Devops para la gestión de contratación del servicio docente de la UGEL ANDAHUAYLAS - Región Apurímac 2018,» Andaguaylas, 2018.
- [5] C. Bello, «¿Sabes realmente qué es DevOps?,» 30 Enero 2017. [En línea]. Available: xeridia.com/blog/sabes-realmente-que-es-devops.
- [6] I. Farías, «Definición de un ambiente de construcción de aplicaciones empresariales a través de Devops, Microservicios y Contenedores,» Loja, 2017.
- [7] «Repositorio Institucional - UNAJMA,» 27 08 2022. [En línea]. Available: <https://repositorio.unajma.edu.pe/handle/20.500.14168/491>.
- [8] «Repositorio Institucional de la UTP,» 07 02 2022. [En línea]. Available: <http://dspace.utpl.edu.ec/jspui/handle/20.500.11962/21205>.
- [9] G. Jiménez:, «DevOps, la nueva tendencia en el desarrollo de sistemas TI, un caso práctico en el análisis de incidencias de software,» 2016.
- [10] «¿Qué es DevOps?,» 14 02 2022. [En línea]. Available: <https://www.netapp.com/es/devops-solutions/what-is-devops/>.
- [11] K. Terán, «Análisi y formulación de estrategia general para preparar la adopción de la cultura DevOps en matrices bancarias de la ciudad de Guayaquil,» Guayaquil, 2020.
- [12] «Guía rápida para entender el Cuadrante de Gartner y evaluar tus opciones en tecnología,» 07 07 2022. [En línea]. Available: <https://www.gb-advisors.com/es/cuadrante-de-gartner/>.
- [13] J. Díaz y M. Romero, «Desarrollo e implementación de un aplicativo web, utilizando la metodología Scrum, para mejorar el procesos de atención al cliente en la empresa Z aditivos S.A.,» Lima, 2017.
- [14] M. Trigas, «Metodología Scrum,» 2016.
- [15] «Scrum: roles y responsabilidades,» 08 08 2022. [En línea]. Available: <https://www2.deloitte.com/es/es/pages/technology/articles/roles-y-responsabilidades-scrum.html>.

- [16] «Compara Software,» 05 05 2022. [En línea]. Available: <https://blog.comparasoftware.com/fases-metodologia-scrum/>.
- [17] «Introducción a las metodologías ágiles: su significado, ventajas y desventajas, tipos y otros datos imprescindibles,» 05 05 2022. [En línea]. Available: <https://blog.ganttpro.com/es/introduccion-a-las-metodologias-agiles-1/>.
- [18] «Ingenio Learnig,» 14 04 2022. [En línea]. Available: <https://ingenio.edu.pe/blog/elementos-basicos-para-dominar-scrum/>.
- [19] «Scrum Manager,» 27 07 2022. [En línea]. Available: https://www.scrummanager.com/files/scrum_manager_historias_usuario.pdf.
- [20] M. Rituay, «Implementación de una aplicación web y el aumento de las conversiones en una Inmobiliaria,» Perú, 2020.
- [21] A. Brunella, D. Bergalli, I. Genta y I. Toledo, «Plataforma de integración con mercado libre,» Uruguay, 2020.
- [22] «Aplicaciones Web Ventajas y Beneficios,» 22 05 2022. [En línea]. Available: <https://neosystems.es/noticias/aplicaciones-web-6-ventajas-y-beneficios/>.
- [23] F. Ibáñez, «Desarrollo de una progressive web app PWA para gestionar plataformas tecnológicas modeladas con BPM,» 2019.
- [24] «PWA: ¿Qué es una Progressive Web App?,» 23 04 2022. [En línea]. Available: <https://www.kiwop.com/pwa-que-es-una-progressive-web-app>.
- [25] N. Miranda y G. Torres, «Implementación de una aplicación web progresiva PWA para la identificación y notificación de mascotas extraviadas usando placas con código de respuesta rápida QR en collares,» Guayaquil, 2018.
- [26] A. Velasco y J. Dután, «Desarrollo de una aplicación web progresiva PWA para determinar si un autobus urbano esta en su límite de capacidad de pasajeros,» Quito, 2021.
- [27] «Service Worker - Desarrollo Web,» 17 05 2022. [En línea]. Available: <https://desarrolloweb.com/articulos/service-workers.html>.
- [28] R. A. Grier, A. Bangor, P. Kortum y S. C. Peres, «The System Usability Scale: Beyond Standard Usability Testing,» 30 Septiembre 2013. [En línea]. Available: <https://journals.sagepub.com/doi/abs/10.1177/1541931213571042>. [Último acceso: 07 abril 2022].
- [29] J. Díaz, «Tooit.com,» [En línea]. Available: <https://tooit.com/es/pwa-multiplataforma-cms/>.
- [30] «Guía rápida para entender el Cuadrante de Gartner y evaluar tus opciones en tecnología,» 22 07 2022. [En línea]. Available: <https://www.gb-advisors.com/es/cuadrante-de-gartner/>.

- [31] «Firebase - Cloud Firestore,» 4 10 2021. [En línea]. Available: <https://firebase.google.com/docs/firestore?hl=es-419>.
- [32] Google, «Cloud Storage para Firebase,» 27 02 2022. [En línea]. Available: <https://firebase.google.com/docs/storage>.
- [33] «NodeJS,» 23 04 2022. [En línea]. Available: <https://nodejs.org/es/>.
- [34] Y. Muradas, «Qué es NPM y para qué sirve,» OpenWebinars, 03 04 2022. [En línea]. Available: <https://openwebinars.net/blog/que-es-node-package-manager/>.
- [35] «React,» 23 04 2022. [En línea]. Available: <https://es.reactjs.org/>.
- [36] «Section - How to Implement Material-UI in React,» 16 10 2021. [En línea]. Available: <https://www.section.io/engineering-education/how-to-implement-material-ui-in-react/>.
- [37] «mui-treasury,» 23 04 2022. [En línea]. Available: <https://mui-treasury.com/>.
- [38] «notistack,» 23 04 2022. [En línea]. Available: <https://www.npmjs.com/package/notistack>.
- [39] «material-ui-dropzone,» 23 04 2022. [En línea]. Available: <https://www.npmjs.com/package/material-ui-dropzone>.
- [40] N. Sebastian, «The React Scripts Start Command,» 23 04 2022. [En línea]. Available: [https://www.freecodecamp.org/news/create-react-app-npm-scripts-explained/#:~:text=Create%20React%20App%20\(CRA\)%20is,run%20it%20on%20the%20browser..](https://www.freecodecamp.org/news/create-react-app-npm-scripts-explained/#:~:text=Create%20React%20App%20(CRA)%20is,run%20it%20on%20the%20browser..)
- [41] «Formik,» 23 04 2022. [En línea]. Available: <https://formik.org/>.
- [42] «Wertarbyte,» 23 04 2022. [En línea]. Available: <https://v0.mui.wertarbyte.com/>.
- [43] Learus, «React Material UI Carousel,» 23 04 2022. [En línea]. Available: <https://learus.github.io/react-material-ui-carousel/>.
- [44] «Redux,» 23 04 2022. [En línea]. Available: <https://es.redux.js.org/>.
- [45] «React Router,» 23 04 2022. [En línea]. Available: <https://www.javatpoint.com/react-router>.

ANEXOS

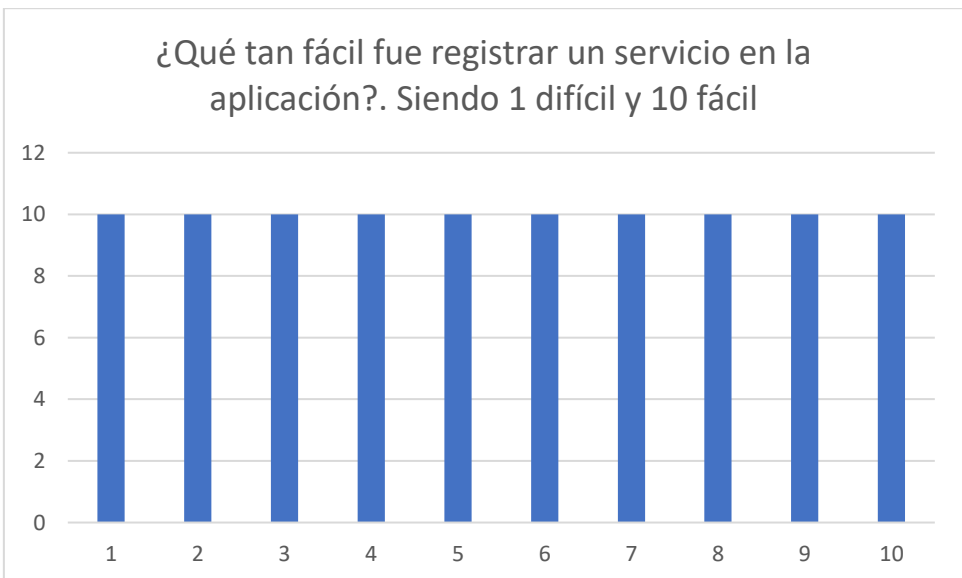
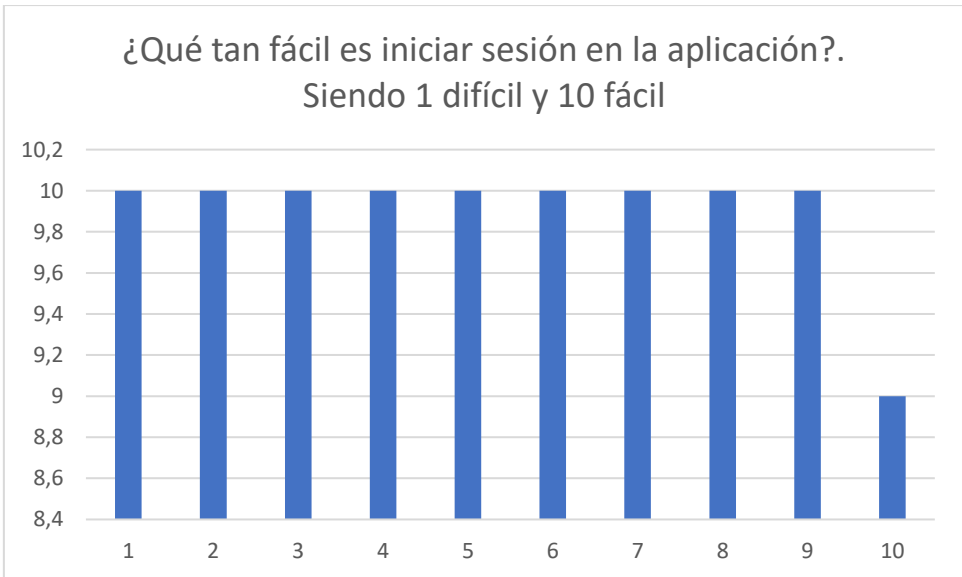
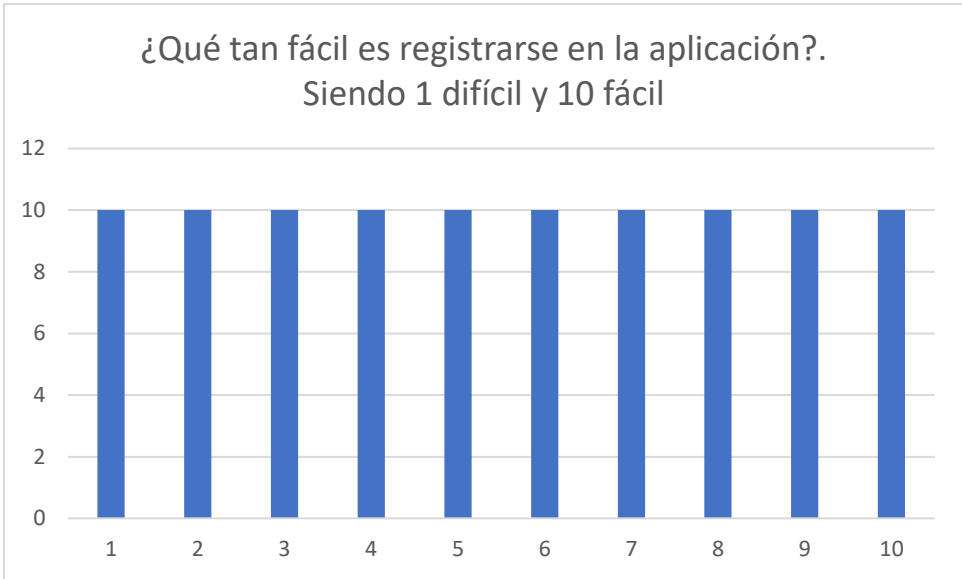
Link de encuesta a los empleados de la empresa

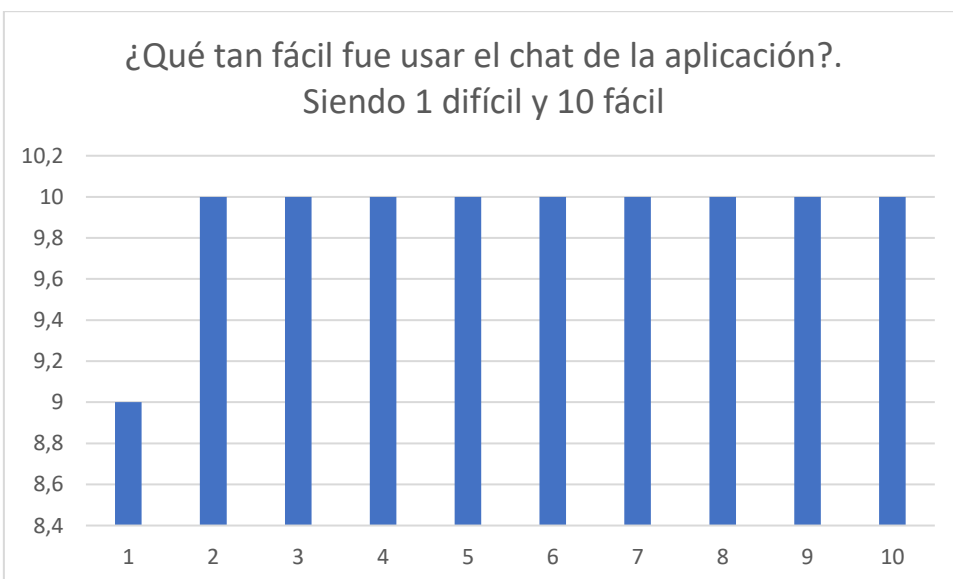
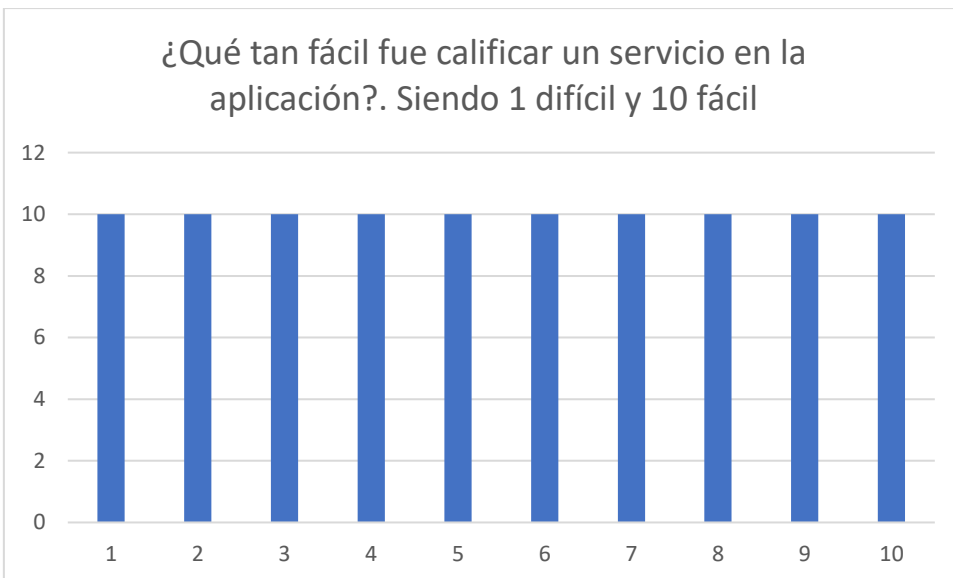
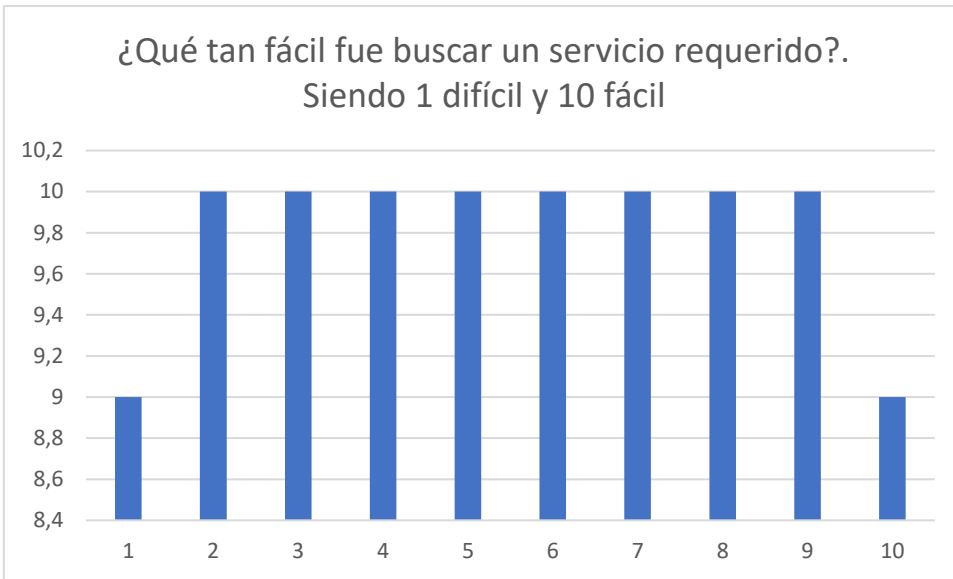
<https://docs.google.com/forms/d/e/1FAIpQLSfXAb81nJyBoevCTNsW2vAhcEqHhDKia6RBjrR7-Yw5S70piQ/viewform>

Link de encuesta de usabilidad

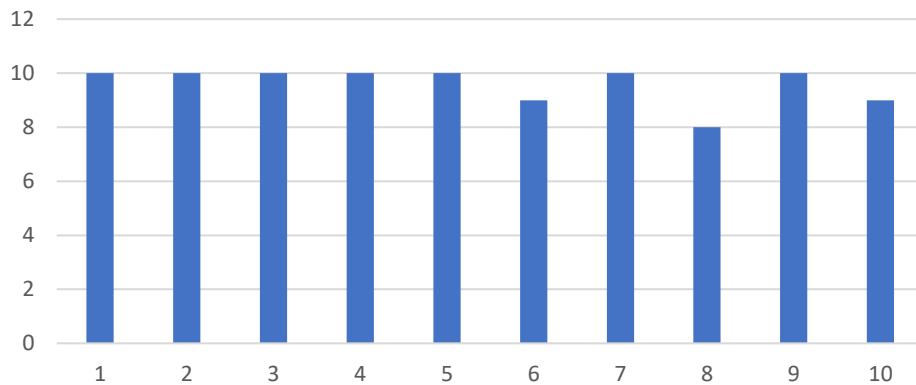
<https://docs.google.com/forms/d/e/1FAIpQLSdpjWVmUdlvpj812thg0QsLMwYQ9dMTSmOXmkueB-0Mx3J7LA/viewform>

Resultados de la encuesta aplicada a 10 empleados de la empresa

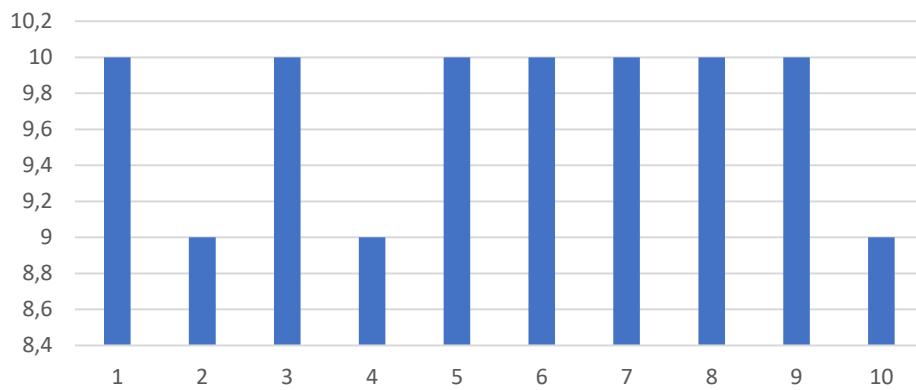




¿Qué tan fácil fue publicar la experiencia de un servicio contratado en la aplicación?. Siendo 1 difícil y 10 fácil



¿Utilizaría esta aplicación para contratar servicios requeridos para una siguiente ocasión?. Siendo 1 poco probable y 10 muy probable



En términos generales, ¿Cómo califica la experiencia de uso de la aplicación? Siendo 1 difícil de usar y 10 fácil de usar

