

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

DESARROLLO DE UNA PLATAFORMA TIPO UBER ORIENTADO A SERVICIOS DOMÉSTICOS EN ECUADOR

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

JAVIER SEBASTIAN TINOCO MACHADO

javier.tinoco@epn.edu.ec

RODNEY ARMANDO TAPIA CHÁVEZ

rodney.tapia@epn.edu.ec

DIRECTOR: ING. MARITZOL TENEMAZA MSc.

maritzol.tenemaza@epn.edu.ec

CODIRECTOR: ING. ADRIAN EGUEZ MSc.

adrian.eguez@epn.edu.ec

Quito, octubre 2019

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Javier Sebastián Tinoco Machado y Rodney Armando Tapia Chávez, bajo nuestra supervisión.

Ing. Maritzol Tenemaza MSc.
DIRECTOR DE PROYECTO

Ing. Adrián Equez MSc.
CODIRECTOR DE PROYECTO

DECLARACIÓN

Nosotros, Javier Sebastián Tinoco Machado y Rodney Armando Tapia Chávez declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros 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.

Javier Sebastián Tinoco Machado

Rodney Armando Tapia Chávez

DEDICATORIA

A mi familia, por el apoyo incondicional brindado.

Armando Tapia

A mis padres, por su apoyo brindado en esta etapa de mi vida.

Javier Tinoco

AGRADECIMIENTOS

A mi familia, por el apoyo incondicional.

A la Escuela Politécnica Nacional y todos los que formaron, forman parte de ella por el aprendizaje brindado.

A nuestros directores de tesis MSc. Maritzol Tenemaza y MSc. Adrián Eguez por el tiempo y esfuerzo dedicado a que este proyecto se concluya de forma exitosa.

A todos los compañeros y amigos que nos han brindado consejo y apoyo, a mi compañero tesis Javier Tinoco, por el tiempo, dedicación y esfuerzo dedicados a este proyecto.

Armando Tapia

A mi familia, por su apoyo y por haber creído siempre en mí.

A mis amigos que fueron parte de este proceso y por su amistad todos estos años que aún continúa.

A todos esos profesores que durante mis estudios demostraron preocuparse porque sus alumnos aprendan y que sean mejores personas.

A nuestros directores de tesis MSc. Maritzol Tenemaza y MSc. Adrián Eguez por su esfuerzo y tiempo dedicado en este proyecto.

A mi compañero de tesis, Armando Tapia.

Javier Tinoco

ÍNDICE DEL CONTENIDO

1. INTRODUCCIÓN.....	1
1.1 Planteamiento del problema	1
1.2 Objetivos	2
1.2.1 Objetivo General.....	2
1.2.2 Objetivos Específicos	2
1.3 Marco Teórico.....	2
1.3.1 Uber Economy (Modelo de negocio de Uber)	2
1.3.2 Marketplace.....	3
1.3.3 Servicios domésticos	3
1.3.4 Servicios domésticos ocasionales	3
1.3.5 Arquitectura Orientada a Servicios [9].....	3
1.3.6 Servicio [10].....	4
1.3.7 Servicios Web.....	4
1.3.8 Servicios Web RESTful [9].....	4
1.4 Herramientas y librerías de desarrollo	4
1.5 Enfoque Ágil	7
1.5.1 Scrum.....	8
1.5.1.1 El Equipo Scrum.....	8
1.5.1.2 Roles de Scrum	8
1.5.1.3 Artefactos de Scrum	8
1.5.1.4 Eventos Scrum	9
1.5.1.5 Release Planning	10
1.5.1.6 Sprint 0.....	11
2. METODOLOGÍA.....	12
2.1 Definición de roles	12
2.2 Arquitectura del sistema	12
2.3 Descripción del proyecto.....	13
2.3.1 Historias Épicas	14
2.3.2 Product Backlog.....	14
2.3.3 Planificación del proyecto.....	14
2.4 Sprint 0.....	15
2.4.1 Descripción básica de datos	15
2.4.2 Requerimientos Funcionales	16
2.4.3 Requerimientos no funcionales.....	17

2.5	Sprint 1	18
2.5.1	Sprint Planning	18
2.5.2	Ejecución del Sprint	20
2.5.3	Sprint Review	22
2.5.4	Sprint Retrospective	28
2.6	Sprint 2	29
2.6.1	Sprint Planning	29
2.6.2	Ejecución del Sprint	30
2.6.3	Sprint Review	34
2.6.4	Sprint Retrospective	38
2.7	Sprint 3	39
2.7.2	Ejecución del Sprint	40
2.7.3	Sprint Review	42
2.7.4	Sprint Retrospective	47
2.8	Sprint 4	47
2.8.1	Sprint Planning	47
2.8.2	Ejecución del Sprint	49
2.8.3	Sprint Review	51
2.8.4	Sprint Retrospective	55
3.	RESULTADOS Y DISCUSIÓN	57
3.2	Pruebas con usuarios finales	64
3.2.1	Utilidad percibida	64
3.2.2	Facilidad de uso percibida	66
4.	CONCLUSIONES Y RECOMENDACIONES	69
5.	Referencias Bibliográficas	71
6.	Anexos	73

Lista de Figuras

Figura 1. Arquitectura del sistema.....	13
Figura 2. Modelo de datos de rol	20
Figura 3. Modelo de datos de usuario.....	20
Figura 4. Petición HTTP en Postman con su resultado.....	21
Figura 5. Configuración de SendGrid.....	21
Figura 6. Mensaje de correo electrónico personalizado	21
Figura 7. Implementación de Angular Google Maps	22
Figura 8. GUI para pre-registro de ofertantes	24
Figura 9. GUI de ingreso a la plataforma.	24
Figura 10. Búsqueda de ofertantes	25
Figura 11. Formulario de registro de Ofertante	25
Figura 12. GUI de registro de domicilio.....	26
Figura 13. GUI de registro para solicitante.....	26
Figura 14. GUI ingreso de información extra de ubicación	27
Figura 15. Envío de correo electrónico	27
Figura 16. Burndown chart del Sprint 1	28
Figura 17. Modelo de datos de domicilio.....	31
Figura 18. Modelo de datos de un servicio.....	31
Figura 19. Geolocalización de ofertantes.....	32
Figura 20. Indexación de coordenadas	32
Figura 21. Coordenadas geográficas	32
Figura 22. Integración de Paypal.	33
Figura 23. Redirección a Paypal.	33
Figura 24. Integración de notificaciones push.	33
Figura 25. GUI de selección de habilidades	35
Figura 26. GUI de selección de horario y frecuencia.....	36
Figura 27. Resultado de la búsqueda de ofertantes	36
Figura 28. Integración con Paypal.....	37
Figura 29. Push notification desde Firebase.	37
Figura 30. Burndown Chart del Sprint 2.	38
Figura 31. Modelo de datos de un servicio.....	41
Figura 32. Lista de servicios según su estado para el solicitante.	41
Figura 33. Lista de servicios según su estado para el ofertante.	41
Figura 34. Despliegue del Frontend en Firebase.	42
Figura 35. GUI para confirmar o rechazar un servicio.	44
Figura 36. GUI para iniciar un servicio	45
Figura 37. GUI para finalizar un servicio	45
Figura 38. GUI para calificación de solicitante y ofertante.....	46
Figura 39. Burndown Chart del Sprint 3	47
Figura 40. Integración con Amazon S3.	49
Figura 41. Integración de chart.js.....	50
Figura 42. Gestión de contraseñas con Bcrypt.....	50
Figura 43. Implementación de JWT para generar Tokens.....	50
Figura 44. Atributo que permite suspender a un usuario	51
Figura 45. Edición de la información de ofertante por el administrador	53
Figura 46. GUI para cambio de contraseña	53
Figura 47. GUI para restaurar contraseña.....	53
Figura 48. GUI para suspender ofertantes o solicitantes.....	54

Figura 49. Edición de información de ofertantes y solicitantes.	54
Figura 50. GUI de gráficas estadísticas.	55
Figura 51. Burndown chart del Sprint 4.	56
Figura 52. GUI de bienvenida par administrador.	57
Figura 53. GUI de bienvenida par solicitante	58
Figura 54. GUI de bienvenida par ofertante.	58
Figura 55. Ingreso de ubicación del servicio.	59
Figura 56. Selección de habilidades requeridas.	60
Figura 57. Selección del horario para el servicio.	60
Figura 58. Selección de ofertante para el servicio.	61
Figura 59. Pago del servicio.	61
Figura 60. Servicio en estado pendiente.	62
Figura 61. Servicio en estado aceptado.	62
Figura 62. Servicio en estado finalizado.	63
Figura 63. Calificación del servicio.	63
Figura 64. Lista de servicios finalizados.	64
Figura 65. Resultados utilidad percibida de ofertante.	65
Figura 66. Resultados utilidad percibida de solicitante.	66
Figura 67. Resultados facilidad de uso de ofertante	67
Figura 68. Resultados facilidad de uso de solicitante.	67
Figura 69. Mock de pre-registro	83
Figura 70. Mock registro de ofertante.	84
Figura 71. Mock registro nuevo usuario.	84
Figura 72. Mock ingreso a la plataforma.	84
Figura 73. Mock ingreso de ubicación.	85
Figura 74. Mock selección de habilidades.	85
Figura 75. Mock ingreso de horario.	85
Figura 76. Mock de ofertantes disponibles.	86
Figura 77. Mock de reserva de servicio.	86
Figura 78. Mock de lista de servicios solicitante.	87
Figura 79. Mock edición de domicilio.	88
Figura 80. Mock de lista de servicios del ofertante.	88
Figura 81. Mock de calificación.	88
Figura 82. Mock cambio de contraseña.	89
Figura 83. Mock restablecimiento de contraseña.	89
Figura 84. Mock de suspensión de usuario.	89
Figura 85. Mock de información de ofertantes.	90
Figura 86. Mock de información estadística.	90
Figura 87. Preguntas al solicitante sobre utilidad percibida.	91
Figura 88. Preguntas al ofertante sobre utilidad percibida.	91
Figura 89. Preguntas al solicitante sobre facilidad de uso.	92
Figura 90. Preguntas al ofertante sobre facilidad de uso.	92
Figura 91. Respuestas de ofertante para pregunta 1 de utilidad percibida.	93
Figura 92. Respuestas de ofertante para pregunta 2 de utilidad percibida.	93
Figura 93. Respuestas de ofertante para pregunta 3 de utilidad percibida.	93
Figura 94. Respuestas de solicitante para pregunta 1 de utilidad percibida.	94
Figura 95. Respuestas de solicitante para pregunta 2 de utilidad percibida.	94
Figura 96. Respuestas de solicitante para pregunta 3 de utilidad percibida.	94
Figura 97. Respuestas de ofertante para pregunta 1 de facilidad de uso.	95
Figura 98. Respuestas de ofertante para pregunta 2 para la utilidad percibida.	95

Figura 99. Respuestas de ofertante para pregunta 3 para la utilidad percibida.	95
Figura 100. Respuestas de solicitante para pregunta 1 de facilidad de uso.	96
Figura 101. Respuestas de solicitante para pregunta 2 de facilidad de uso.	96
Figura 102. Respuestas de solicitante para pregunta 3 de facilidad de uso.	96

LISTA DE TABLAS

Tabla 1. Definición de Roles.	12
Tabla 2. Horas de trabajo del equipo de desarrollo.	13
Tabla 3. Historias épicas de usuarios.	14
Tabla 4. Product Backlog inicial.	14
Tabla 5. Release Planning.	15
Tabla 6. Requerimientos funcionales de ofertante.	16
Tabla 7. Requerimientos funcionales de solicitante.	17
Tabla 8. Requerimientos funcionales de administrador.	17
Tabla 9. Historias de usuario para el sprint 1.	18
Tabla 10. Sprint Backlog del sprint 1.	19
Tabla 11. Pruebas de aceptación del Sprint 1.	23
Tabla 12. Product Backlog adaptado al finalizar el sprint 1.	28
Tabla 13. Historias de usuario para el Sprint 2.	29
Tabla 14. Sprint Backlog del Sprint 2.	30
Tabla 15. Pruebas de aceptación del Sprint 2.	35
Tabla 16. Product Backlog adaptado al finalizar el sprint 2.	38
Tabla 17. Historias de usuario para el Sprint 3.	39
Tabla 18. Sprint Backlog del Sprint 3.	40
Tabla 19. Pruebas de aceptación del Sprint 3.	43
Tabla 20. Product Backlog adaptado al finalizar el Sprint 3.	46
Tabla 21. Historias de usuario para el Sprint 4.	48
Tabla 22. Sprint Backlog del Sprint 4.	49
Tabla 23. Pruebas de aceptación del Sprint 4.	52
Tabla 24. Product Backlog adaptado al finalizar el Sprint 4.	55
Tabla 25. Historia Épica HUE01.	73
Tabla 26. Historia Épica HUE02.	73
Tabla 27. Historia Épica HUE03.	73
Tabla 28. Historia Épica HUE04.	73
Tabla 29. Historia de usuario 1 (HU01).	74
Tabla 30. Historia de usuario 2 (HU02).	74
Tabla 31. Historia de usuario 3 (HU03).	75
Tabla 32. Historia de usuario 4 (HU04).	75
Tabla 33. Historia de usuario 5 (HU05).	76
Tabla 34. Historia de usuario 6 (HU06).	76
Tabla 35. Historia de usuario 7 (HU07).	77
Tabla 36. Historia de usuario 8 (HU08).	77
Tabla 37. Historia de usuario 9 (HU09).	78
Tabla 38. Historia de usuario 10 (HU10).	78
Tabla 39. Historia de usuario 11 (HU11).	79
Tabla 40. Historia de usuario 12 (HU12).	79
Tabla 41. Historia de usuario 13 (HU13).	80
Tabla 42. Historia de usuario 14 (HU14).	80
Tabla 43. Historia de usuario 15 (HU15).	81
Tabla 44. Historia de usuario 16 (HU16).	81
Tabla 45. Historia de usuario 17 (HU17).	82
Tabla 46. Historia de usuario 18 (HU18).	82
Tabla 47. Historia de usuario 18 (HU18).	83
Tabla 48. Historia de usuario 20 (HU20).	83

RESUMEN

En Ecuador, el sector de trabajadores informales se encuentra en aumento cada año. Dentro de este sector, existe un alto porcentaje de personas que se dedican a ofrecer servicios domésticos, como su principal o única fuente de ingresos. A pesar de que en el país se han aprobado leyes para proteger al servidor doméstico y ofrecerle un pago justo por su trabajo, todavía existen muchas personas que ganan muy poco en comparación a las horas trabajadas o que no encuentran lugares donde ofrecer sus servicios. Por otro lado, las personas que requieren contratar servicios domésticos tienen dificultades para encontrar personas de confianza que se ajusten a sus necesidades, lo cual hace que desistan en esa búsqueda.

Como solución a este problema, se propone desarrollar una plataforma digital donde los trabajadores domésticos puedan ofertar sus servicios, realizar trabajos que se encuentren cerca de su domicilio, y que les permita ver el número de horas trabajadas y ganancias obtenidas. De igual manera, la plataforma permitirá a los usuarios solicitantes buscar a estos trabajadores según sus necesidades y el domicilio en el que requieran el servicio de manera rápida y segura.

Gracias a esta plataforma que servirá como intermediario entre solicitante y ofertante, se podrá facilitar el proceso de contratación y que existan mayores ofertas para los trabajadores domésticos en todo el país, asegurando un pago justo por el trabajo realizado.

PALABRAS CLAVE: SERVICIOS DOMÉSTICOS/ MARKETPLACE/ PLATAFORMA DIGITAL/ SERVICIO REST

ABSTRACT

In Ecuador, the informal workers sector is raising every year. Inside this sector, there is a big percentage of people who offer domestic services as their principal or only source of income. Despite that the Ecuadorian government has approved laws to protect the domestic worker and offer him/her a fair payment for his/her job, there are still a lot of people that gain very little in comparison with the hours they have worked. Moreover, this people can't find places where they can offer their services. On the other hand, people who require to hire domestic services, have difficulties to find people whom they can trust and adjust to their needs, which in the end makes them to stop looking.

As a solution to this problem, this project proposes the development of a digital platform where domestic workers can offer their services, work in places near their home and let them be able to keep track of the hours they have worked and how much they have gain. In the same way, this platform will allow the users that require these services to search for domestic workers according to their needs and the place where they need the service, in a secure and quick way.

Thanks to this platform that serves as a link between requesters and bidders, the hiring process will be easier and fast, and there will be more job offers for domestic workers around the country, assuring a fair payment for the hours they have worked.

KEYWORDS: DOMESTIC SERVICES/ MARKETPLACE/ DIGITAL PLATFORM/ REST SERVICE.

1. INTRODUCCIÓN

1.1 Planteamiento del problema

El Instituto Nacional de Estadísticas y Censos (INEC) en su informe de Encuesta Nacional de Empleo, Desempleo y Subempleo 2019 (ENEMDU) indica que en el Ecuador del total de personas con empleo, un porcentaje cercano al 46.7% se encuentra ubicado en el sector informal [1].

Por otra parte, el Código de Trabajo ecuatoriano, estipula que los trabajadores domésticos tienen derecho a remuneración del sueldo básico, afiliación de forma obligatoria al Instituto de Seguridad Social (IESS) y cumplir con una jornada laboral de 8 horas. Sin embargo, las remuneraciones de algunos de estos trabajadores están por debajo del salario básico y no reflejan un pago justo por el tiempo invertido y actividades realizadas. Esto ha sido ratificado por la Unión Nacional de Trabajadoras del Hogar y la fundación CARE Ecuador, que es una organización internacional sin fines de lucro, cuya misión principal es la lucha contra la pobreza e injusticia social en el país [2].

En el ámbito nacional hay varias empresas dedicadas al servicio de limpieza, pero no ofrecen una interacción directa entre el solicitante y ofertante, además los servicios ofertados no siempre se ajustan a las necesidades de los clientes. Acorde al presente proceso investigativo, se pudo identificar una plataforma digital de servicios de limpieza en Guayaquil, llamada Casera [3]. Esta plataforma permite a usuarios ofertar servicios de limpieza a través de su página web y personalizar el servicio como lo requieran los ofertantes. Sin embargo, el área de funcionamiento de la plataforma está restringida solo a ciertos sectores de Guayaquil, además no permite tener un registro de usuarios. Con esto, se limita al ofertante de la posibilidad de acceder a los datos de los servicios que ha realizado, las horas trabajadas y su ganancia total. En el caso del solicitante, no hay manera de acceder a un historial de servicios, ver perfiles profesionales, guardar domicilios para próximos servicios, entre otros. Es por esto, que se propone desarrollar una plataforma digital para que usuarios puedan ofertar servicios domésticos en todo el país, además de permitir a ofertantes decidir cuándo y dónde trabajar, garantizando un pago justo por el número de horas trabajadas a fin de cumplir con las remuneraciones que exige la ley y de esta forma evitar la sobreexplotación que sufre este sector en el país.

El empleado doméstico registrado en esta plataforma puede observar las ganancias obtenidas por sus servicios, así como horarios flexibles y la oportunidad de realizar servicios cerca de su domicilio, para evitar el recorrido de grandes distancias para llegar a su lugar de trabajo.

De igual manera, el usuario puede solicitar un servicio según el horario, habilidades y lugar requerido. Además, observar y comparar el perfil de los distintos trabajadores disponibles, y ver detalles del servicio, costo y duración.

Con el uso de esta plataforma también se pretende, disminuir problemas sociales como el trabajo infantil, ya que en la plataforma solo se registran personas mayores de edad que se encuentren aptas y calificadas para realizar el trabajo.

1.2 Objetivos

1.2.1 Objetivo General

Desarrollar mediante un enfoque ágil una plataforma web/móvil tipo UBER para ofrecer servicios domésticos en el Ecuador, utilizando una arquitectura orientada a servicios y base de datos No SQL.

1.2.2 Objetivos Específicos

- Diseñar e implementar un API REST para el manejo de datos y funcionalidades de la plataforma.
- Diseñar e implementar un aplicativo móvil/web que consuma la información de la API REST.
- Desarrollar un módulo de empleados para la gestión de su perfil y aceptación de servicios.
- Desarrollar un módulo para el registro de usuarios y proceso de solicitud del servicio.
- Desarrollar un módulo de administrador para control de usuarios y empleados.
- Desarrollar un módulo que permita tanto al usuario como al trabajador verificar que el servicio prestado se cumplió de manera satisfactoria.
- Implementar un servicio de pago digital para la realización de transacciones dentro de la plataforma.

1.3 Marco Teórico

1.3.1 Uber Economy (Modelo de negocio de Uber)

Uber es una empresa estadounidense fundada en el año 2009 con el objetivo de crear una plataforma virtual, la cual permite a usuarios y prestadores de servicios encontrarse y concretar transacciones, para el caso específico de Uber estas transacciones están enfocadas en movilidad y transporte [4], lo que dio paso a una nueva forma de comercializar productos o servicios.

El término “Uber economy” o “crowdwork”, inspirado en el nuevo modelo de negocio propuesto por Uber, consiste en la creación de plataformas digitales como: páginas web, aplicaciones móviles, etc., donde los clientes encuentran un trabajador autónomo que realice el servicio que ellos requieren [5].

La plataforma virtual toma el rol de facilitador e intermediario de las transacciones realizadas entre sus usuarios, este nuevo modelo de negocio donde el comercio se realiza sobre medios digitales se conoce como marketplace.

1.3.2 Marketplace

Marketplace, es un término que hace referencia a plataformas como Uber, Airbnb donde utilizando como medio el internet se facilita la interacción entre: ofertantes y solicitantes de productos o servicios, estas plataformas cumplen un papel de intermediario digital [6] que sirve para garantizar que los productos ofertados son de calidad y el pago de los clientes se realiza de la forma establecida.

La plataforma digital desarrollada en el presente proyecto sirve como punto de encuentro entre ofertantes y solicitantes, en donde las transacciones se realizan con base a servicios domésticos ocasionales.

1.3.3 Servicios domésticos

Un servicio doméstico es una actividad de mantenimiento aplicado a una vivienda u oficina para que esta sea habitada, usada o alcanzar cierto nivel de confort. Comúnmente en caso de oficinas, estos servicios se realizan de forma periódica [7].

Por otra parte, los servicios que requiere una vivienda son muy variados, pero se los puede clasificar acorde a la finalidad y se pueden clasificar en cocina, lavado, planchado, limpieza entre otros.

1.3.4 Servicios domésticos ocasionales

Los servicios domésticos ocasionales buscan satisfacer la demanda de los usuarios en función de sus necesidades, en este tipo de trabajo no es necesario contar con un contrato formal, si no que estas actividades se realizan de forma ocasional, es decir, que se prestan por horas o días específicos y buscan satisfacer la demanda en horarios extendidos como noches, fines de semana, feriados, etc. [8].

1.3.5 Arquitectura Orientada a Servicios [9].

Arquitectura orientada a servicios (SOA, por sus siglas en inglés: Service Oriented Architectures) propone una forma de desarrollar sistemas distribuidos en los que los componentes del sistema son servicios independientes y se ejecutan en computadores distribuidos geográficamente.

Los sistemas de software pueden componer servicios locales y externos de diferentes proveedores, con interacción uniforme entre los servicios del sistema.

Una arquitectura SOA consta básicamente de proveedores de servicios y solicitantes de servicios (llamados en ocasiones clientes), dichos proveedores diseñan e implementan servicios y especifican las interfaces, mientras que los consumidores pueden unir su aplicación con un servicio específico y comunicarse con él mediante protocolos estándar [9].

1.3.6 Servicio [10]

Dentro del contexto de una Arquitectura Orientada a Servicios, se puede definir un servicio como una unidad de lógica que corre dentro de una red. Un servicio tiene las siguientes características:

- Maneja funciones que son parte del proceso del negocio.
- Maneja tareas técnicas (como el acceso a la base de datos).
- Provee data del negocio que es necesaria para construir una interfaz gráfica
- Puede acceder a otro servicio.
- Es independiente de otro software.

1.3.7 Servicios Web

Los servicios web son una colección de protocolos abiertos y estándares usados para intercambiar data entre aplicaciones o sistemas a través de la red [11]. En una arquitectura orientada a servicios, los servicios web se utilizan para facilitar la comunicación entre los proveedores y los consumidores del servicio [12].

1.3.8 Servicios Web RESTful [9]

REST (acrónimo de Representational State Transfer, es decir, transferencia de estado representacional) es un estilo de arquitectura de software basado en la transferencia de representaciones de recursos de un servidor a un cliente. Es un método que permite implementar servicios web.

Un servicio web RESTful se identifica mediante su URL (identificador universal de recurso) y se comunica mediante el protocolo HTTP. Responde a métodos HTTP como GET, POST, PUT y DELETE y regresa una representación del recurso al cliente.

1.4 Herramientas y librerías de desarrollo

- **Balsamiq**

Es una herramienta de maquetación que permite realizar interfaces de usuario, enfocándose principalmente en la estructura, contenido y navegación [13]. En el desarrollo se utilizó para levantar requerimientos, diseño de interfaces gráficas, bosquejo de una estructura básica de la ubicación de la información y la navegación entre páginas.

- **Sails**

Es un framework de Javascript para el desarrollo web del lado del servidor o back-end [16]. Basado en Node.js, en el proyecto se utilizó para el desarrollo e implementación del API REST.

- **GitHub**

Es una plataforma de almacenamiento o repositorio de código informático que utiliza Git para control de versiones y desarrollo colaborativo entre los miembros del equipo [18].

- **Heroku**

Es una plataforma como servicio basada en contenedores que permite implementar y ejecutar aplicaciones modernas enfocada en entrega de software [14]. Se utilizó para desplegar el backend de la plataforma en un servidor online con acceso público.

- **Angular**

Es un framework para crear aplicaciones web e híbridas en HTML y Typescript. Angular integra patrones de diseño, buenas prácticas, modelos, directivas, componentes, etc. para resolver problemas de desarrollo [20]. Fue utilizado para desarrollo del frontend de la plataforma.

- **Angular Google Maps**

Es un conjunto de componentes para Angular 2 que proporcionan soluciones de integración para el API oficial de google Maps Core V3 [15]. En el desarrollo se utilizó para gestionar las coordenadas de geolocalización y visualización de Google Maps.

- **Bootstrap**

Es una librería de componentes de código abierto, para construir proyectos con HTML, CSS y JS [16]. Se utilizó para realizar un diseño web adaptable (adaptar la apariencia de las páginas web al dispositivo que se esté utilizando) y estilizar las interfaces.

- **Firebase Cloud Messaging**

Es una solución de mensajería y notificaciones multiplataforma [22]. Se utilizó para implementar notificaciones push (mensajes enviados desde el servidor al cliente), con el fin de facilitar la interacción entre usuarios de la plataforma.

- **Postman**

Es un API de desarrollo que entre sus herramientas provee un cliente HTTP [23], que permite comprobar de manera fácil el funcionamiento de un servicio Web. Se utilizó para realizar peticiones al backend y comprobar su funcionalidad.

- **PayPal**

Es un servicio global que ofrece un sistema de pagos en línea que soporta transferencias de dinero entre usuarios [24]. Con la utilización de PayPal se garantiza seguridad en los cobros, pagos y reembolsos de los pagos.

- **SendGrid**

Servicio de envío de mensajes de correo electrónico [25]. Gracias a SendGrid se puede enviar correos electrónicos desde el servidor a los usuarios de la plataforma.

- **Joi**

Es una librería de JavaScript que provee un lenguaje de descripción y validación de esquemas para objetos Javascript [17]. En el proyecto se utilizó para validación de contraseñas fuertes, e ingreso de campos obligatorios en formularios.

- **MongoDB**

Es una base de datos NoSQL de documentos, que ofrece una gran flexibilidad y escalabilidad, adicionalmente provee un modelo de consultas e indexación avanzado [18]. En el proyecto se utilizó para el almacenamiento de datos e información relacionada al proyecto.

- **RoboMongo**

Es una herramienta de código abierto, multiplataforma que permite la administración de bases de datos. Además incluye la consola actual de MongoDB [19]. En el desarrollo se utilizó para la gestión, realizar consultas, modificaciones y visualización de la base de datos NoSQL.

- **Chart.js**

Es una librería de Javascript creada para desarrolladores que permite crear gráficos estadísticos simples y flexibles [20]. En el proyecto se utilizó para realizar graficas estadísticas relacionadas con el uso y el tráfico en la plataforma.

- **JWT**

JSON Web Tokens son métodos de estándar abierto que permiten comunicar de forma segura dos partes mediante solicitudes [21]. En el proyecto se utilizó para generar tokens de acceso con un tiempo establecido de duración.

- **Amazon EC2**

Amazon Elastic Compute Cloud es un servicio web que proporciona una capacidad informática segura y redimensionable en la nube. Permite alquilar máquinas virtuales para ejecutar distintas aplicaciones [22]. En el desarrollo se utilizó como servidor de la base de datos en MongoDB.

- **Amazon S3**

Es un servicio de almacenamiento de recursos informáticos multimedia online [23]. En el desarrollo se utilizó para almacenar archivos como imágenes de perfil y documentos pdf.

1.5 Enfoque Ágil

En la década de 1980 como alternativa al enfoque de desarrollo de software basado en planeación, apareció el enfoque ágil. Se basa en un conjunto de métodos que permiten centrar la atención en la creación y entrega incremental e iterativa del software en lugar de la documentación y el diseño. Tiene como objetivo la entrega de un producto funcional al final de cada iteración. En base a estas entregas, los clientes propondrán requerimientos que deberán ser analizados para determinar si serán o no incluidos en posteriores iteraciones [9].

Los modelos ágiles de creación de software han demostrado tener éxito en proyectos en los que se construyen productos de software pequeños, medianos o grandes y en los cuales el cliente está dispuesto a participar activamente en el proceso. Hoy en día el 71% de las organizaciones a nivel mundial utiliza metodologías ágiles para todo tipo de proyectos de software [24].

Los principios de los métodos ágiles [9]:

- **Participación del cliente:** los clientes o usuarios finales son parte activa en el proceso de desarrollo, son responsables de proveer, priorizar requerimientos, evaluar y aprobar el producto.
- **Entrega Incremental:** el software se entrega en incrementos, el cliente indica los requerimientos que se incluirán en cada incremento.
- **Personas, no procesos:** el equipo de desarrollo tiene la capacidad de auto organización, no existen procesos establecidos en la forma de trabajar del equipo.
- **Adoptar el cambio:** el software está diseñado para aceptar el cambio como parte del proceso.
- **Mantener la simplicidad:** mantener la simplicidad del producto de software como en el proceso de desarrollo.

1.5.1 Scrum

Es un “framework” o marco de referencia para administrar procesos, entre ellos el proceso de creación de software. Scrum trabaja de forma adaptativa, iterativa, rápida, flexible y efectiva, diseñado para entregar software con valor significativo en cada iteración [25].

1.5.1.1 El Equipo Scrum

El equipo Scrum es un conjunto de personas, conformado aproximadamente por una cantidad de 3 a 9 integrantes. Cada integrante tiene un rol definido (puede tener más de uno, aunque no es recomendable) que dictará sus funciones dentro del proyecto.

Los equipos Scrum son autoorganizados y multifuncionales. La autoorganización permite que sea el propio equipo el que administre y organice el tiempo y sus esfuerzos en el desarrollo, mientras que la multifuncionalidad permite contar con la experiencia y experticia para realizar el trabajo sin ayuda externa [26].

1.5.1.2 Roles de Scrum

- **Product Owner:** dueño del producto (su traducción al español). Es quien comprende y orienta la lógica del negocio, y además prioriza las funcionalidades con más valor al producto [26].
- **Development Team:** equipo de desarrollo (su traducción al español). Es el equipo de especialistas responsables del desarrollo del producto [26].
- **Scrum Master:** facilita la comunicación entre los diferentes roles del equipo, mientras ayuda a que Scrum sea entendido y adaptado [26].

1.5.1.3 Artefactos de Scrum

Los artefactos Scrum son elementos físicos que se producen como resultado de la aplicación de este marco de trabajo, los tres principales son: Product Backlog, Sprint Backlog y el incremento [27].

- **Lista del producto (Product Backlog).**

El Product Backlog es una lista ordenada de todo lo que se sabe que requiere el producto final, usualmente descritos como historias de usuario [28]. Es la única fuente de requerimientos para cualquier cambio que se realice al producto, donde cada punto está descrito en lenguaje no técnico con el fin de ser presentado a cualquier miembro del equipo [26].

- **Lista de pendientes del Sprint (Sprint Backlog).**

El sprint backlog es un conjunto de elementos seleccionados de lista del producto o Product Backlog que deberán ser completados por el equipo de desarrollo

durante el transcurso del Sprint [29], también contiene un plan para entregar el incremento de producto y alcanzar el objetivo del Sprint [30].

La lista de pendientes o Sprint Backlog es una imagen visible en tiempo real del trabajo que el equipo de desarrollo planea llevar a cabo durante el Sprint [30].

- **Incremento.**

Al final de cada Sprint el equipo de desarrollo es responsable de presentar un incremento de producto potencialmente entregable [31].

El incremento es el producto analizado y desarrollado durante el Sprint. Este incremento debe estar en condiciones de utilizarse [30].

- **Burn-down**

Es una representación gráfica del trabajo pendiente en función del tiempo. Es útil para predecir y medir la velocidad del equipo y comprobar si el proyecto finalizará según el cronograma establecido [32].

1.5.1.4 Eventos Scrum

Hay 5 eventos en un proyecto Scrum.

- **Sprint**

Es una iteración o ciclo dentro del proyecto Scrum, en la cual se crea un incremento del producto. Cada Sprint contiene los siguientes eventos: Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective [26].

Al final de cada sprint el producto debe tener una parte terminada, utilizable y potencialmente desplegable. Cada nuevo Sprint comienza inmediatamente al finalizar el sprint anterior [30].

- **Objetivo del sprint:** es lo que el equipo de desarrollo se compromete a terminar al finalizar el Sprint.

- **Sprint Planning**

Es el primer evento dentro de un Sprint. El equipo Scrum planifica las características que serán incluidas en el proyecto durante el sprint y la forma en que estas serán entregadas [26].

- **Daily Scrum**

Son reuniones diarias, normalmente de 15 minutos, para coordinar el trabajo de las próximas 24 horas [26].

Se analiza el trabajo realizado desde la última reunión, y se realiza una proyección del trabajo que podrá realizarse. Las reuniones diarias deben ser a la misma hora y en la misma ubicación [30].

Cada miembro del equipo de desarrollo responde las siguientes preguntas [30]:

- ¿Qué se hizo ayer que ayudo al equipo a lograr el objetivo del Sprint?
- ¿Qué se hará hoy para que el equipo de desarrollo logre el objetivo del Sprint?
- ¿Hay algún bloqueante que impida que el equipo o yo alcance el objetivo del Sprint?

- **Sprint Review**

En la etapa previa al final del Sprint, el equipo de desarrollo se reúne con el cliente para presentar y demostrar el avance que se realizó, en donde el cliente aprueba, propone cambios o rechaza las funcionalidades realizadas durante el Sprint [26].

Es una reunión informal, la presentación de los avances son para facilitar la retroalimentación de parte del usuario y fomentar su colaboración dentro del desarrollo [30].

El resultado de la revisión es una lista de producto (product backlog) revisada, determinando cuales de los requerimientos del Sprint se han finalizado y cuales están incompletos, además pueden aparecer nuevos requerimientos que deben ser analizados y de ser pertinente añadidos a la lista del producto para futuros Sprints [30].

- **Sprint Retrospective**

En la etapa final del Sprint, el equipo de desarrollo se reúne de forma interna. En esta reunión se encuentran presentes el Scrum Master y el equipo de desarrollo. El objetivo es definir los puntos positivos y negativos del Sprint, para en base a estos, plantear mejoras que serán implementadas en los Sprints futuros [26].

1.5.1.5 Release Planning

Es un conjunto de historias de usuario agrupadas por entrega, que sirve como planificación a medio plazo [28]. No es requerido por Scrum, y nunca debe ser tratado como un compromiso ni como herramienta de presión para el equipo [33]. Sirve para comunicar las expectativas sobre lo que se puede desarrollar y el tiempo estimado, ayuda a tener una idea de progreso [34].

1.5.1.6 Sprint 0

Scrum se basa en un enfoque incremental iterativo. No siempre se abordan cuestiones estratégicas o tácticas como la asignación de recursos entre otros factores que se debe considerar de manera previa a la realización de un proyecto.

Para solventar este problema se utiliza el Sprint 0, que oficialmente no forma parte de Scrum.

El Sprint 0 no pretende entregar ninguna historia de usuario, es una forma para que el nuevo equipo aprenda el ritmo diario de los enfoques de SCRUM antes de integrar al product Owner. El equipo revisa el Product Backlog, realiza un bosquejo de los componentes involucrados en la arquitectura, se incluye también el manejo de la calidad y un plan arquitectónico [35].

Lo más importante del Sprint 0 es que se establecen límites claros sobre el tipo de características que los desarrolladores incluirán en el proyecto, el product Owner conoce sobre la lógica del negocio y cuando propone nuevas características, el equipo de desarrollo podrá decir si eso se puede o no hacer con base a la arquitectura de la aplicación [35].

2. METODOLOGÍA

Para el desarrollo del proyecto se utilizó metodologías ágiles, específicamente SCRUM. En el presente capítulo se detalla la implementación de esta metodología.

2.1 Definición de roles

En la tabla 1 se observa la asignación de roles Scrum en el proyecto.

Nombre	Rol
Maritzol Tenemaza.	Scrum Máster.
Maritzol Tenemaza.	Product Owner.
Armando Tapia. Javier Tinoco.	Scrum Team.

Tabla 1. Definición de Roles.

2.2 Arquitectura del sistema

En la figura 1 se observa la arquitectura del sistema. Para el sistema se utilizó una arquitectura orientada a servicios (SOA), y específicamente servicios web RESTful. El backend se desarrolló mediante Node.js, usando el framework de Sails.js; el cual se comunica con las siguientes APIs externas que aumentan las funcionalidades de la aplicación:

- Sendgrid: para enviar mensajes de correo electrónico.
- Firebase: para enviar de notificaciones push.
- Google: para gestionar la información geográfica.
- Paypal: para gestión de pagos seguros.

La información es almacenada en una base de datos NoSQL utilizando MongoDB, la misma que está alojada en un servidor EC2 provisto por AWS.

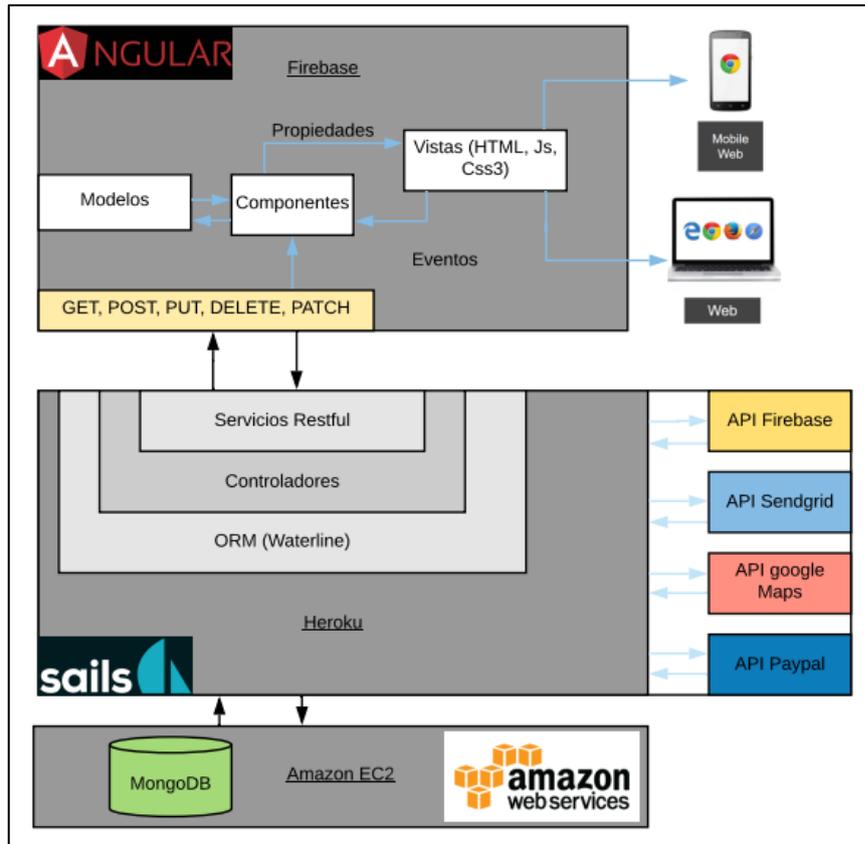


Figura 1. Arquitectura del sistema.

El cliente fue implementado en Angular que consume los servicios web expuestos en el backend mediante el uso de los verbos del protocolo HTTP (GET, POST, PUT, DELETE, PATCH) y genera las vistas para aplicación móvil y web. Finalmente, la aplicación fue desplegada usando el servicio de hosting de Firebase.

2.3 Descripción del proyecto

Los Sprint realizados durante el desarrollo del proyecto tendrán una duración de 2 semanas cada uno, los días de trabajo por semana son de 5 días, cada uno con 4 horas de trabajo.

Horas de trabajo

Desarrollador	Horas diarias
Javier Tinoco.	4
Armando Tapia.	4
Días a la semana.	5
Total, por semana.	$(4 + 4) * 5 = 40$ horas.
Total, por sprint.	$(40 * 3) = 120$ horas.
Total, por proyecto.	$(120 * 4) = 480$ horas.

Tabla 2. Horas de trabajo del equipo de desarrollo.

2.3.1 Historias Épicas

El título de las historias épicas identificadas para el desarrollo del proyecto se encuentra en la tabla 3, la información detallada de cada una de estas historias épicas se encuentra en el anexo I.

CÓDIGO	TÍTULO	PRORIDAD
HUE01	Gestión de la información del solicitante.	Alta
HUE02	Gestión de la información del empleado.	Alta
HUE03	Gestión administrativa de la plataforma.	Alta
HUE04	Proceso de solicitud de servicio.	Alta

Tabla 3. Historias épicas de usuarios.

2.3.2 Product Backlog

En la tabla 7 se encuentra detallado el Product Backlog creado a partir de las historias épicas (En los anexos: II, III, IV y V se encuentran detalladas cada una de las historias según su Sprint).

Product Backlog				
Épica	Código	Historia de Usuario	Prioridad	Estimación
HUE01	HU01	Pre-registro de ofertante.	Alta	13
HUE02	HU02	Registro de ofertantes en la plataforma.	Alta	21
HUE01	HU03	Registro de nuevo solicitante.	Alta	21
HUE01	HU04	Ingreso a la plataforma.	Alta	13
HUE02	HU05	Ingreso de ubicación.	Alta	13
HUE04	HU06	Selección de habilidades.	Alta	13
HUE04	HU07	Ingresar horario de servicio.	Alta	21
HUE04	HU08	Ofertantes disponibles.	Alta	21
HUE04	HU09	Reserva del servicio.	Alta	13
HUE01	HU10	Lista de servicios de solicitante.	Alta	13
HUE04	HU11	Inicio y finalización del servicio.	Alta	21
HUE01	HU12	Edición de domicilios u oficina.	Alta	21
HUE02	HU13	Lista de los servicios de ofertante.	Alta	21
HUE04	HU14	Calificación de ofertante a solicitante.	Alta	13
HUE04	HU15	Calificación de solicitante a ofertante.	Alta	13
HUE02	HU16	Cambio de contraseña.	Alta	13
HUE02	HU17	Restablecimiento de contraseña.	Alta	13
HUE03	HU18	Suspensión usuarios.	Alta	8
HUE03	HU19	Edición de información de empleados.	Alta	13
HUE03	HU20	Información estadística.	Alta	13

Tabla 4. Product Backlog inicial.

2.3.3 Planificación del proyecto

El desarrollo del proyecto se realizará en 4 Sprints sin incluir el Sprint 0, el tiempo de duración de cada Sprint será 2 semanas (14 días) cada uno.

La distribución de historias de usuario se realizó en función al esfuerzo que conlleva cada una de ellas.

En la tabla 8 se indica el Release Planning con la distribución de las historias de usuario en cada sprint.

Sprint 1	Sprint 2	Sprint 3	Sprint 4
HU01	HU06	HU11	HU16
HU02	HU07	HU12	HU17
HU03	HU08	HU13	HU18
HU04	HU09	HU14	HU19
HU05	HU10	HU15	HU20

Tabla 5. Release Planning.

2.4 Sprint 0

El Sprint 0 tuvo una duración de 14 días, donde el equipo de desarrollo estableció los siguientes elementos necesarios previos al inicio del proyecto.

- Con respecto al proyecto se identificó a los actores involucrados en el sistema, realizando un análisis básico de datos y requerimientos sobre cada uno de ellos, además de una arquitectura inicial de la plataforma.
- Con respecto al equipo se definió los roles Scrum y el tiempo asignado para cada sprint por parte del equipo de desarrollo.
- También se estableció las historias épicas a partir de las cuales fueron creadas las historias de usuario, el Product Backlog y por último se definió el número de Sprints con su respectiva duración y número de historias de usuario.

2.4.1 Descripción básica de datos

Los actores que interactuarán con la plataforma son:

- **Aspirante:** es el estado previo a ofertante (pre-registro) donde la persona interesada ingresa información básica para luego de un proceso manual convertirse en ofertante. La información del aspirante que se almacenará es:
 - Nombre.
 - Apellido.
 - Correo electrónico.
 - Número de teléfono.
 - Género.
 - Fecha de nacimiento.
- **Ofertante o empleado:** es la persona que ofrecerá sus servicios en la plataforma, para ser contratada por el solicitante, la información del ofertante que se almacenará es:
 - La información previamente registrada como aspirante.
 - Cédula.
 - Nombre de usuario.
 - Contacto extra.

- Habilidades que oferta: limpieza, cocina, lavado o planchado (al menos una).
 - Imagen de perfil.
 - Contraseña.
 - Plantilla de al menos un servicio básico (agua, electricidad, teléfono).
 - Información del domicilio: zona de ubicación, calle principal, calle secundaria, numeración, información extra, latitud y longitud.
- **Solicitante:** es la persona que busca ofertantes y contrata servicios a través de la plataforma. La información del solicitante que se almacena es:
 - Nombre y apellido.
 - Nombre de usuario.
 - Contraseña.
 - Teléfono.
 - Información del domicilio: Zona de ubicación, calle principal, calle secundaria, numeración, información extra, latitud y longitud.
- **Administrador:** es el encargado de gestión de usuarios (ofertantes y solicitantes), la información del administrador que se almacena es:
 - Nombre y apellidos.
 - Nombre de usuario.
 - Contraseña.

2.4.2 Requerimientos Funcionales

Se detallan los requerimientos funcionales del sistema, cada requerimiento se categoriza en:

- M: Requerimientos mandatorios.
- D: Requerimientos deseables.
- O: Requerimientos opcionales.

Ofertante		
ID	Requerimiento	Nivel
1	Pre-registro de un ofertante.	M
2	Edición de datos y habilidades.	M
3	Notificaciones de un nuevo servicio.	M
4	Aceptar o rechazar un servicio.	M
5	Notificación del estado de cada servicio.	M
6	Acceder a los detalles de los servicios.	M
7	Visualizar el perfil del solicitante.	M
8	Visualizar ganancias mensuales o semanales.	M
9	Calificar al solicitante una vez finalizado el servicio.	M
10	Realizar comentarios del servicio una vez finalizado.	M

Tabla 6. Requerimientos funcionales de ofertante.

Solicitante

ID	Requerimiento	Nivel
1	Registro de un solicitante.	M
2	Solicitar de un servicio.	M
3	Guardar información de domicilios (casa y oficina).	M
4	Guardar información de más de dos tipos de domicilios.	D
4	Visualizar y acceder a los detalles de un servicio.	M
5	Pago por el servicio contratado.	M
6	Editar la información básica.	M
7	Calificar al ofertante una vez finalizado el servicio.	M
8	Realizar comentarios del servicio una vez finalizado.	M

Tabla 7. Requerimientos funcionales de solicitante.

Administrador		
ID	Requerimiento	Nivel
1	Registro de ofertantes.	M
2	Editar información de ofertantes.	M
3	Suspender usuarios (solicitantes u ofertantes) de la plataforma.	M
4	Visualizar cuadros estadísticos de la plataforma.	D

Tabla 8. Requerimientos funcionales de administrador.

2.4.3 Requerimientos no funcionales

Se han identificado los siguientes tipos de requerimientos no funcionales para la plataforma.

- **Rapidez:**

Se refiere al tiempo de respuesta de la aplicación frente a las peticiones o eventos realizados por el usuario [9]. Para esta plataforma se espera un tiempo de respuesta no mayor a los 2 segundos por cada evento o petición, excepto por la búsqueda de ofertantes, la cual debe tener un tiempo de espera máximo de 15 segundos.

- **Facilidad de uso:**

Se refiere al tiempo de capacitación que requiere el usuario para usar la aplicación y la facilidad con la que interactúa con la misma [9]. Para esta plataforma se espera que tanto el usuario ofertante como solicitante puedan interactuar y utilizar la plataforma de forma intuitiva y sin mucha ayuda (documentación).

- **Fiabilidad**

Se refiere al tiempo medio entre falla y la probabilidad de que el sistema no se encuentre disponible [9]. Para lograr fiabilidad y alta disponibilidad, se utilizarán las plataformas como servicio (PaaS) de Heroku y Amazon Web Services (AWS), que ofrecen una arquitectura robusta con tiempo bajo de recuperación entre fallas y disponibilidad de al menos 99,99% al año.

- **Portabilidad**

Se refiere a la capacidad de un sistema de ser migrado a diferentes entornos, sin comprometer su funcionalidad [9].

El uso Amazon Web Services ofrece la creación de máquinas virtuales dedicadas, las que tienen la posibilidad de ser copiadas o clonadas facilitando la portabilidad de la plataforma.

2.5 Sprint 1

2.5.1 Sprint Planning

Objetivo del Sprint

- Implementar interfaz de usuario para el proceso de pre-registro, registro, ingreso a la plataforma y gestión de ubicación según se indica en el anexo VI.
- Desarrollar los métodos que permitan el almacenamiento, visualización, edición y gestión de la información generada para el proceso de pre-registro, registro, ingreso a la plataforma y gestión de ubicación.

Historias de usuario

En la tabla 9 se encuentran los títulos de las historias de usuario que fueron tomadas para el Sprint 1. (En Anexo II se encuentra información más detallada cada historia de usuario).

ID	Título	Estimación	Estado
HU01	Pre-registro de ofertante.	13	Desarrollo
HU02	Registro de ofertantes en la plataforma.	21	Desarrollo
HU03	Registro de nuevo solicitante.	21	Desarrollo
HU04	Ingreso a la plataforma.	13	Desarrollo
HU05	Ingreso de ubicación.	13	Desarrollo

Tabla 9. Historias de usuario para el sprint 1.

Sprint Backlog

En la tabla 10 se observa el Sprint Backlog donde se detalla las tareas involucradas en el desarrollo de cada una de las historias de usuario de este Sprint.

Sprint Backlog	
HISTORIA DE USUARIO	TAREAS
HU01	Implementación del formulario y campos del pre-registro.
	Implementar un método que permita validar el correo electrónico.
	Implementar un método que permita calcular la edad y verificar que el aspirante sea mayor de edad.
	Implementar un método en el backend que permita realizar las operaciones CRUD sobre los datos del aspirante.
HU02	Implementación del formulario y campos de registro.
	Implementar un método que permita buscar aspirantes por su nombre o apellido.
	Implementar un método que permita validar los campos del formulario.
	Implementación del mapa a través de la API de Google Maps.
	Envío de contraseña a la ofertante por correo electrónico.
	Implementación de un controlador en el backend que permita almacenar la información del empleado en la base de datos.
HU03	Implementación de bcrypt para almacenamiento seguro de contraseñas.
	Implementación del formulario y campos de registro.
	Validar los campos del formulario.
	Implementar un método que verifique que el nombre de usuario y correo electrónico ingresados sean únicos.
	Implementar un método que verifique la fortaleza de la contraseña.
HU04	Implementar un controlador en el backend para almacenar los datos ingresados por el usuario en la base de datos.
	Implementación del login.
	Implementar un método que valide que el campo de usuario y contraseña estén llenos.
	Implementar método que permita identificar qué rol del usuario de forma automática, para redireccionar a la pantalla correcta.
HU05	Implementar la interfaz principal de bienvenida al usuario con su nombre y apellido.
	Desarrollar la interfaz gráfica con el formulario y campos de registro del domicilio.
	Implementación del mapa a través de la API de Google Maps.
	Implementar un método para buscar lugares en el mapa y seleccionar a la posición exacta a través de un marcador.
HU05	Implementar un método para validar de los campos del formulario.
	Desarrollo de un controlador en el backend que indexe la información de longitud y latitud de la ubicación seleccionada, de acuerdo con el formato requerido por MongoDB.

Tabla 10. Sprint Backlog del sprint 1.

2.5.2 Ejecución del Sprint

A continuación, se presentan los detalles técnicos importantes que se realizaron durante el desarrollo del Sprint.

Modelos de datos para solicitantes, ofertantes

Para el desarrollo del backend se utilizó Sails.js (ver sección 1.4). Sails permite la definición de los modelos de datos requeridos, junto con sus atributos. Además, permite integrar estos modelos dentro de la base de datos para su almacenamiento.

En la figura 2 se observa la definición del modelo de datos correspondientes para la entidad rol, junto con su respectiva relación al modelo de datos de la entidad usuario.

Este modelo se observa en la figura 3.

```
module.exports = {
  attributes: {
    tipoRol: {
      type: "string",
      isIn: ["Administrador", "Usuario", "Empleado"],
      required: true
    },
    usuario: { collection: "Usuario", via: "rol" }
  }
};
```

Figura 2. Modelo de datos de rol.

```
module.exports = {
  attributes: {
    nombre: { type: "string" },
    username: { type: "string", unique: true, required: true },
    password: { type: "string" },
    cedula: { type: "string" },
    email: { type: "string", unique: true, required: true },
    telefono: { type: "string" },
    contacto_extra: { type: "string" },
    imagenPerfil: { type: "string" },
    sexo: { type: "string", isIn: ["Masculino", "Femenino"] },
    fecha_nacimiento: { type: "number" },
    rol: { collection: "Rol", via: "usuario" }
  }
};
```

Figura 3. Modelo de datos de usuario.

Despliegue del Backend

El despliegue y la configuración de los servicios del backend se realizaron en la plataforma Heroku (véase sección 1.4). Heroku provee una IP pública, a través de la cual se pueden realizar operaciones CRUD (Create, Read, Update and Delete, por sus siglas en inglés) a los servicios de forma online y permanente.

En la figura 4 se muestra el uso del cliente HTTP conocido como Postman (ver sección 1.4), para realizar una petición al backend (desplegado en línea) con el fin de visualizar los datos de todos los usuarios registrados en formato JSON.

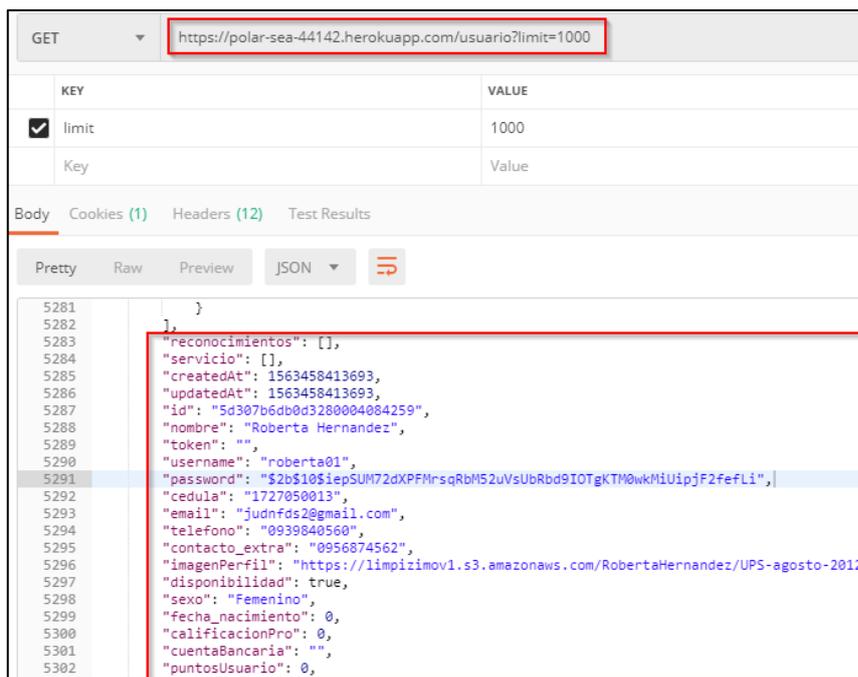


Figura 4. Petición HTTP en Postman con su resultado.

Integración con SendGrid

Una de las vías de envío de información desde la plataforma hacia los solicitantes u ofertantes es el correo electrónico. Para esto se utilizó SendGrid (ver sección 1.4) en la figura 5 se observa su integración con Node.js (ver sección 1.4).

```

const nodemailer = require("nodemailer");
const sendgridTransport = require("nodemailer-sendgrid-transport");
const transporter = nodemailer.createTransport(
  sendgridTransport({
    auth: {
      api_key: process.env.API_KEY_SENDGRID
    }
  })
);
  
```

Figura 5. Configuración de SendGrid.

SendGrid permite el envío de correos electrónicos personalizados. En la figura 6 se observa el segmento de código que permite personalizar un mensaje mediante HTML para ser enviado posteriormente.

```

// Envío de correo
transporter
  .sendMail({
    to: email,
    from: "servicios@epn.edu.ec.com",
    subject: "Bienvenid@ a Servicios",
    html: "<h1>Su password es: </h1>"
  })
  .catch(err => {
    console.log(err);
  });
  
```

Figura 6. Mensaje de correo electrónico personalizado.

Integración con Google Maps

Para la gestión de ubicación y localización mediante coordenadas es necesario utilizar la API de Google Maps, ya que permite integrar mapas en la plataforma y gestionar los datos de ubicación. En este caso se utilizó una librería especialmente diseñada para Angular llamada Angular Google Maps (ver sección 1.4), la misma que permite integrar Google Maps en la aplicación y ofrece métodos para el control de datos como latitud, longitud, calles, lugares de interés, etc.

La ubicación es almacenada como coordenadas. En la figura 7 se observa la implementación de Google Maps en Angular a través de un componente.

```
export class MapaComponent {
  marcadores: Marcador[] = [];
  // EPN -0.2104304, -78.4892959
  lat = -0.2104304;
  lng = -78.4892959;

  constructor(private snackBar: MatSnackBar, public dialog: MatDialog) {
    if (localStorage.getItem("marcadores")) {
      this.marcadores = JSON.parse(localStorage.getItem("marcadores"));
    }
  }

  agregarMarcador(e: any) {
    const coords: { lat: number; lng: number } = e.coords;
    let nuevoMarcador = new Marcador(coords.lat, coords.lng);
    this.marcadores.push(nuevoMarcador);
    this.guardarStorage();
    this.snackBar.open("Marcador Agregado", "Cerrar", { duration: 3000 });
  }
}
```

Figura 7. Implementación de Angular Google Maps.

2.5.3 Sprint Review

Pruebas de aceptación

En la tabla 11 se presentan los criterios de aceptación definidos por cada historia de usuario del Sprint 1 (ver anexo II) junto con su respectiva revisión de cumplimiento y satisfacción por parte del Product Owner.

Historia de Usuario	Criterio de Aceptación	Cumple
HU01	Comprobar validez del correo electrónico ingresado. Si es válido mostrar un mensaje de error al usuario.	Si
	Comprobar en la fecha de nacimiento, que el aspirante sea mayor de edad. Si es mayor de edad, permitir que se registre, caso contrario mostrar un mensaje que diga que solo se pueden registrar personas mayores de edad.	Si
	El botón de Aplicar debe estar habilitado solo si el formulario está completo y los campos ingresados son válidos. En caso de que el botón no esté habilitado, mostrar mensaje que diga que se deben llenar todos los campos y que deben ser válidos.	Si

Historia de Usuario	Criterio de Aceptación	Cumple
HU02	Búsqueda de usuarios pre-registrados por nombre.	Si
	El botón para registrar al ofertante se habilitará solo cuando el formulario está completo y todos los campos son válidos. Caso contrario se deberá mostrar un mensaje diciendo que todos los campos deben estar llenos y deben ser válidos.	Si
	Se debe enviar un mail al correo electrónico del ofertante, una vez finalizado el registro de forma automática. Este mail debe contener las credenciales de acceso a la plataforma.	Si
	La generación de la contraseña debe ser automática y solo debe ser vista por el ofertante en el correo enviado.	Si
HU03	Verificar que el correo electrónico ingresado sea válido.	Si
	Verificar que los campos de contraseña y confirmación de contraseña coincidan.	
	Validar que el correo ingresado no se encuentre registrado en la plataforma previamente.	Si
	Validar que el nombre de usuario sea único en la plataforma.	Si
	Verificar que la contraseña ingresada tenga: <ul style="list-style-type: none"> o Entre 8 y 16 caracteres. o Al menos un dígito. o Al menos una letra mayúscula. o Al menos una letra minúscula. 	Si
El botón "regístrate" debe estar habilitado solo cuando el formulario está completo y todos los campos son válidos.	Si	
HU04	El campo de la contraseña no debe mostrar el contenido.	Si
	El botón "Ingresar" debe habilitarse solo si el formulario está completo.	Si
	Si al momento de ingresar, las credenciales no son válidas, mostrar el siguiente mensaje: "El usuario o contraseña está equivocado. Vuelve a intentarlo."	Si
HU05	Si previamente el solicitante guardó uno de sus domicilios, se debe mostrar una lista con estos domicilios, y una opción para seleccionarlos y cargar su información en el formulario.	Si
	Integración de un mapa con Google Maps para seleccionar la ubicación del domicilio.	Si
	Debe existir una barra de búsqueda para encontrar con mayor facilidad la ubicación donde se realizará el servicio.	Si
	El botón "Continuar" debe estar habilitado cuando el formulario está completo y todos los campos son válidos.	Si

Tabla 11. Pruebas de aceptación del Sprint 1.

Incremento obtenido

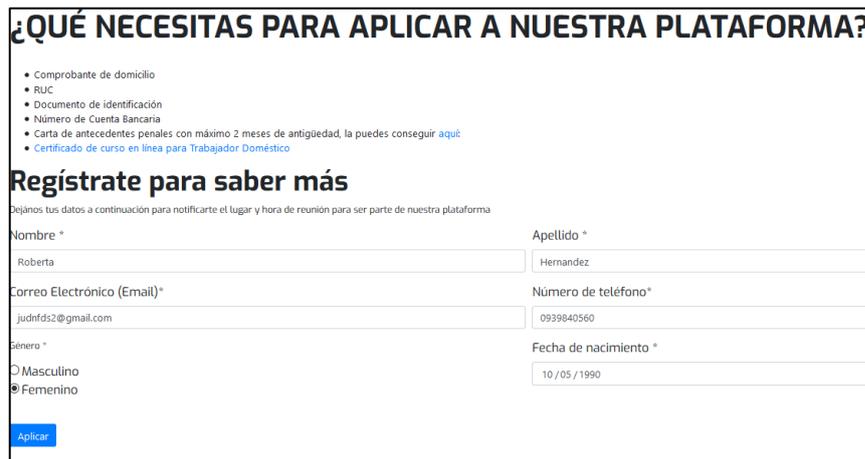
Los entregables para el primer incremento son:

Interfaces gráficas con sus respectivas validaciones.

Desarrollo de interfaces gráficas que permiten el registro de nuevos solicitantes, el pre-registro de ofertantes, y el acceso a la plataforma con los roles de ofertante, solicitante o administrador.

Pre-registro de ofertante

En la figura 8 se observa la implementación gráfica del proceso de pre-registro de ofertantes, según se especifica en la historia de usuario (ver anexo II) y el mock (ver anexo VI) respectivos.



¿QUÉ NECESITAS PARA APLICAR A NUESTRA PLATAFORMA?

- Comprobante de domicilio
- RUC
- Documento de identificación
- Número de Cuenta Bancaria
- Carta de antecedentes penales con máximo 2 meses de antigüedad, la puedes conseguir [aquí](#)
- [Certificado de curso en línea para Trabajador Doméstico](#)

Regístrate para saber más

Dejarnos tus datos a continuación para notificarte el lugar y hora de reunión para ser parte de nuestra plataforma

Nombre * Apellido *

Correo Electrónico (Email)* Número de teléfono*

Sexo *

Masculino

Femenino

Fecha de nacimiento *

Figura 8. GUI para pre-registro de ofertantes.

Interfaz de ingreso a la plataforma

En la figura 9 se muestra la pantalla de login a la plataforma. Esta pantalla es la misma para todos los usuarios, independientemente del rol que tienen (solicitante, ofertante o administrador).



Ingresa aquí

Usuario:

Contraseña:

Figura 9. GUI de ingreso a la plataforma.

Búsqueda de aspirantes y registro de ofertantes

En la figura 10 se observa la interfaz gráfica en donde se muestra la lista de aspirantes a la plataforma, junto con una barra de búsqueda. Además, se muestra los detalles del aspirante seleccionado por el usuario como nombre, fecha de nacimiento, teléfono y correo electrónico.

Buscar Empleado:

Roberta Hernandez

Aceptar

Información del aspirante

Nombre: Roberta Hernandez

Teléfono: 0939840560

Correo Electrónico: judnfs2@gmail.com

Fecha de Nacimiento: 1990-10-05

Continuar con el registro

Figura 10. Búsqueda de ofertantes.

En la figura 11 se observa el formulario de registro para que un aspirante se convierta en ofertante dentro de la plataforma. Aquí se puede subir información como la imagen de perfil, las habilidades que desee ofertar la persona (limpieza, cocina, lavado y planchado) y un documento en PDF del recibo de algún servicio básico que confirme la localización del domicilio de la persona.

Nombre de usuario *
roberta01

elecciona las habilidades que vas a prestar:

Limpieza
 Cocinar
 Lavar
 Planchar

Sube una foto de perfil aquí:

Seleccionar archivo | Subir

Vista previa:

Sube el recibo de agua, luz o teléfono del domicilio:

Seleccionar archivo | Subir

Vista previa:

Figura 11. Formulario de registro de Ofertante.

Integración de Google Maps

La geolocalización de los ofertantes como de los solicitantes es un requerimiento transversal en la plataforma, durante el desarrollo del sprint 1 se realiza las configuraciones necesarias para la implementación, gestión y almacenamiento de la geolocalización de sus domicilios.

Es importante resaltar que esta información es entregada en coordenadas longitud y latitud las mismas que deben estar indexadas para optimizar la búsqueda sobre ellas.

En la figura 12 se visualiza el formulario de registro del domicilio del ofertante junto con su ubicación en el mapa.

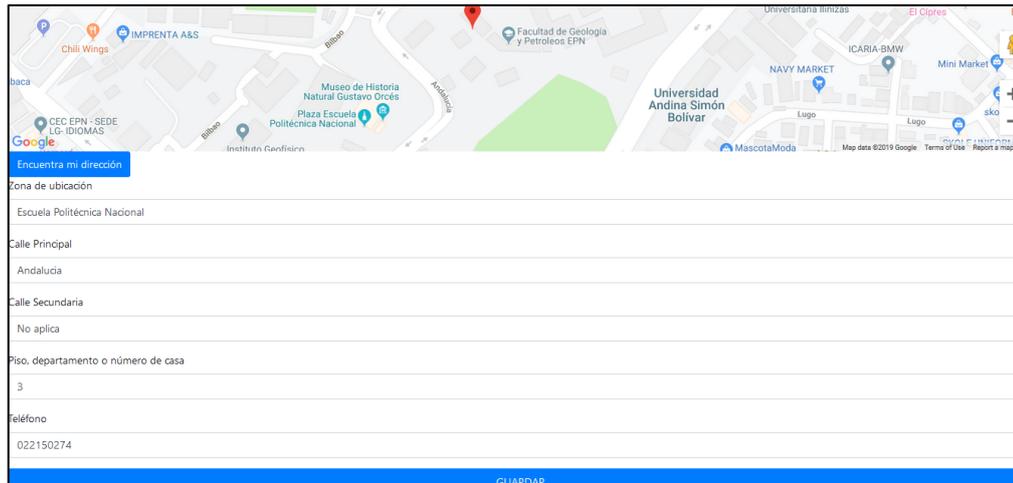


Figura 12. GUI de registro de domicilio.

Registro de usuario e ingreso de ubicación

Para que un solicitante de servicios (usuario) acceda a la plataforma debe registrarse. En la figura 13 se observa la interfaz gráfica que permite el registro de un nuevo solicitante.

Nombre	<input type="text"/>
	Ingrese su nombre
Apellido	<input type="text"/>
	Ingrese su apellido
Correo Electrónico	<input type="text"/>
	Ingrese su mail
Teléfono	<input type="text"/>
	Ingrese su número de teléfono
Nombre de usuario	<input type="text"/>
	Ingrese un nombre de usuario
Contraseña	<input type="password"/>
	Contraseña
Confirme su contraseña	<input type="password"/>
	Ingrese su contraseña de nuevo
Recuerda llenar todos los campos para continuar	
REGISTRATE	

Figura 13. GUI de registro para solicitante.

Para solicitar un servicio, es necesario ingresar la ubicación junto con la información referencial del domicilio donde se requiere este servicio. El usuario podrá guardar los datos del domicilio si desea, para realizar futuros servicios y no tener que llenar todo el formulario otra vez. En la figura 14 se puede visualizar la interfaz gráfica que permite ingresar la información de domicilio.

Figura 14. GUI ingreso de información extra de ubicación.

Envío de correos electrónicos y gestión de contraseñas

Una vez que el aspirante pasa a ser ofertante (empleado), la contraseña en texto plano es enviada por correo electrónico, como se observa en la figura 15. A su vez, la contraseña se almacena en la base de datos, pero a través de un hash para garantizar que solo el usuario conozca su contraseña.

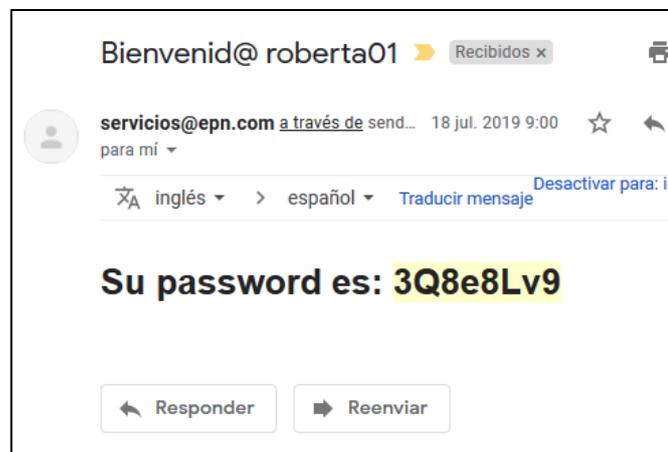


Figura 15. Envío de correo electrónico.

Adaptación del Product Backlog

En la tabla 12 se visualiza la actualización del Product Backlog al finalizar el Sprint 1. Todas las historias del Sprint han sido finalizadas.

Product Backlog				
Código	Historia de Usuario	Prioridad	Estimación	Estado
HU01	Pre-registro de ofertante.	Alta	13	Finalizado
HU02	Registro de ofertantes en la plataforma.	Alta	21	Finalizado
HU03	Registro de nuevo solicitante.	Alta	21	Finalizado
HU04	Ingreso a la plataforma.	Alta	13	Finalizado
HU05	Ingreso de ubicación.	Alta	13	Finalizado

Código	Historia de Usuario	Prioridad	Estimación	Estado
HU06	Selección de habilidades.	Alta	13	Pendiente
HU07	Ingresar horario de servicio.	Alta	21	Pendiente
HU08	Ofertantes disponibles.	Alta	21	Pendiente
HU09	Reserva del servicio.	Alta	13	Pendiente
HU10	Lista de servicios de solicitante.	Alta	13	Pendiente
HU11	Inicio y finalización del servicio.	Alta	21	Pendiente
HU12	Edición de domicilio u oficina.	Alta	21	Pendiente
HU13	Lista de los servicios de ofertante.	Alta	21	Pendiente
HU14	Calificación de ofertante a solicitante.	Alta	13	Pendiente
HU15	Calificación de solicitante a ofertante.	Alta	13	Pendiente
HU16	Cambio de contraseña.	Alta	13	Pendiente
HU17	Restablecimiento de contraseña.	Alta	13	Pendiente
HU18	Suspensión usuarios.	Alta	8	Pendiente
HU19	Edición de información de empleados.	Alta	13	Pendiente
HU20	Información estadística.	Media	13	Pendiente

Tabla 12. Product Backlog adaptado al finalizar el sprint 1.

2.5.4 Sprint Retrospective

En la figura 16 se visualiza el Burndown chart correspondiente a este Sprint. Se observa que en general el desarrollo de las historias estuvo por debajo del tiempo esperado. Existieron complicaciones al intentar integrar la API de Google Maps en Angular, debido a que la documentación de Google ofrecía ejemplos en Javascript y Angular trabaja con Typescript. Esto se pudo solucionar utilizando la librería Angular Google Maps (AGM), la cual ofrecía todos los métodos y funcionalidades necesarios para integrarse directamente en Angular. Gracias a esta librería se pudo completar las historias oportunamente, a pesar de haber estado por debajo del tiempo esperado durante el desarrollo.

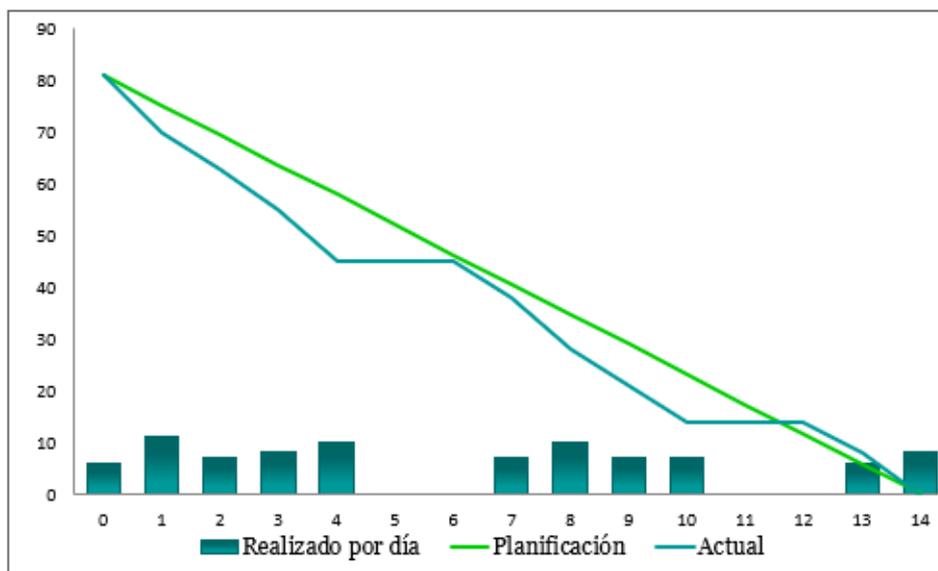


Figura 16. Burndown chart del Sprint 1.

2.6 Sprint 2

2.6.1 Sprint Planning

Objetivo del Sprint

- Implementar interfaces de usuario para el proceso de solicitud del servicio, según se indica en el anexo VII.
- Desarrollar los métodos que permitan el almacenamiento, visualización, edición y gestión de la información generada para el proceso de solicitud del servicio.

Historias de Usuario

En la tabla 13 se encuentran los títulos de las historias de usuario que fueron tomadas para el Sprint 2. (En Anexo III se encuentra información más detallada cada historia de usuario).

ID	Título	Estimación	Estado
HU06	Selección de habilidades.	13	Desarrollo
HU07	Ingresar horario de servicio.	21	Desarrollo
HU08	Ofertantes disponibles.	21	Desarrollo
HU09	Reserva del servicio.	13	Desarrollo
HU10	Lista de servicios de solicitante.	13	Desarrollo

Tabla 13. Historias de usuario para el Sprint 2.

Sprint Backlog

En la tabla 14 se encuentra el Sprint Backlog donde se detalla las tareas de cada una de las historias de usuario para este Sprint.

Sprint Backlog	
HISTORIA DE USUARIO	TAREAS
HU06	Implementación de la interfaz gráfica. Se deben tener las siguientes habilidades: limpieza, cocina, planchado y lavado, junto con una imagen de estas.
	Implementar un método para verificar que se ha seleccionado al menos una habilidad.
	Implementación del formulario de detalle del domicilio, en caso de que el usuario requiera la habilidad de limpieza.
	Implementación de un método para calcular el tiempo y el costo del servicio.
	Implementación de un controlador en el backend para registrar la información de las habilidades seleccionadas y almacenarlas en la base de datos.
HU07	Implementación de la interfaz que conste de un calendario con la semana actual y las horas por día en la que se puede contratar un servicio.

HISTORIA DE USUARIO	TAREAS
HU07	Implementar un método para obtener y desplegar la semana actual en la pantalla.
	Mostrar en pantalla un resumen de los días, horarios y frecuencia escogida por el usuario para su servicio.
	Implementar las opciones de frecuencia que puede tener un servicio (una sola vez o hasta por 4 semanas).
	Implementar un método que bloquee las horas del día en las que no se pueda dar el servicio.
HU08	Implementación de una interfaz con la lista de empleados disponibles, junto con su imagen de perfil y resumen de sus servicios.
	Implementación de un método que permita encontrar a los empleados más cercanos en función de su geolocalización.
	Implementación de función que permita seleccionar a los ofertantes que estén activos en la plataforma.
	Implementación una función que permita seleccionar los ofertantes que ofrezcan las habilidades requeridas para el servicio.
	Implementar función para enviar notificaciones push a los empleados seleccionados para realizar un servicio.
HU09	Implementar interfaz gráfica con un detalle del servicio solicitado.
	Agregar el método de pago mediante PayPal, a través de un botón en la pantalla.
	Implementar controlador en el backend para almacenar la información del servicio solicitado, junto con los datos de ofertante y solicitante.
	Enviar notificación al empleado, una vez que se haya efectuado el pago del servicio.
HU10	Implementar interfaz gráfica, con una lista de todos los servicios que ha contratado el solicitante.
	Realizar función para filtrar y mostrar los servicios según el estado actual del mismo.
	Implementar un método para ordenar los servicios por fecha de inicio.
	Mostrar detalles del servicio seleccionado en la pantalla.
	Implementar un botón que redirija al perfil del empleado que está a cargo de mi servicio.

Tabla 14. Sprint Backlog del Sprint 2.

2.6.2 Ejecución del Sprint

Se presenta los detalles técnicos importantes que se realizaron durante el desarrollo de este sprint.

Modelo de datos de Domicilio.

Se requiere que la ubicación de ofertantes y solicitantes se almacene en una estructura de datos cuyo cuerpo tenga el atributo "location", como se observa en la figura 17. Este es un formato requerido por MongoDB para realizar operaciones de geolocalización con los datos.

```

module.exports = {
  attributes: {
    tipoDomicilio: { type: "string", isIn: ["casa", "oficina"] },
    numeroMts2: { type: "number" },
    numeroHabitaciones: { type: "number" },
    numeroBanios: { type: "number" },
    sector: { type: "string" },
    callePrincipal: { type: "string" },
    calleSecundaria: { type: "string" },
    numeroCasa: { type: "string" },
    infoExtra: { type: "string" },
    estado: { type: "boolean" },
    location: { type: "json" },
    usuario: { model: "Usuario" }
  }
};

```

Figura 17. Modelo de datos de domicilio.

Modelo de datos del Servicio

El modelo de datos para un servicio contiene información relacionada con el solicitante, el ofertante, y los detalles de este, como se visualiza en la figura 18.

```

{
  "habilidades": [
    {
      "id": "5d31db5e6b41c10004683d06",
      "tipoHabilidad": "Limpieza",
      "imgSrc": "/assets/iconos/icono-limpieza.png"
    }
  ],
  "usuario": [
    {
      "id": "5d31dd496b41c10004683d16",
      "nombre": "Roberta Hernandez"
    },
    {
      "id": "5d31e2816b41c10004683dc1",
      "nombre": "usuario prueba"
    }
  ],
  "id": "5d32088aa23e9c00043384c9",
  "fechaInicio": [ "Sábado 20 de Julio", "Domingo 21 de Julio" ],
  "fechaFin": "Domingo 28 de Julio del 2019",
  "horaInicio": "13H17",
  "horaFin": "18H21",
  "horario": [ "13:00", "13:00" ],
  "duracion": "5",
  "aux_id_domicilio": "5d31e2eb6b41c10004683dc3",
  "calificacionE": 5,
  "calificacionU": 0,
  "comentarioE": "",
  "comentarioU": "",
  "costo": 24.16,
  "frecuencia": 2
}

```

Figura 18. Modelo de datos de un servicio.

Búsqueda de ofertantes

La información del servicio es procesada por la plataforma y como resultado se tiene una lista de ofertantes que se encuentran ubicados en un radio máximo de 4 km tomando como centro la ubicación del servicio. En la figura 19 se observa la implementación la función propia de MongoDB (ver sección 1.4) que permite realizar la geolocalización de ofertantes.

```

await db
  .collection("domicilio")
  .aggregate([
    {
      $geoNear: {
        near: { type: "Point", coordinates: [longitud, latitud] },
        distanceField: "dist.calculated",
        maxDistance: 4000,
        spherical: true
      }
    },
    {
      $lookup: {
        from: "usuario",
        localField: "usuario",
        foreignField: "_id",
        as: "usuario"
      }
    }
  ])
  .toArray()
  .then(data => {
    usuariosCerca.push(data);
  });

```

Figura 19. Geolocalización de ofertantes.

Indexación de la ubicación

Para realizar la búsqueda de geolocalización, MongoDB utiliza un índice conocido como "2dsphere", el cual lee las coordenadas almacenadas en la base de datos y calcula la distancia que existe entre otras coordenadas. Esta distancia se puede especificar según se requiera. En la figura 20 se muestra la consulta que se requiere para indexar las coordenadas de la ubicación en MongoDB, y la figura 21 muestra el formato que deben cumplir.

```

var db = sails.getDatastore().manager;
await db.collection('domicilio').createIndex({location:"2dsphere"})
return res.ok({results});

```

Figura 20. Indexación de coordenadas.

```

"location": {
  "type": "Point",
  "coordinates": [
    -78.4819092,
    -0.1762966
  ]
}

```

Figura 21. Coordenadas geográficas.

Integración con Paypal

Para la implementación de transacciones de dinero seguro se utilizó Paypal (ver sección 1.4). En la figura 22 se observa la configuración para integrar Paypal con Node.js y en la figura 23 se observa la petición hacia la API de Paypal para realizar el pago.

```

paypal1.configure({
  'mode': 'sandbox', //sandbox or live
  'client_id': process.env.CLIENT_ID_PAYPAL ,
  'client_secret': process.env.CLIENT_SECRET_PAYPAL
});

```

Figura 22. Integración de Paypal.

```

executePayment: async function(req, res) {
  let total = req.param("total");
  var PAYPAL_API = "https://api.sandbox.paypal.com";
  // 2. Get the payment ID and the payer ID from the request body.
  var paymentID = req.body.paymentID;
  var payerID = req.body.payerID;
  // 3. Call /v1/payments/payment/PAY-XXX/execute to finalize the payment.
  request.post(
    PAYPAL_API + "/v1/payments/payment/" + paymentID + "/execute",
    {
      auth: {
        user: process.env.CLIENT_ID_PAYPAL,
        pass: process.env.CLIENT_SECRET_PAYPAL
      },
      body: {
        payer_id: payerID,
        transactions: [
          {
            amount: {
              total: Number(total),
              currency: "USD"
            }
          }
        ]
      }
    },
    json: true
  ),
};

```

Figura 23. Redirección a Paypal.

Integración de Firebase

La interacción entre solicitante y ofertante se realiza enviando notificaciones push, a través de la implementación del servicio de Firebase Cloud Messaging (ver sección 1.4). En la figura 24 se observa el segmento de código para integrar Node.js con Firebase y poder enviar notificaciones a los clientes.

```

var fcm = require('fcm-notification');
var FCM = new fcm('config/env/limpizimo-4755e-firebase-adminsdk-itco5-8a0e83e7b1.json');

module.exports = {
  enviarpush(tokenDestino, msm, data){
    var message = {
      token: tokenDestino,
      notification:{
        title: 'Uber Servicios',
        body: msm,
      },
      data: data
    };
  };
};

```

Figura 24. Integración de notificaciones push.

2.6.3 Sprint Review

Pruebas de aceptación

En la tabla 15 se presentan los criterios de aceptación definidos por cada historia de usuario del Sprint 2 (ver anexo III) junto con su respectiva revisión de cumplimiento y satisfacción por parte del Product Owner.

Historia de Usuario	Criterio de Aceptación	Cumple
HU06	Marcar y desmarcar las habilidades requeridas.	Si
	El formulario de información extra del domicilio solo aparece cuando se selecciona la habilidad de limpieza.	Si
	Si previamente se seleccionó un domicilio ya guardado, debe mostrar la información adicional en el formulario, en caso de tenerla.	Si
	Dentro del formulario de información adicional del domicilio, restringir los campos numéricos a enteros positivos.	Si
	El botón "continuar" se debe habilitar solo si hay al menos una habilidad seleccionada.	Si
	Mostrar en pantalla el tiempo estimado para el servicio.	Si
HU07	Permitir que el usuario solo pueda escoger una hora por día.	Si
	Se deben bloquear las horas del día actual que ya pasaron.	Si
	Si la duración estimada del servicio excede la hora máxima de trabajo para un ofertante (7 de la noche), desde la hora inicial, entonces se debe bloquear dichas horas.	Si
	Se debe tener la posibilidad de escoger el inicio de un servicio hasta 3 semanas después de la actual.	Si
	El botón de reservar debe estar habilitado solo si se ha seleccionado al menos un día y un horario, y una frecuencia para el servicio. En caso contrario el usuario no puede continuar con el proceso de contratación del servicio.	Si
HU08	Lista de ofertantes que se encuentren máximo a 4 km de la ubicación donde se solicitó el servicio.	Si
	La lista de ofertantes contiene solo ofertantes activos.	Si
	La lista de ofertantes debe contener solo ofertantes que cumplan con las habilidades requeridas para el servicio.	Si
	Mostrar fotografía, calificación y 3 de los últimos comentarios que ha recibido el ofertante.	Si
	Al dar clic en el botón de "Reservar", continuar con el proceso de contratación del servicio con el empleado elegido.	Si
HU09	Realizar el pago a través de la plataforma Paypal.	Si
	Realizar pago del servicio.	Si
	El servicio se confirma solo si se realiza el pago exitosamente. Caso contrario, debe existir un botón que permita cancelar el servicio.	Si
	Si se guarda el servicio, se debe enviar una notificación al empleado seleccionado, con el fin de que pueda aceptar o rechazar el mismo.	Si

Historia de Usuario	Criterio de Aceptación	Cumple
HU10	Filtro de servicios según su estado.	Si
	Servicios ordenados de forma cronológica.	Si
	Recibir una notificación cuando el ofertante reservado acepte el servicio.	Si
	El botón “ver detalles” este activo solo cuando hay un servicio seleccionado.	Si
	Visualizar los detalles del servicio mediante el botón “Ver detalles”.	Si
	El botón “ver empleado” permite visualizar el perfil del empleado.	Si

Tabla 15. Pruebas de aceptación del Sprint 2.

Incremento obtenido

Los entregables para el segundo incremento son:

Personalización del servicio:

Selección de habilidades

En la figura 25 se observa la interfaz gráfica que permite al solicitante escoger las habilidades que requiere para su servicio. Además, se agregó una interfaz gráfica dinámica, en donde el formulario del tamaño del domicilio se muestra solo en caso de que el solicitante requiera la habilidad de limpieza.

Figura 25. GUI de selección de habilidades.

Selección de horario

En la figura 26 se observa la interfaz gráfica para seleccionar el día y la hora en la que el solicitante requiere el servicio. El calendario está separado por semanas, dando al solicitante la opción de pedir un servicio hasta 3 semanas después de la fecha actual.



Figura 26. GUI de selección de horario y frecuencia.

Búsqueda de ofertantes

Con la información ingresada para el servicio, se procede a mostrar una lista de ofertantes que se encuentren en un radio de 4 kilómetros, como se visualiza en la figura 27, tomando como punto de referencia la ubicación desde donde se solicitó el servicio.

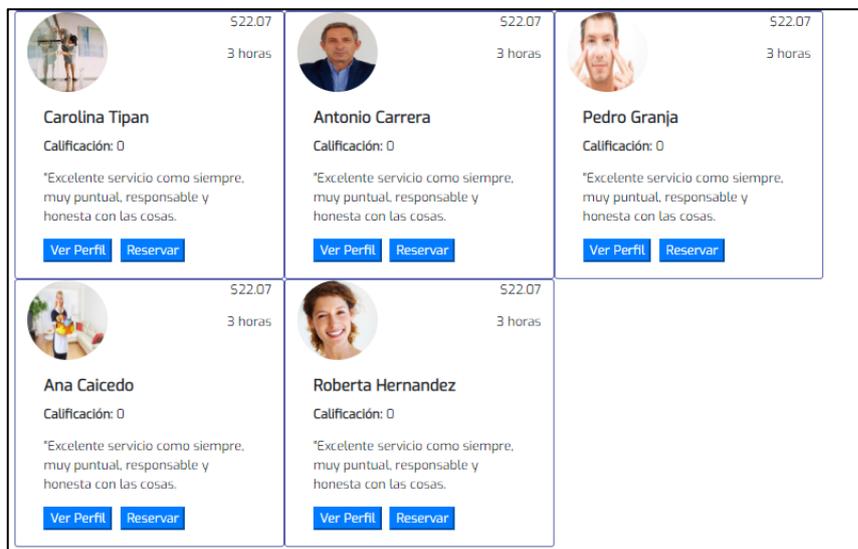


Figura 27. Resultado de la búsqueda de ofertantes.

Implementación de pagos

Paypal (ver sección 1.4) permite agregar la funcionalidad de pagos mediante un botón que redirecciona a su sitio oficial donde el comprador ingresa a su cuenta y realiza la transferencia. De esta forma, se delega la seguridad de transacciones al servicio de Paypal. En la figura 28 se observa la integración del botón de pagos con este sistema.

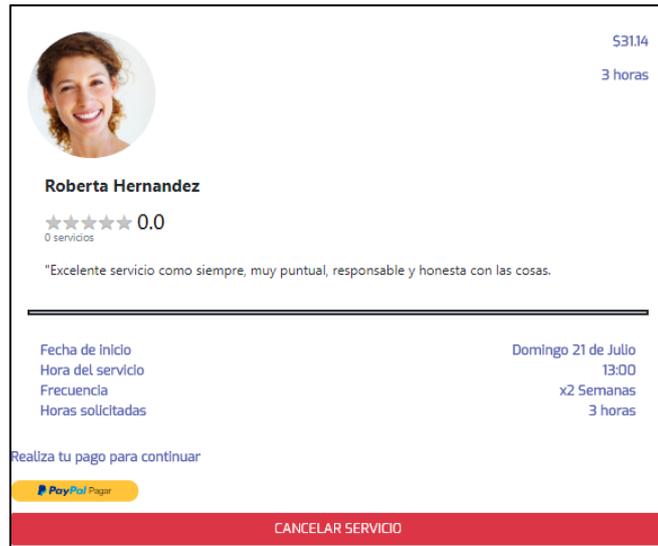


Figura 28. Integración con Paypal.

Envío de notificaciones

Para el envío de notificaciones desde el servidor hacia el cliente se usa los servicios de Firebase (ver sección 1.4) que permite enviar notificaciones push hacia el backend en Sails.js y posteriormente hacia el cliente.

Tanto el cliente como el ofertante recibirán estas notificaciones que le indicarán el estado de su servicio.

En la figura 29 se observa el ejemplo de una notificación push.

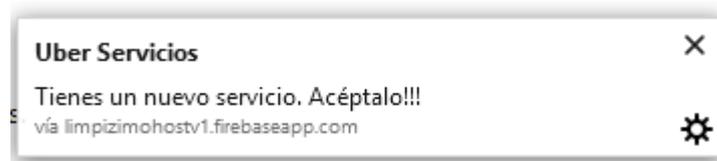


Figura 29. Push notification desde Firebase.

Adaptación del Product Backlog

En la tabla 16 se visualiza la actualización del Product Backlog al finalizar el Sprint 2, en donde las historias realizadas en este Sprint finalizaron de forma exitosa.

Product Backlog				
Código	Historia de Usuario	Prioridad	Estimación	Estado
HU01	Pre-registro de ofertante.	Alta	13	Finalizado
HU02	Registro de ofertantes en la plataforma.	Alta	21	Finalizado
HU03	Registro de nuevo usuario.	Alta	21	Finalizado
HU04	Ingreso a la plataforma.	Alta	13	Finalizado
HU05	Ingreso de ubicación.	Alta	13	Finalizado
HU06	Selección de habilidades.	Alta	13	Finalizado
HU07	Ingresar horario de servicio.	Alta	21	Finalizado
HU08	Ofertantes disponibles.	Alta	21	Finalizado
HU09	Reserva del servicio.	Alta	13	Finalizado
HU10	Lista de servicios de solicitante.	Alta	13	Finalizado

Código	Historia de Usuario	Prioridad	Estimación	Estado
HU11	Inicio y finalización del servicio.	Alta	21	Finalizado
HU12	Edición de domicilio u oficina.	Alta	21	Pendiente
HU13	Lista de los servicios de ofertante.	Alta	21	Pendiente
HU14	Calificación de ofertante a solicitante.	Alta	13	Pendiente
HU15	Calificación de solicitante a ofertante.	Alta	13	Pendiente
HU16	Cambio de contraseña.	Alta	13	Pendiente
HU17	Restablecimiento de contraseña.	Alta	13	Pendiente
HU18	Suspensión usuarios.	Alta	8	Pendiente
HU19	Edición de información de empleados.	Alta	13	Pendiente
HU20	Información estadística.	Media	13	Pendiente

Tabla 16. Product Backlog adaptado al finalizar el sprint 2.

2.6.4 Sprint Retrospective

En la figura 30 se observa el Burndown Chart para el Sprint 2. En este caso se tuvo complicaciones con la implementación de horarios y fechas dentro de la aplicación, lo cual retrasó el tiempo esperado en los primeros días. A pesar de esto, se lograron los objetivos esperados al final del Sprint.

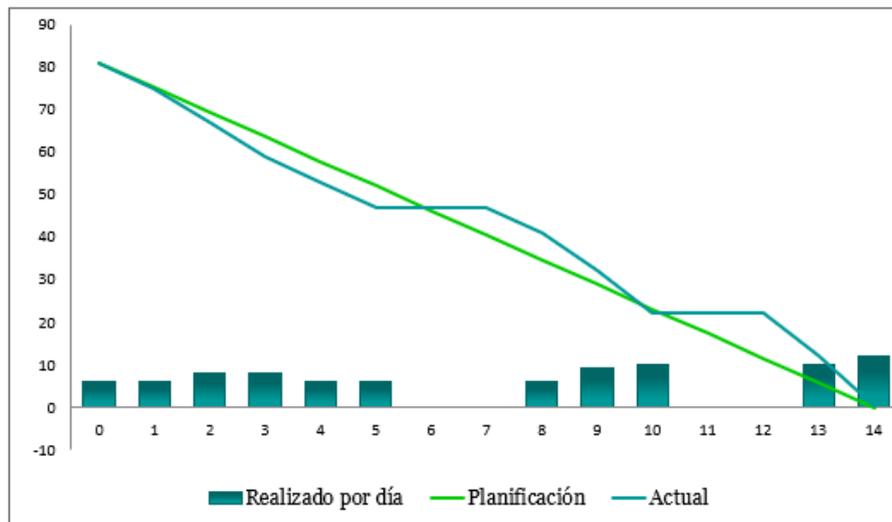


Figura 30. Burdown Chart del Sprint 2.

2.7 Sprint 3

2.7.1 Sprint Planning

Objetivo del Sprint

- Implementar el proceso para aceptar, iniciar y finalizar un servicio según se indica en el anexo VIII.
- Desarrollar los métodos que permitan el almacenamiento, visualización, edición y gestión de la información generada para el proceso de aceptar, iniciar y finalizar un servicio.

Historias de Usuario

En la tabla 17 se encuentran las historias de usuarios realizadas durante el desarrollo del sprint 3. (En Anexo IV se encuentra información más detallada cada historia de usuario).

ID	Título	Estimación	Estado
HU11	Inicio y finalización del servicio.	21	Desarrollo
HU12	Edición de domicilio u oficina.	21	Desarrollo
HU13	Lista de los servicios de ofertante.	21	Desarrollo
HU14	Calificación de ofertante a solicitante.	13	Desarrollo
HU15	Calificación de solicitante a ofertante.	13	Desarrollo

Tabla 17. Historias de usuario para el Sprint 3.

Sprint Backlog

En la tabla 18 se encuentra el Sprint Backlog donde se detalla las tareas involucradas en el desarrollo de cada una de las historias de usuario del Sprint 3.

Sprint Backlog	
HISTORIA DE USUARIO	TAREAS
HU11	Implementar la interfaz gráfica de acuerdo con el diseño.
	Implementar un método para filtrar servicios por estado (aceptado, iniciado o finalizado).
	Implementar un método que permita actualizar el estado de un servicio.
	Implementar envío de notificaciones cuando un servicio cambie de estado.
HU12	Implementación del formulario y campos para edición de domicilios.
	Implementar un método para obtener los datos del domicilio u oficina que fueron ingresados previamente por el usuario.
	Implementación de Google Maps para mostrar la información actual del domicilio y poder reemplazar con la nueva ubicación del usuario.
	Implementar una barra de búsqueda en el mapa para facilitar la localización de la ubicación.

HISTORIA DE USUARIO	TAREAS
HU13	Implementar la interfaz gráfica de acuerdo con el diseño.
	Implementar un método que liste todos los servicios de un ofertante y los filtre por su estado.
	Realizar un botón que permita redireccionar a la información del perfil del solicitante que requirió el servicio.
	Obtener en pantalla el detalle del servicio seleccionado, como fecha de inicio, fecha de finalización, usuario que solicita, costo y ubicación.
	Implementación de un mapa con la API de Google Maps que indique la ubicación exacta del domicilio donde se realiza el servicio.
HU14	Implementar interfaz gráfica para realizar la calificación de acuerdo con el diseño.
	Implementar un método para obtener el promedio de calificación del usuario.
	Implementación del formulario de calificación del solicitante al ofertante, con posibilidad de añadir puntuación, reconocimientos y comentarios.
HU15	Implementar interfaz gráfica para realizar la calificación de acuerdo con el diseño.
	Implementar un método que permita obtener los tres últimos comentarios de servicio realizados hacia el usuario para mostrarlos en pantalla.
	Implementación del formulario de calificación del ofertante al solicitante, con posibilidad de añadir puntuación y comentarios.

Tabla 18. Sprint Backlog del Sprint 3.

2.7.2 Ejecución del Sprint

Se presenta los detalles técnicos importantes que se realizaron durante el desarrollo de este Sprint.

Interacción entre ofertante y solicitante

El vínculo que existe entre un solicitante y un ofertante es el servicio, para un servicio se definió los siguientes estados:

- **Pendiente:** en espera a que la ofertante acepte el servicio.
- **Aceptado:** una vez que la ofertante ya acepto el servicio.
- **Iniciado:** cuando la ofertante arriba al lugar y oficialmente inicia el servicio.
- **Finalizado:** cuando finaliza el servicio.

Al finalizar un servicio tanto el solicitante como el ofertante pueden emitir de forma opcional una calificación y un comentario el uno al otro.

En la figura 31 se presenta el modelo del servicio, con los atributos requeridos para almacenar la información de la fecha, horario, duración, calificación, etc.

```

attributes: {
  habilidades:{collection:"Habilidad",via:'servicio'},
  usuario:{collection:'Usuario', via : 'servicio'},
  estado:{type:'string',isIn: ['Pendiente', 'Aceptado', 'Iniciado', 'Finalizado','Cancelado']},
  fechaInicio:{type:'ref'},
  fechaFin:{type:'string'},
  horaInicio:{type:'string'},
  horaFin:{type:'string'},
  horario:{type:'ref'},
  duracion:{type:'string'},
  aux_id_domicilio:{type:'string'},
  calificacionE:{type:'number',defaultsTo:-1},
  calificacionU:{type:'number',defaultsTo:-1},
  comentarioE:{type:'string'},
  comentarioU:{type:'string'},
  costo:{type:'number'},
  frecuencia:{type:'number',isIn: [1, 2, 3, 4], required: true}
},

```

Figura 31. Modelo de datos de un servicio.

Filtro de Servicios

Los servicios deben ser filtrados para cada usuario (solicitante u ofertante) filtrando el estado del servicio y el rol que cumple cada usuario.

- **Solicitante**

En la figura 32 se observa los estados por los que atraviesa un servicio visto desde el rol de solicitante.



Figura 32. Lista de servicios según su estado para el solicitante.

- **Ofertante**

En la figura 33 se visualiza los estados por los que atraviesa un servicio visto desde el rol de ofertante. Se tiene más estados debido a que el ofertante es quien acepta un servicio que se encuentra pendiente.



Figura 33. Lista de servicios según su estado para el ofertante.

Despliegue del Frontend

En la figura 34 se visualiza el despliegue del Frontend que se realizó utilizando el servicio de hosting que ofrece Firebase (ver sección 1.4). De esta manera, la plataforma ya puede ser utilizada para realizar pruebas con usuarios reales.

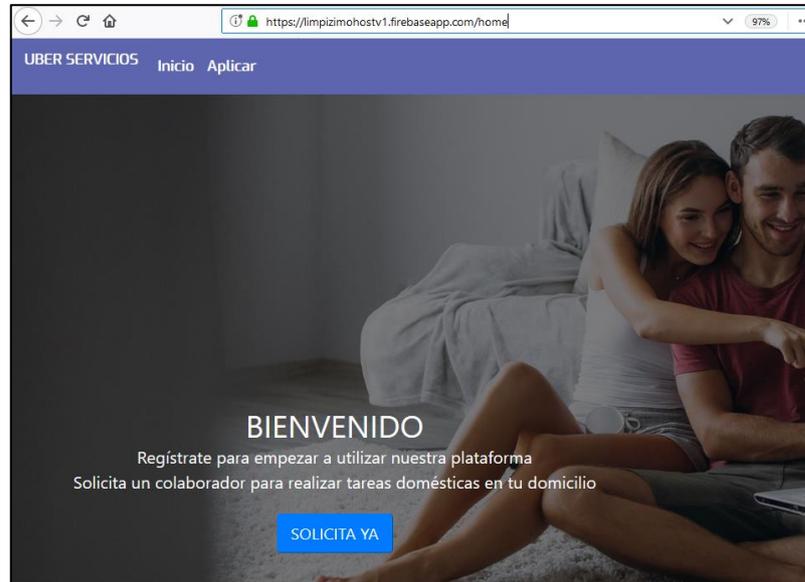


Figura 34. Despliegue del Frontend en Firebase.

2.7.3 Sprint Review

Pruebas de aceptación

En la tabla 19 se presentan los criterios de aceptación definidos por cada historia de usuario del sprint 3 (ver anexo IV) junto con su respectiva revisión de cumplimiento y satisfacción por parte del Product Owner.

Historia de Usuario	Criterio de Aceptación	Cumple
HU11	Si la lista de servicios está filtrada por "Pendientes", debe haber dos botones. Uno para aceptar el servicio, y otro para cancelar el servicio.	Si
	Si la lista de servicios está filtrada por "Aceptados", debe haber un botón para iniciar el servicio.	Si
	El botón para iniciar el servicio actualiza el estado a "iniciado" y guarda en la base de datos la hora exacta en la que comienza el servicio en el campo "hora de inicio".	Si
	Si la lista de servicios está filtrada por "Iniciados", debe haber un botón para finalizar el servicio.	Si
	El botón para finalizar el servicio actualiza el estado a "finalizado" y guarda en la base de datos la hora exacta en la que termina el servicio en el campo "hora de fin".	Si
	Si la lista de servicios está filtrada por "Finalizados", no debe haber ningún botón ya que es el último estado posible.	Si

Historia de Usuario	Criterio de Aceptación	Cumple
HU11	En cada cambio de estado se debe enviar una notificación al solicitante, informando el nuevo estado del servicio.	Si
HU12	Se debe mostrar la lista de domicilios, solo en caso de que se haya guardado uno previamente. Los tipos de domicilio que se pueden guardar son casa u oficina.	Si
	El mapa debe mostrar la ubicación actual del domicilio y permitir seleccionar una nueva ubicación. El mapa debe tener una barra de búsqueda de ubicaciones.	Si
	El botón "guardar" se habilita solo cuando el formulario está completo y todos los campos son válidos, caso contrario debe mostrar un mensaje que informe al usuario que debe llenar todos los campos y que deben ser válidos.	Si
HU13	Filtro de servicios según su estado.	Si
	Animación de carga al aplicar un filtro a la lista de servicios.	Si
	Servicios ordenados de forma cronológica.	Si
	El botón "ver detalles" está activo solo cuando hay un servicio seleccionado.	Si
	Los detalles del servicio deben aparecer en la misma página.	Si
	Al dar clic en el enlace del perfil del solicitante, se debe visualizar el perfil en una nueva página.	Si
HU14	Visualizar el nombre del solicitante (usuario).	Si
	Visualizar la foto de perfil.	Si
	En la calificación, se debe mostrar 5 estrellas de color blanco y cuando el empleado pulse sobre una deben marcarse de amarillo.	Si
	Los comentarios pueden ser opcionales, por lo que se debe poder enviar la calificación, aunque el comentario este vacío.	Si
	El botón enviar debe estar habilitado solo si se ha seleccionado un número de estrellas, caso contrario debe existir la opción de cerrar para omitir el paso de calificación.	Si
HU15	Visualizar el nombre y foto de perfil del ofertante.	Si
	En la calificación, se debe mostrar 5 estrellas de color blanco y cuando el solicitante pulse sobre ellas, estas deben marcarse de amarillo.	Si
	Deben aparecer popups por cada servicio que tiene el solicitante sin calificar, cada vez que ingrese a la plataforma.	Si
	Los comentarios y reconocimientos pueden ser opcionales, por lo que se debe poder enviar la calificación, aunque esos campos estén vacíos.	Si
	El botón enviar debe estar habilitado solo si se ha seleccionado un número de estrellas, caso contrario debe existir la opción de cerrar para omitir el paso de calificación.	Si
	Si el solicitante no desea calificar el servicio y da clic en cerrar, ya no debe volver a aparecerle el popup.	Si

Tabla 19. Pruebas de aceptación del Sprint 3.

Incremento obtenido

Los entregables para el tercer incremento son:

Confirmar o Rechazar un servicio

En la figura 35 se observa la interfaz gráfica que permite confirmar o rechazar el servicio. El proceso inicia cuando el solicitante hace una reserva, en ese momento el ofertante recibe una notificación. Después puede visualizar los detalles, la ubicación y el perfil del solicitante; y según esto decidir si aceptar o rechazar el trabajo.

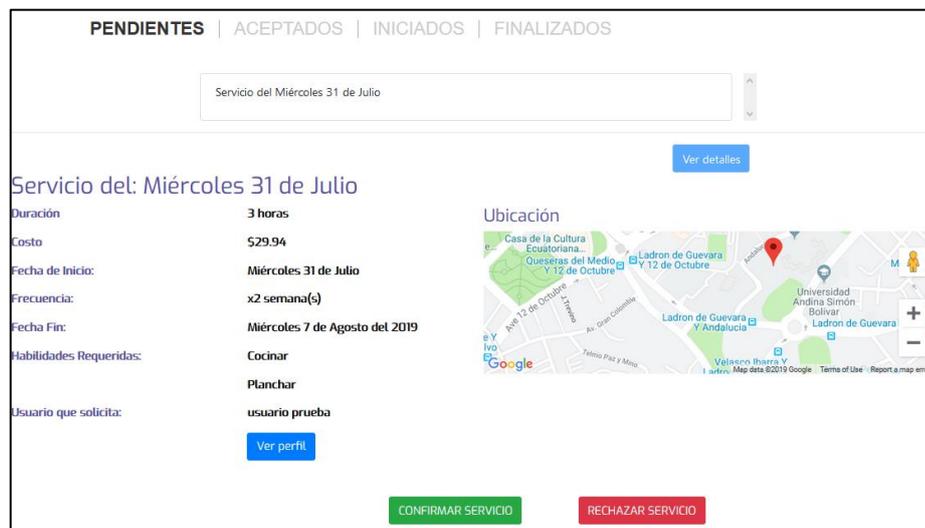


Figura 35. GUI para confirmar o rechazar un servicio.

Servicios Rechazados

En caso de que el ofertante decline el servicio para el que fue reservado, se le indicará mediante una notificación al solicitante la cancelación. En este caso se podrá verificar en la base de datos para realizar un reembolso por el servicio.

Servicios Aceptados

Un servicio ya aceptado, será iniciado una vez que el ofertante arribe a la ubicación donde se solicitó el servicio y comience con el trabajo. Una vez ahí, debe pulsar el botón "Iniciar Servicio", como se observa en la figura 36, para tener constancia del momento exacto en que empieza, tanto para el ofertante como el solicitante.

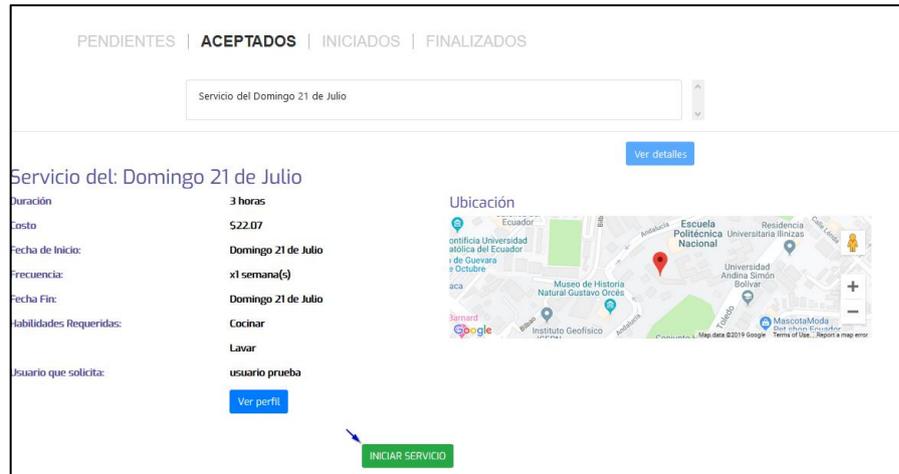


Figura 36. GUI para iniciar un servicio.

Finalización del servicio

En la figura 37 se observa la interfaz gráfica para finalizar un servicio en donde al pulsar el botón “Finalizar servicio”, se registra la fecha y hora exacta de finalización del trabajo en la base de datos.

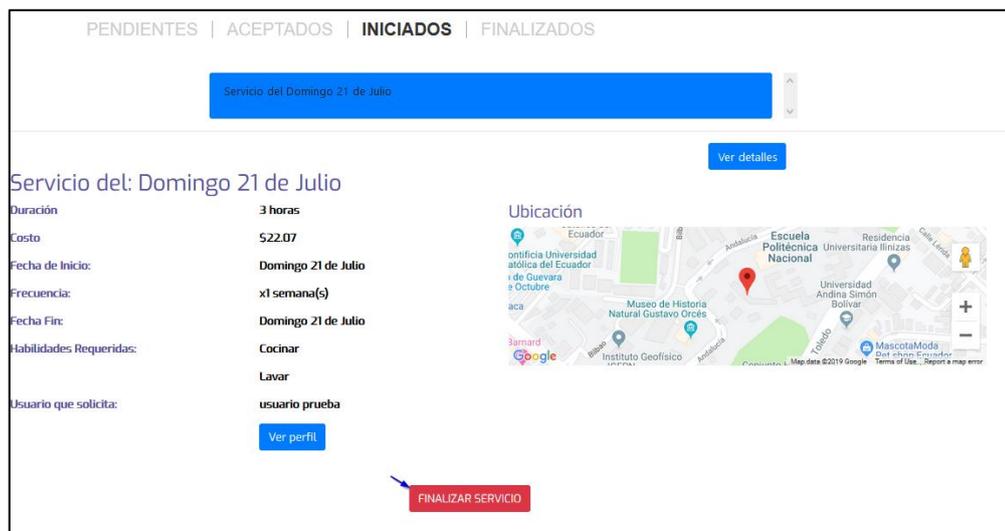


Figura 37. GUI para finalizar un servicio.

Calificación entre ofertante y solicitante

En la figura 38 se observa las interfaces gráficas para calificar, al solicitante por parte del ofertante y al ofertante por parte del solicitante respectivamente.

Califica a tu usuario!

Ponle una calificación a tu usuario y cuéntanos cualquier comentario que tengas

Nombre: usuario prueba



Calificación

★★★★★

Comentarios:

Muy Amable y cordial

Califica a tu empleado!

Ponle una calificación al empleado y cuéntanos cualquier comentario que tengas

Nombre: Roberta Hernandez



Calificación

★★★★★

Comentarios:

Puntual y respetuosa, excelente servicio

Figura 38. GUI para calificación de solicitante y ofertante.

Adaptación del Product Backlog

En la tabla 20 se visualiza la actualización del Product Backlog la finalizar el sprint 3, en donde las historias realizadas en este sprint finalizaron de forma exitosa.

Product Backlog				
Código	Historia de Usuario	Prioridad	Estimación	Estado
HU01	Pre-registro de ofertantes.	Alta	13	Finalizado
HU02	Registro de ofertantes en la plataforma.	Alta	21	Finalizado
HU03	Registro de nuevo solicitante.	Alta	21	Finalizado
HU04	Ingreso a la plataforma.	Alta	13	Finalizado
HU05	Ingreso de ubicación.	Alta	13	Finalizado
HU06	Selección de habilidades.	Alta	13	Finalizado
HU07	Ingresar horario de servicio.	Alta	21	Finalizado
HU08	Ofertantes disponibles.	Alta	21	Finalizado
HU09	Reserva del servicio.	Alta	13	Finalizado
HU10	Lista de servicios de solicitante.	Alta	13	Finalizado
HU11	Inicio y finalización del servicio.	Alta	21	Finalizado
HU12	Edición de domicilio u oficina.	Alta	21	Finalizado
HU13	Lista de los servicios de ofertante.	Alta	21	Finalizado
HU14	Calificación de ofertante a solicitante.	Alta	13	Finalizado
HU15	Calificación de solicitante a ofertante.	Alta	13	Finalizado
HU16	Cambio de contraseña.	Alta	13	Pendiente
HU17	Restablecimiento de contraseña.	Alta	13	Pendiente
HU18	Suspensión usuarios.	Alta	8	Pendiente
HU19	Edición de información de empleados.	Alta	13	Pendiente
HU20	Información estadística.	Alta	13	Pendiente

Tabla 20. Product Backlog adaptado al finalizar el Sprint 3.

2.7.4 Sprint Retrospective

En la figura 39 se observa el gráfico Burndown Chart para el Sprint 3. En general, no hubo problemas para presentar a tiempo los objetivos marcados por el Sprint, ya que las tareas que se realizaron consumían servicios que se implementaron previamente en los anteriores Sprints.

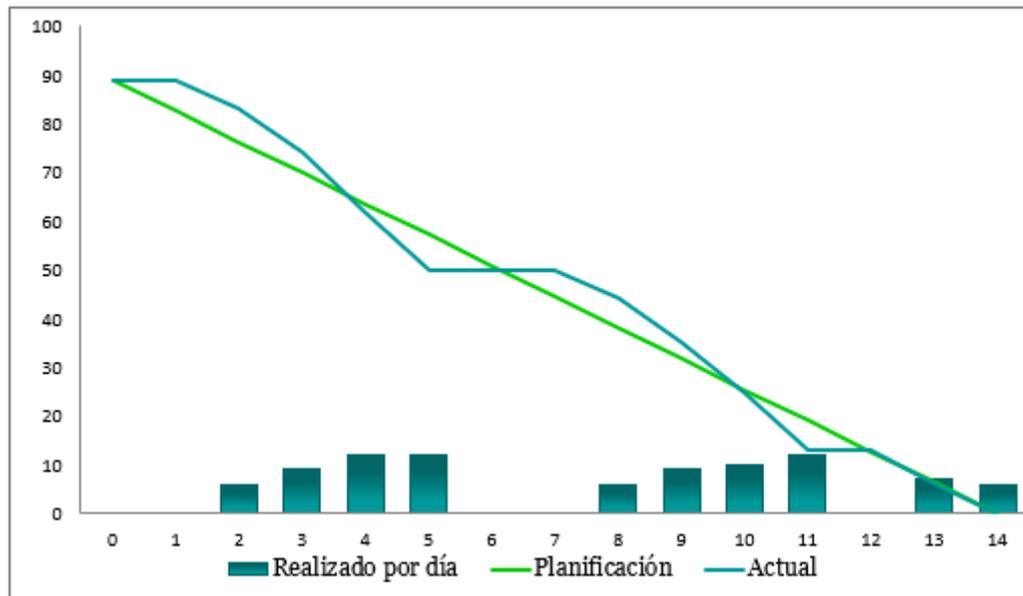


Figura 39. Burndown Chart del Sprint 3.

2.8 Sprint 4

2.8.1 Sprint Planning

Objetivo del Sprint

- Implementar los procesos de cambio y restablecimiento de contraseñas, suspensión de usuarios, edición de la información de ofertantes y configuración de datos del solicitante según el anexo IX.
- Desarrollar los métodos que permitan el almacenamiento, visualización, edición y gestión de la información generada para los procesos de cambio y restablecimiento de contraseñas, suspensión de usuarios, edición de la información de ofertantes y configuración de datos del solicitante.

Historias de Usuario

En la tabla 21 se encuentran las historias de usuarios realizadas durante el desarrollo del Sprint 4. (En Anexo V se encuentra información más detallada cada historia de usuario).

ID	Título	Estimación	Estado
HU16	Cambio de contraseña.	13	Desarrollo
HU17	Restablecimiento de contraseña.	8	Desarrollo
HU18	Suspensión usuarios.	13	Desarrollo
HU19	Edición de información de ofertantes.	13	Desarrollo
HU20	Información estadística.	13	Desarrollo

Tabla 21. Historias de usuario para el Sprint 4.

Sprint Backlog

En la tabla 22 se encuentra el Sprint Backlog donde se detalla las tareas involucradas en el desarrollo de cada una de las historias de usuario de este Sprint.

Sprint Backlog	
HISTORIA DE USUARIO	TAREAS
HU16	Implementación de interfaz gráfica de acuerdo con el diseño.
	Integración de la librería Bcrypt para comparar si la contraseña actual coincide con el registro hash en la base de datos.
	Integración de librería Joi para realizar validación de la nueva contraseña.
	Implementar función para reemplazar el hash de la nueva contraseña en la base de datos.
HU17	Implementación del formulario para recuperar la contraseña.
	Implementar un método de búsqueda por nombre de usuario o correo electrónico para enviar automáticamente una nueva contraseña.
	Generar nueva contraseña de forma automática.
HU18	Implementación de la interfaz gráfica de acuerdo con el diseño.
	Desarrollo de la función para listar a todos los ofertantes (empleado) y solicitantes (usuarios) registrados en la plataforma.
	Desarrollo de un método que permite realizar una búsqueda por nombre del ofertante (empleado) o solicitante (usuario).
	Desarrollo de un método que permita suspender al solicitante (usuario) u ofertante (empleado) seleccionado de la lista.
	Denegar el acceso a la plataforma a los usuarios suspendidos.
HU19	Implementación de interfaz gráfica de acuerdo con el diseño.
	Desarrollo de un método que liste todos los ofertantes (empleados).
	Desarrollo de un método que permita la búsqueda de ofertante (empleado) por nombre.
	Desplegar la información básica del ofertante en pantalla.
	Integración de un mapa con la API de Google Maps para ingresar la nueva dirección del ofertante (empleado).
	Desarrollo de una función que permita la actualización de la imagen de perfil y de documentos del ofertante.
	Integración con Amazon S3 para el almacenamiento de imágenes y documentos en la nube de forma segura.

HISTORIA DE USUARIO	TAREAS
HU20	Implementación de una interfaz gráfica de acuerdo con el diseño, utilizando gráficos estadísticos (barras).
	Implementación de un método para contabilizar el número de solicitantes y ofertantes.
	Implementación de un método para numerar las zonas donde se registre mayor demanda de servicios.
	Implementación de un método para contabilizar los usuarios según su género, edad y localización.

Tabla 22. Sprint Backlog del Sprint 4.

2.8.2 Ejecución del Sprint

A continuación, se presenta los detalles técnicos importantes que se realizaron durante el desarrollo del Sprint.

Integración de Amazon S3

Se realizó la implementación de Amazon S3 (ver sección 1.4) para el almacenamiento de la información multimedia de los usuarios (Solicitante y ofertante) como imagen de perfil, documentos y certificados.

En la figura 40 se observa la forma en que se almacena la información en Amazon S3. Por cada usuario se crea una carpeta que contendrá su imagen de perfil y documentos digitales necesarios para estar registrado en la plataforma.

<input type="checkbox"/>	Name ▾	Last modified ▾
<input type="checkbox"/>	AnaCaicedo	--
<input type="checkbox"/>	PedroMora	--
<input type="checkbox"/>	RobertaHernandez	--

Figura 40. Integración con Amazon S3.

Gráficas estadísticas

Para la visualización de graficas estadísticas se utilizó la librería Chart.js (ver sección 1.4) que permite la creación de gráficos de barras, áreas, líneas, dispersión, etc.

En la figura 41 se observa la integración de Chart.js con Angular para realizar cuadros estadísticos.

```

dibujar() {
  this.estadisticaService.empleadosSexo().subscribe(data => {
    const cantidad = data.map(res => res.Cantidad);
    const leyendas = data.map(res => res.Leyenda);
    const sum = cantidad.reduce(this.add);
    this.totalEmpleados = sum;
    this.chart = new Chart("canvas", {
      type: this.tipo,
      data: {
        labels: leyendas,
        datasets: [
          {
            label: "Cantidad",
            data: cantidad,
            borderColor: "#3cba9f",
            fill: false,
            backgroundColor: [
              "rgba(255, 99, 132, 0.2)",
              "rgba(54, 162, 235, 0.2)",
              "rgba(255, 206, 86, 0.2)",
              "rgba(75, 192, 192, 0.2)",
              "rgba(153, 102, 0, 0.2)",
              "rgba(255, 159, 64, 0.2)"
            ]
          }
        ]
      }
    });
  });
}

```

Figura 41. Integración de chart.js.

Restauración de Contraseñas

Bcrypt (ver sección 1.4) ofrece métodos que permiten gestionar contraseñas, almacenando el hash de estas y comparando si la contraseña ingresada coincide con el hash almacenado. En la figura 42 se observa la integración e implementación de Bcrypt en Node.js.

```

const bcrypt = require("bcrypt");
const SALT_ROUND = 10;
module.exports = {
  async hashPassword(password) {
    return await bcrypt.hash(password, SALT_ROUND);
  },
  async comparePassword(password, hash) {
    return await bcrypt.compare(password, hash);
  }
};

```

Figura 42. Gestión de contraseñas con Bcrypt.

Tokens JWT

JSON Web Tokens (ver sección 1.4) permite la gestión del tiempo que un usuario (solicitante u ofertante) está en la plataforma. En la figura 43 se observa la integración e implementación de JWT en Node.js.

```

if (!matchPassword) {
  return res.badRequest({ err: "4" });
}
if (user.bloqueado === true) {
  return res.badRequest({ err: "5" });
}
const token = JWTService.issuer({ user: user.id }, "1h");
if (user.firstLog === true) {
  await Usuario.update({ id: user.id }).set({ firstLog: false });
}
return res.ok({ token });

```

Figura 43. Implementación de JWT para generar Tokens.

Edición de domicilios de ofertantes

Debido al uso de geolocalización los ofertantes, para asegurar rapidez en la realización del servicio, estos no pueden cambiar su domicilio. Este es un trabajo realizado por el administrador de la plataforma ya que debe verificar a través de documentos que el usuario se encuentra viviendo en otro domicilio.

Suspensión de usuarios

En la figura 44 se observa la interfaz gráfica para la suspensión de usuarios de la plataforma. Se incrementó un campo en el modelo de usuario, el mismo que permite excluir a los usuarios (solicitantes y ofertantes) de sus actividades en la plataforma.

```
attributes: {
  nombre: { type: "string" },
  username: { type: "string", unique: true, required: true },
  password: { type: "string" },
  cedula: { type: "string" },
  email: { type: "string", unique: true, required: true },
  telefono: { type: "string" },
  contacto_extra: { type: "string" },
  imagenPerfil: { type: "string" },
  sexo: { type: "string", isIn: ["Masculino", "Femenino"] },
  fecha_nacimiento: { type: "number" },
  rol: { collection: "Rol", via: "usuario" },
  //Campo adicional para suspender un usuario
  bloqueado: { type: "boolean", defaultsTo: false },
}
};
```

Figura 44. Atributo que permite suspender a un usuario.

2.8.3 Sprint Review

Pruebas de aceptación

En la tabla 23 se presentan los criterios de aceptación definidos por cada historia de usuario del Sprint 4 (ver anexo V) junto con su respectiva revisión de cumplimiento y satisfacción por parte del Product Owner.

Historia de Usuario	Criterio de Aceptación	Cumple
HU16	Si la contraseña actual no coincide con el registro en la base de datos que tiene el usuario mostrar el mensaje "La contraseña actual no es correcta".	Si
	Verificar que la nueva contraseña cumple con los siguientes requisitos: <ul style="list-style-type: none">• Entre 8 y 16 caracteres.• Al menos un dígito.• Al menos una letra mayúscula.• Al menos una letra minúscula.	Si
	Mostrar un mensaje que recuerde al usuario las condiciones que debe tener la contraseña.	Si

Historia de Usuario	Criterio de Aceptación	Cumple
HU16	El botón “cambiar” se habilitará cuando los campos estén llenos y se haya verificado que la nueva contraseña cumpla con los requisitos indicados.	Si
HU17	Si el usuario o correo electrónico ingresado no se encuentra registrado, mostrar un mensaje que diga “Usuario no encontrado”.	Si
	Si se encuentra el usuario o correo electrónico, se debe generar una nueva contraseña automáticamente y enviarla al mail del usuario. Además, debe mostrar un mensaje que diga al usuario que la contraseña fue reestablecida con éxito	Si
HU18	Se visualiza una lista con todos los ofertantes y solicitantes registrados en la plataforma.	Si
	Debe existir una barra búsqueda por nombre de solicitante u ofertante.	Si
	Los usuarios suspendidos, no deben tener acceso a la plataforma.	Si
	Envió de un correo electrónico de forma automática al usuario, indicando que fue suspendido de la plataforma.	Si
HU19	Se visualizará una lista que contiene a todos los ofertantes.	Si
	Búsqueda por nombre de ofertante.	Si
	Visualizar la ubicación actual mediante un mapa con la posibilidad de edición.	Si
	Visualizar la información que esta almacenada actualmente acerca del ofertante.	Si
	Actualizar imagen de perfil y documentos incluyendo una visualización o vista previa de la imagen y el PDF que se va a actualizar.	Si
	El botón “guardar” se activa cuando el formulario está completo.	Si
HU20	Gráfica estadística de solicitantes (usuarios) y número de ofertantes (empleados) activos en la plataforma.	Si
	Gráfica estadística de ofertantes (empleados) en función de su género.	Si
	Gráfica estadística de la habilidad más solicitada.	Si

Tabla 23. Pruebas de aceptación del Sprint 4.

Incremento obtenido

Los entregables para el cuarto incremento son.:

Edición de la información de ofertante

Los requerimientos de la plataforma especifican que un ofertante (empleado) debe realizar servicios a una distancia máxima que inicialmente es de 4km de su ubicación. Para garantizar esto, el administrador de la plataforma es quien gestiona la información relacionada con la ubicación del ofertante, quien solicita el cambio de ubicación mediante la interfaz gráfica que se visualiza en la figura 45.

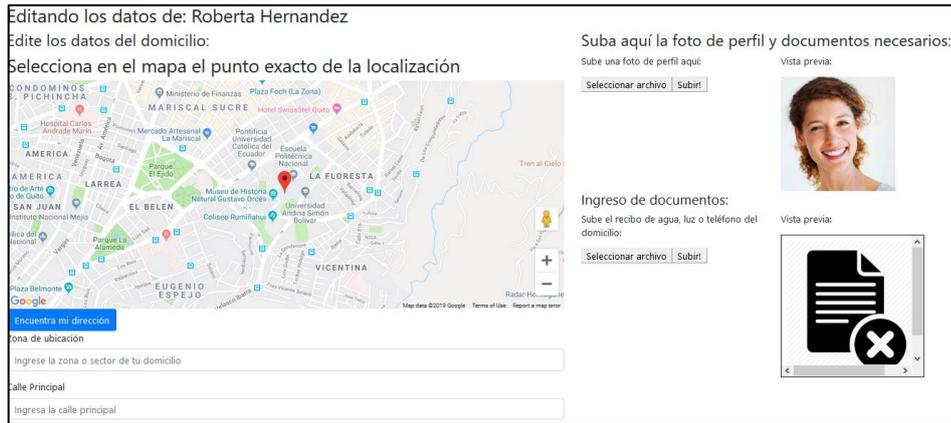


Figura 45. Edición de la información de ofertante por el administrador.

Cambio y restablecimiento de contraseña

Los usuarios de la plataforma (ofertantes o solicitantes) como medida de seguridad pueden cambiar de contraseña para así evitar suplantaciones de identidad. En la figura 46 se observa la interfaz gráfica en la que se solicita la contraseña actual como medida de seguridad para poder ingresar una nueva.

Cambio de contraseña

Contraseña actual

Contraseña nueva

[Cambiar](#)

Figura 46. GUI para cambio de contraseña.

Restablecimiento contraseña

En caso que el usuario (ofertante o solicitante) ya registrado pierda la contraseña para ingresar a la plataforma tiene la opción de restablecer su contraseña. En la figura 47 se observa la interfaz gráfica en donde el usuario debe ingresar su nombre de usuario o su correo electrónico y automáticamente se le enviará al correo una nueva contraseña generada ese momento.

UBER SERVICIOS [Inicio](#) [Aplicar](#)

Restaura tu contraseña aquí

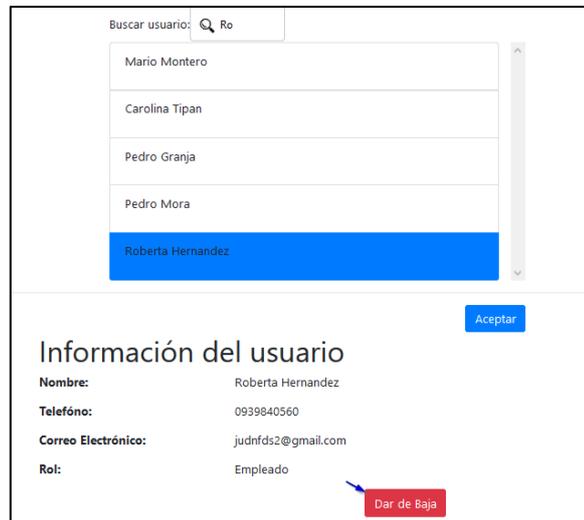
Ingrese su nombre de usuario o mail:

[Enviar](#)

Figura 47. GUI para restaurar contraseña.

Suspensión de usuarios

Cuando un usuario (solicitante u ofertante) realiza una falta grave, el administrador tiene la opción de suspenderlo de la plataforma. El usuario (solicitante u ofertante) suspendido no podrá ofertar ni solicitar servicios y se le niega el acceso a la plataforma, la suspensión de usuarios se realiza mediante la interfaz gráfica que se visualiza en la figura 48.



Buscar usuario:

- Mario Montero
- Carolina Tipan
- Pedro Granja
- Pedro Mora
- Roberta Hernandez**

Aceptar

Información del usuario

Nombre: Roberta Hernandez
Teléfono: 0939840560
Correo Electrónico: judnfd2@gmail.com
Rol: Empleado

Dar de Baja

Figura 48. GUI para suspender ofertantes o solicitantes.

Edición de información básica

Un usuario (solicitante u ofertante) puede cambiar su información básica como: nombre, email o número de contacto. Esta información debe estar siempre actualizada ya que es importante para la comunicación dentro de la plataforma (notificaciones, contacto entre ofertante y solicitante, avisos de mantenimiento, etc.). En la figura 49 se observa la interfaz gráfica que permite realizar la edición de información básica.



Mi perfil

Edita tus datos básicos

Nombre

Email

Número de contacto

Guardar

Figura 49. Edición de información de ofertantes y solicitantes.

Información Estadística

El administrador del sitio requiere ver información estadística del tráfico de uso, género y el número de ofertantes y solicitantes. Las gráficas deben mostrarse en forma de barras o de pastel. En la figura 50 se observa las gráficas estadísticas sobre servicios y genero de los usuarios.



Figura 50. GUI de gráficas estadísticas.

Adaptación del Product Backlog

En la tabla 24 se visualiza la actualización del Product Backlog la finalizar el Sprint 4, en donde las historias realizadas en este Sprint finalizaron de forma exitosa.

Product Backlog				
Código	Historia de Usuario	Prioridad	Estimación	Estado
HU01	Pre-registro de ofertantes.	Alta	13	Finalizado
HU02	Registro de ofertantes en la plataforma.	Alta	21	Finalizado
HU03	Registro de nuevo solicitante.	Alta	21	Finalizado
HU04	Ingreso a la plataforma.	Alta	13	Finalizado
HU05	Ingreso de ubicación.	Alta	13	Finalizado
HU06	Selección de habilidades.	Alta	13	Finalizado
HU07	Ingresar horario de servicio.	Alta	21	Finalizado
HU08	Ofertantes disponibles.	Alta	21	Finalizado
HU09	Reserva del servicio.	Alta	13	Finalizado
HU10	Lista de servicios de solicitante.	Alta	13	Finalizado
HU11	Inicio y finalización del servicio.	Alta	21	Finalizado
HU12	Edición de domicilio u oficina.	Alta	21	Finalizado
HU13	Lista de los servicios de ofertante.	Alta	21	Finalizado
HU14	Calificación de ofertante a solicitante.	Alta	13	Finalizado
HU15	Calificación de solicitante a ofertante.	Alta	13	Finalizado
HU16	Cambio de contraseña.	Alta	13	Finalizado
HU17	Restablecimiento de contraseña.	Alta	13	Finalizado
HU18	Suspensión usuarios.	Alta	8	Finalizado
HU19	Edición de información de empleados.	Alta	13	Finalizado
HU20	Información estadística.	Alta	13	Finalizado

Tabla 24. Product Backlog adaptado al finalizar el Sprint 4.

2.8.4 Sprint Retrospective

La figura 51 corresponde al gráfico Burndown para el Sprint 4. Durante este sprint, se tuvo complicaciones con la configuración del servidor de Amazon S3 para almacenar los archivos PDF e imágenes de los empleados registrados en la plataforma. La configuración debía ser correcta tanto en el servidor de Amazon, como en el backend y frontend para que se almacenen los archivos de forma segura. A pesar de esto, se logró completar el Sprint a tiempo.

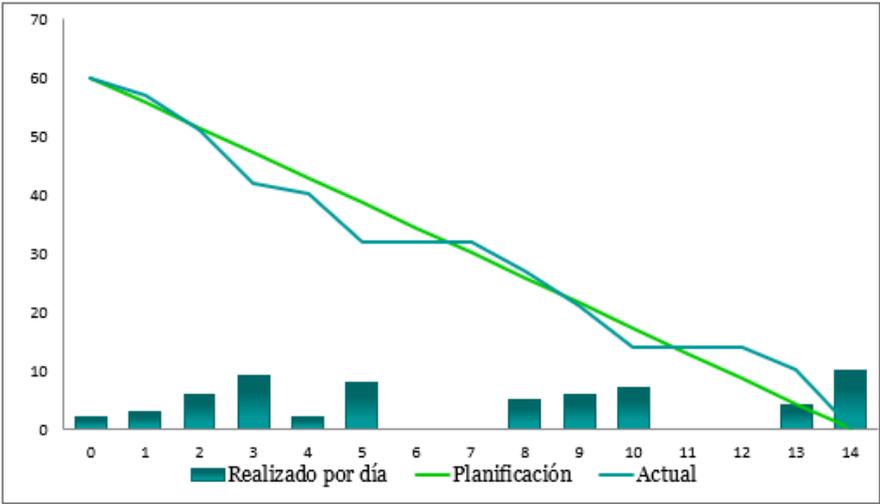


Figura 51. Burndown chart del Sprint 4.

3. RESULTADOS Y DISCUSIÓN

3.1 Producto Final

El producto final del proyecto es una plataforma digital que sirve de intermediario y punto de encuentro entre ofertantes y solicitantes de servicios domésticos ocasionales, las plataformas que poseen estas características son conocidas como un marketplace (ver sección 1.3.2).:

Consta de 3 roles.:

- Administrador.
- Solicitante de servicios.
- Ofertante de servicios.

Cada uno de estos roles tiene funciones específicas dentro de esta plataforma.

Administrador

En la figura 52 se observa la interfaz gráfica de bienvenida para el administrador de la plataforma.

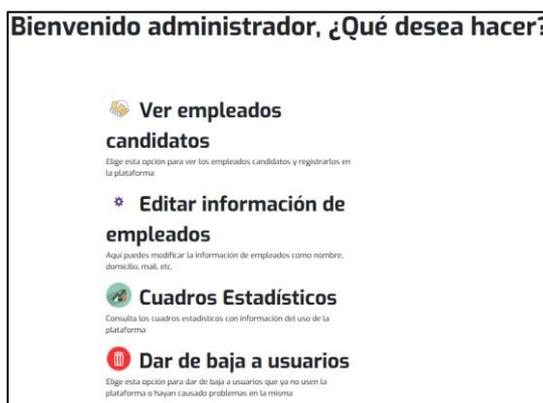


Figura 52. GUI de bienvenida par administrador.

Las actividades que realiza el administrador de la plataforma son.:

- Registro de ofertantes.
- Edición de información de ofertantes o empleados.
- Visualización de cuadros estadísticos.
- Suspende o dará de baja a ofertantes o solicitantes cualquier usuario de la plataforma del que se reporten de forma reiterada quejas.

Solicitante de servicios

En la figura 53 se observa la interfaz gráfica de bienvenida para un solicitante de servicios.

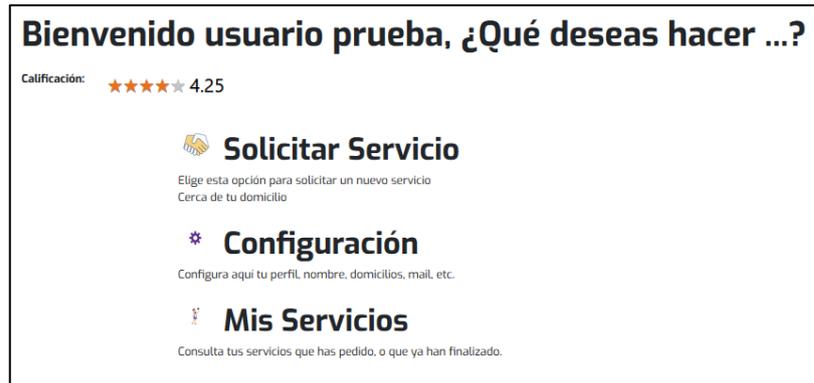


Figura 53. GUI de bienvenida par solicitante.

Las actividades que realiza el solicitante de servicios en la plataforma son.:

- Personalizar y solicitar un servicio.
- Pago por el servicio requerido.
- Configuración de sus datos personales.
- Gestión y visualización de sus servicios.

Ofertante de Servicios

En la figura 54 se observa la interfaz gráfica de bienvenida para un ofertante de servicios.

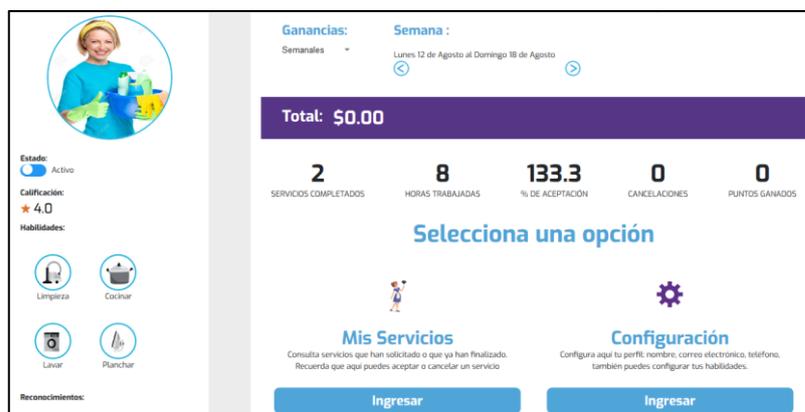


Figura 54. GUI de bienvenida par ofertante.

Las actividades que realiza el ofertante de servicios en la plataforma son.:

- Gestión y visualización de sus servicios.
- Configuración y edición de sus datos.
 - Habilidades.
 - Activa o no en la plataforma.
- Visualización de sus ganancias.

Servicio

El solicitante especifica los detalles que requiere para contratar un servicio, los cuales son su domicilio, habilidades requeridas, horario y frecuencia. Esta información permite encontrar a ofertantes que se ajusten a esos requerimientos y puedan realizar el servicio.

Un servicio atraviesa por diferentes etapas y sirve como vínculo entre un solicitante y un ofertante. Las diferentes etapas de un servicio son: Iniciado, pendiente, aceptado y finalizado a continuación de detalla cada una de las etapas mencionadas.

Etapas de un servicio

- **Pendiente**

Un servicio se origina cuando un solicitante personaliza según su necesidad el servicio.

El solicitante a través de la plataforma personaliza su servicio.

Un servicio pendiente consta con la siguiente información.:

- Ubicación.
- Tipo de domicilio.
- Precio aproximado del servicio.
- Habilidades requeridas.
- Horario.
- Frecuencia.
- Duración.
- Ofertante que realiza el servicio.
- Solicitante que requiere el servicio.

La información necesaria para el servicio es recolectada con el uso de la plataforma.

El solicitante ingresa la ubicación para que se realice el servicio, como se indica en la figura 55. La ubicación es seccionada mediante un mapa.

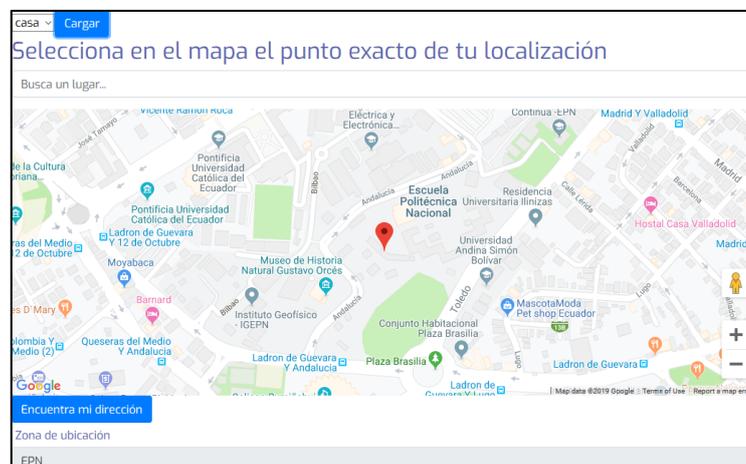


Figura 55. Ingreso de ubicación del servicio.

El solicitante ingresa las habilidades que requiere mediante la interfaz gráfica que se visualiza en la figura 56.

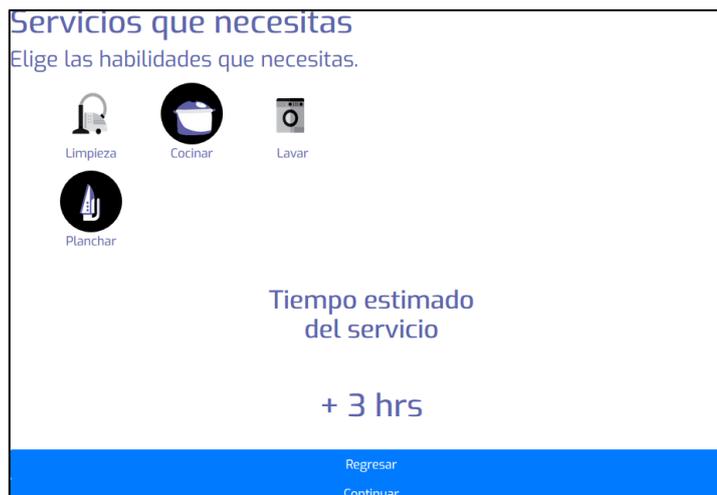


Figura 56. Selección de habilidades requeridas.

El solicitante ingresa el horario y la frecuencia para el servicio mediante la interfaz gráfica que se visualiza en la figura 57.

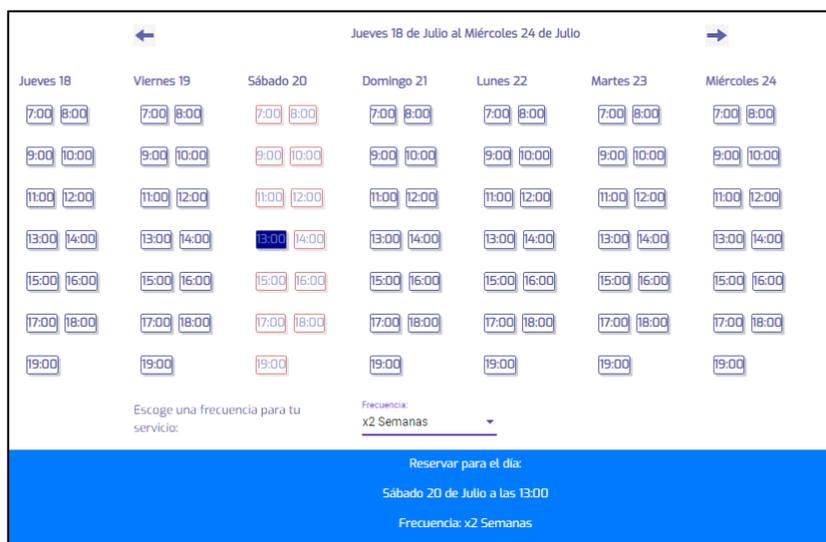


Figura 57. Selección del horario para el servicio.

En la figura 58 se observa la interfaz gráfica en que se muestra una lista y mediante la cual el solicitante escoge al ofertante para que realice el servicio.

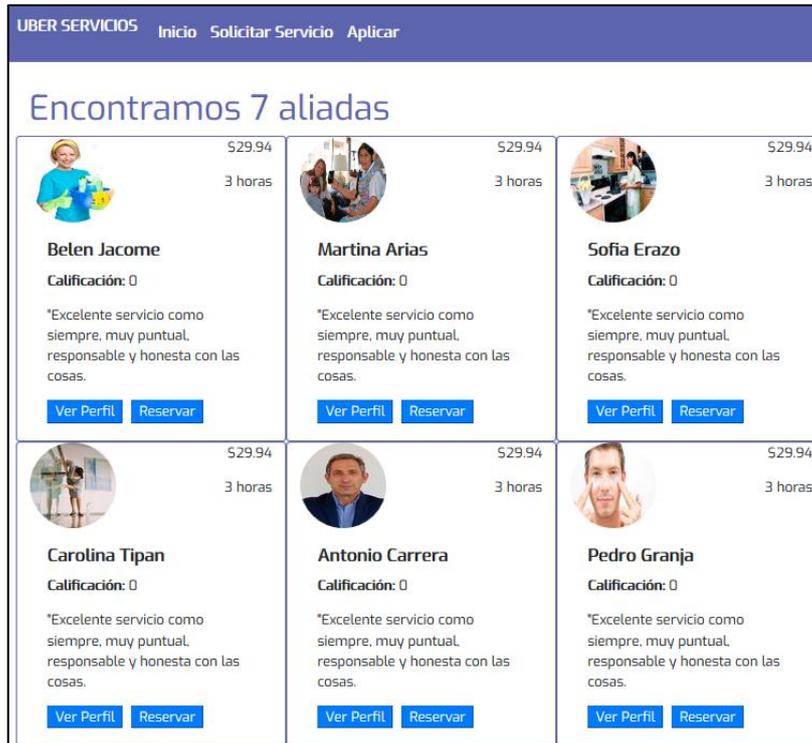


Figura 58. Selección de ofertante para el servicio.

Con el uso de Paypal realiza el pago por el servicio (ver figura 59).

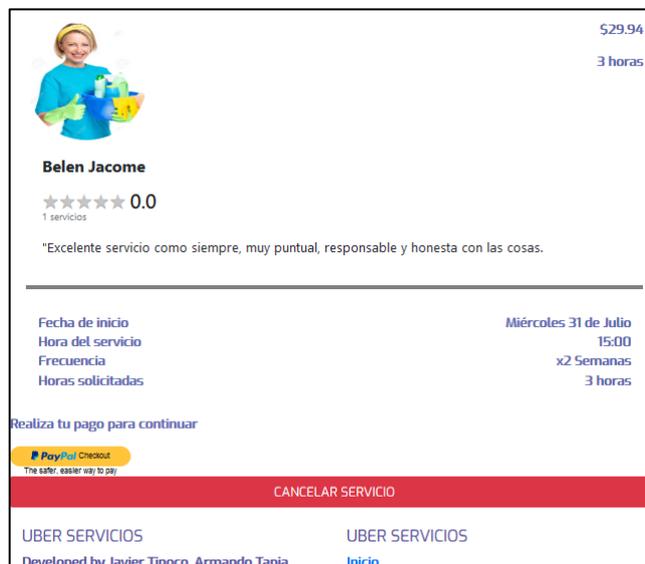


Figura 59. Pago del servicio.

El ofertante es el encargado de aceptar o rechazar un servicio pendiente. En este estado el ofertante puede.:

- Observar la ubicación donde se requiere el servicio.
- Ver el perfil del solicitante.
- Aceptar o rechazar el servicio.

En la figura 60 se observa la interfaz gráfica que permite realizar estas actividades sobre el servicio.



Figura 60. Servicio en estado pendiente.

- **Aceptado**

En la figura 61 se observa la interfaz gráfica donde el ofertante acepta el servicio y se dirige a la ubicación especificada para el servicio, una vez ahí comienza el servicio al pulsar el botón.

Aquí comienza un registro del tiempo real de la duración del servicio.



Figura 61. Servicio en estado aceptado.

- **Finalización**

En la figura 62 se observa la interfaz gráfica que permite que el ofertante finalice el servicio, para dejar constancia, debe pulsar el botón finalizar servicio, con lo que se registra la hora de finalización y se procede a calificar al solicitante.

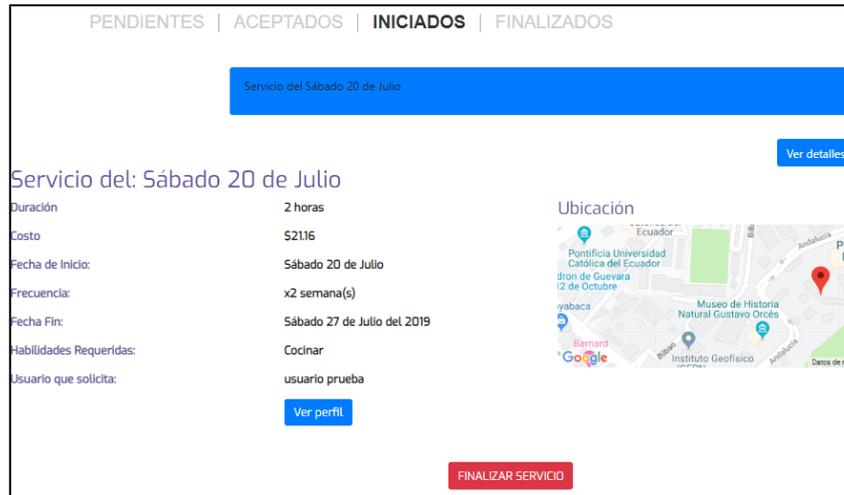


Figura 62. Servicio en estado finalizado.

Calificación entre ofertante y solicitante

En la figura 63 se observa las interfaces gráficas para calificación al solicitante y al ofertante respectivamente.

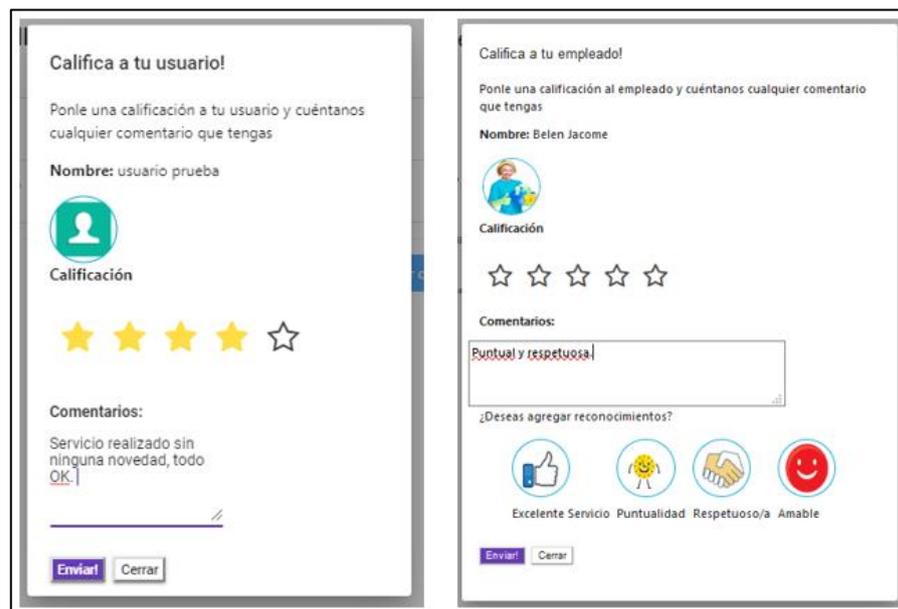


Figura 63. Calificación del servicio.

- **Finalizado**

Además, tanto solicitante como ofertante cuentan con un registro histórico de todos los servicios que han realizado en la plataforma como se visualiza en la figura 64.



Figura 64. Lista de servicios finalizados.

Análisis de los servicios

El cálculo del tiempo estimado se realiza en base a las habilidades seleccionadas por el usuario durante el proceso de solicitud del servicio. Se toma como duración mínima del servicio un tiempo de 2 horas y se agrega una hora extra por cada habilidad seleccionada.

En caso de requerir el servicio de limpieza, se hace un estimado con la información de los metros cuadrados del domicilio, número de baños y número de habitaciones. Posteriormente con el registro de inicio y finalización del servicio se obtienen las horas reales de la duración de este.

3.2 Pruebas con usuarios finales

Las pruebas de producto final se realizaron a un total de veinte personas donde diez de ellas tomaron el rol de ofertante de servicios y las diez restantes tomaron el rol de solicitante de servicios. Se evaluó la utilidad y la facilidad de uso percibidas al utilizar la plataforma digital de acuerdo con el modelo TAM (Technology Acceptance Model) [36].

Por cada rol (ofertante y solicitante) se realizaron un total de seis preguntas, tres para determinar la utilidad percibida y tres para la facilidad de uso, la respuesta de cada pregunta se ajustó a una escala de 1 a 5 (donde 1 es la nota más baja y 5 la más alta), las preguntas realizadas se encuentran en el anexo X. Las preguntas son similares, pero se analiza de forma separada debido a que el uso de la aplicación varía en función del rol con el que se ingrese.

3.2.1 Utilidad percibida

La utilidad percibida es el grado en el cual una persona cree que al utilizar un sistema particular mejoraría su trabajo [36]. A continuación, se presenta un análisis de los resultados promedio para la utilidad percibida por los ofertantes y solicitantes, las respuestas de cada pregunta se encuentran en el anexo XI.

Ofertante

La figura 65 es un gráfico del promedio de respuesta de cada pregunta con respecto a la utilidad percibida para los ofertantes. La primera pregunta se enfoca en el tiempo ahorrado para encontrar un servicio al utilizar la plataforma y obtuvo un promedio de (4.2/5). Esto se debe a que un ofertante registrado en la plataforma que aparece en los resultados de búsqueda, tiene más oportunidad de ser seleccionado para ofertar un servicio, ya que el solicitante puede ver su perfil, habilidades y comentarios.

La segunda pregunta se relaciona con la facilidad del proceso de contratación de servicios domésticos y obtuvo un promedio de (4.4/5). Esto se debe a que el ofertante, solo debe aceptar un servicio para iniciar a trabajar, dejando a un lado las entrevistas y contratos que se deben realizar normalmente en un proceso de contratación normal, sin el uso de la plataforma.

Finalmente, la última pregunta se refiere a la utilidad de la aplicación para ofertar y brindar servicios domésticos, obteniendo un promedio de (4.4/5). Esto se debe a que los ofertantes encuentran que la plataforma es un lugar indicado y específico para ofertar sus servicios, donde es más fácil que sean vistas y tienen más probabilidad de ser contratadas.

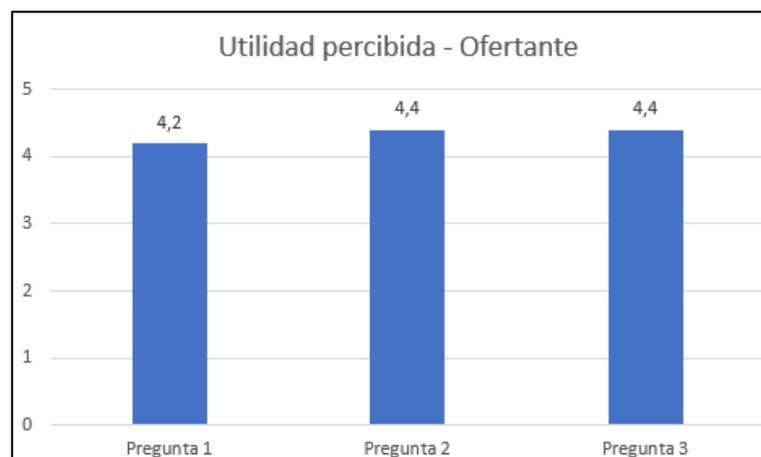


Figura 65. Resultados utilidad percibida de ofertante.

Solicitante

La figura 66 es un gráfico promedio de la utilidad percibida desde la perspectiva del solicitante de servicios. La primera pregunta realizada se enfoca en el tiempo ahorrado en encontrar ofertantes de servicios domésticos mediante el uso de la plataforma. El puntaje obtenido fue de (4.5/5), lo cual indica que en general a los usuarios no les tomó mucho tiempo encontrar una persona para que realice servicios domésticos en su domicilio, ya que la búsqueda de ofertantes en la plataforma es inmediata.

La segunda se relaciona con la facilidad en la contratación de ofertantes mediante el uso de la plataforma. Esta pregunta obtuvo una calificación promedio de (4.3/5), lo cual significa que en general los solicitantes pudieron contratar fácilmente un ofertante, siguiendo los pasos indicados por la plataforma.

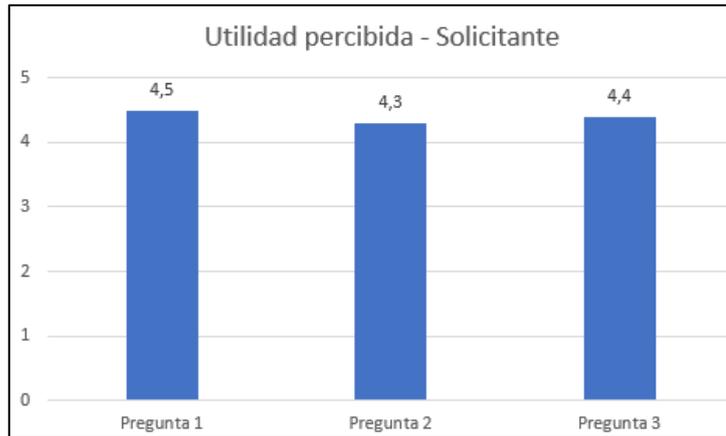


Figura 66. Resultados utilidad percibida de solicitante.

La tercera evalúa si el solicitante encuentra útil contratar servicios utilizando la plataforma. Esta pregunta obtuvo un resultado de (4.4/5) debido a la facilidad de buscar, contratar y pagar un servicio desde la plataforma.

3.2.2 Facilidad de uso percibida

La facilidad de uso percibida es el grado en el cual una persona cree que usando un sistema particular no conlleva demasiado esfuerzo [36]. A continuación, se presenta un análisis de los resultados promedio para la facilidad de uso percibida por los ofertantes y solicitantes, las respuestas de cada pregunta se encuentran en el anexo XI.

Ofertante

La figura 67 es un gráfico promedio de la facilidad de uso percibida desde la perspectiva del ofertante. La primera pregunta se enfoca en la facilidad para ofertar o brindar servicios utilizando la plataforma y obtuvo una calificación promedio de (3.9/5).

La segunda pregunta se enfoca en la interacción con la plataforma, la ubicación y funcionamiento de los botones, y demás funcionalidades. Esta pregunta recibió una calificación promedio de (3.2/5) debido a que inicialmente los ofertantes no están familiarizados con la interfaz y funcionamiento de la plataforma, y al tener varias opciones, era fácil para ellos sentirse un poco perdidos y con la necesidad de tener asistencia sobre su funcionamiento.

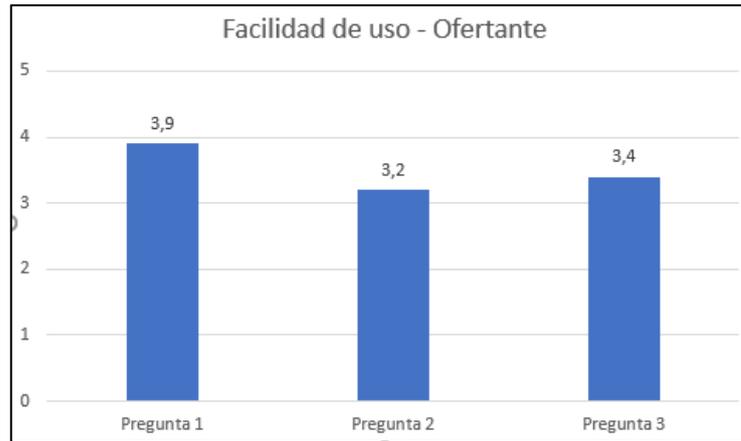


Figura 67. Resultados facilidad de uso de ofertante.

La tercera pregunta evalúa de manera general la facilidad en el uso de la plataforma, en esta pregunta se obtuvo un promedio de (3.4/5). Este resultado, y el de las anteriores preguntas, se deben a que los ofertantes en general, son un público que no está muy familiarizado con la tecnología y uso de aplicaciones, por lo cual se les dificulta un poco el uso de la plataforma y los pasos que deben seguir para tener una interacción adecuada.

Solicitante

La figura 68 es un gráfico promedio de la facilidad de uso percibida desde la perspectiva del solicitante. La primera pregunta se enfoca en la facilidad para solicitar un servicio mediante la plataforma. En esta pregunta se obtuvo un resultado de (3.8/5). Este resultado se da debido a la falta de entrenamiento del usuario respecto de la interfaz presentada, ya que, al tener varias opciones, puede requerir asistencia para entender su uso.

La segunda pregunta se enfoca en la interacción con la plataforma, ubicación y funcionamiento de los botones, y demás funcionalidades. Esta pregunta recibió una calificación promedio de (3.7/5). Al igual que la pregunta anterior, el resultado refleja que el usuario tiene una buena interacción con la plataforma, pero el desconocimiento de las interfaces inicialmente hace que tenga ciertas dificultades en su uso.

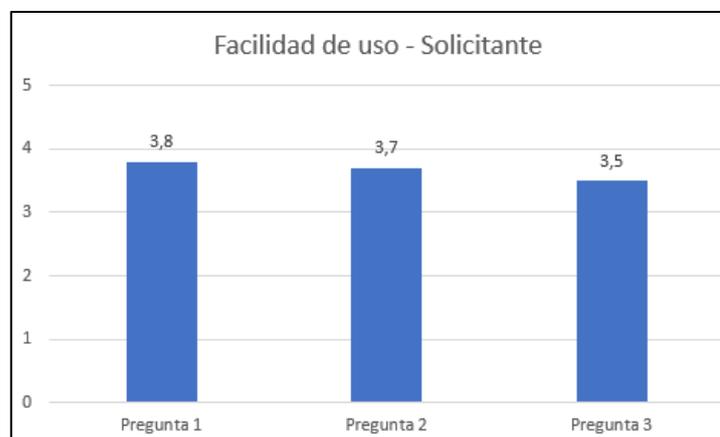


Figura 68. Resultados facilidad de uso de solicitante.

La tercera pregunta evalúa de manera general la facilidad en el uso de la plataforma. En esta pregunta se obtuvo un promedio de (3.5/5), lo cual indica que a pesar de que la plataforma intenta ser lo más amigable posible con el usuario y tener varias interfaces intuitivas para que no se pierda, se requiere asistencia o un entrenamiento previo al usuario para que comprenda su uso. También se debe tomar en cuenta que el público objetivo de esta plataforma son personas que no están familiarizados con el uso de tecnología, por lo que una capacitación previa del uso de la misma es indispensable.

Por lo tanto, se concluye que el problema con la facilidad de uso es el desconocimiento del usuario y no el diseño de la plataforma como tal.

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- La plataforma ofrece una opción más rápida y accesible para solicitar y ofertar servicios domésticos ocasionales, permitiendo a solicitantes encontrar de forma más rápida alguien de confianza que realice las tareas domésticas en su hogar. De igual forma, se permite a los ofertantes brindar servicios en una ubicación cercana a su domicilio evitando así que tenga que recorrer grandes distancias para dirigirse a su lugar de trabajo.
- El uso de SCRUM como marco de trabajo permitió una rápida adaptación a los cambios que surgieron en la etapa de desarrollo. Además, se cumplieron las fechas de entrega de forma según el cronograma inicial.
- El módulo de empleados permite a cada ofertante modificar su perfil, su estado (activo o inactivo) y también ofrece la posibilidad de aceptar o rechazar un servicio pendiente. Gracias a esto, los ofertantes tienen flexibilidad en el horario que deseen trabajar y la posibilidad de rechazar servicios en caso de que no puedan realizarlo.
- La implementación del módulo de administrador permite garantizar la identidad de los ofertantes registrados, además de que la información registrada de los mismos, como domicilio, nombre, foto de perfil, correo electrónico, sea verídica; ya que los ofertantes no pueden registrar, ni editar sus datos. Este proceso solo puede realizarlo un administrador a través de una solicitud previa por parte del ofertante. De esta forma, los solicitantes pueden estar seguros de que las personas que contratan son de confianza, ya que todos sus datos se han verificado previamente.
- La plataforma permite realizar un seguimiento a cada servicio por parte del ofertante y del solicitante, verificando el estado del servicio y además evaluar la calidad de dicho servicio mediante comentarios y calificaciones. Con esto, la plataforma puede garantizar la calidad del servicio y de sus ofertantes, pues en caso de incumplir con alguna norma o tener baja calificación, pueden ser expulsados.
- La seguridad en los cobros realizados mediante la plataforma está garantizada gracias a la utilización de PayPal. Este servicio de pagos permite la gestión segura de los datos sensibles de tarjetas de crédito como número, fecha de caducidad y el código de seguridad. Además, PayPal se ajusta a las necesidades de la plataforma, ya que permite realizar reembolsos, lo cual puede suceder en caso de que algún servicio sea cancelado o no se haya completado totalmente.

4.2 Recomendaciones

Para futuros trabajos se recomienda.:

- Considerando los tiempos de duración del servicio registrados en la plataforma, se debería desarrollar un modelo estadístico que permita predecir de manera más exacta la duración del servicio.
- Ampliar el radio de búsqueda de ofertantes de forma progresiva para que un solicitante siempre pueda encontrar un ofertante que realice el servicio requerido, aumentando el valor de este debido al coste del transporte.
- Para la gestión de acceso, autenticación, credenciales, doble factor de autenticación y acceso federado mediante usuarios de redes sociales se debería utilizar un proveedor de servicios como auth0 que proporciona escalabilidad y gestión óptima de las credenciales de usuarios.
- Implementar un módulo de quejas y sugerencias donde cada usuario (solicitante u ofertante) pueda reportar abusos y faltas de respeto para que el administrador después de realizar un seguimiento del caso proceda a suspender al usuario de la plataforma.
- Mejorar el diseño de las interfaces y la experiencia de usuario, con el fin de facilitar el uso de esta para usuarios ofertantes y solicitantes.
- Automatizar el proceso de cambio de domicilio de los ofertantes, donde el ofertante envíe un documento que garantice el cambio de domicilio y una vez comprobado este sea almacenado de forma automática.
- Automatizar el proceso de reembolsos para que este se realice de forma automática cada vez que se cancele un servicio y no de forma manual.
- Implementar otro servicio de pagos ya que en Ecuador no existe un porcentaje alto de personas que utilicen PayPal, lo cual limita el mercado al que puede llegar la plataforma.

5. Referencias Bibliográficas

- [1] INEC, “Boletín técnico N° 02-2019-ENEMDU,” p. 14.
- [2] El comercio, “Trabajadoras domésticas buscan visibilizarse.,” 2018. [Online]. Available: <https://www.elcomercio.com/actualidad/trabajadoras-domesticas-visibilizacion-derechos-ecuador.html>.
- [3] Casera, “Aplicación Casera.” [Online]. Available: <https://www.casera.ec/>.
- [4] F. Valdés Dal-Ré, “Escuela Internacional de Alta Formación en Relaciones Laborales y de Trabajo de ADAPT Comité de Gestión Editorial,” *Rev. Int. y Comp. Relac. Laborales y Derecho del Empl.*, vol. 1, no. 4, pp. 1–24, 2017.
- [5] S. Ginès i Fabrellas, A.; Gálvez Duran, “Sharing economy vs. uber economy y las fronteras del Derecho del Trabajo: la (des)protección de los trabajadores en el nuevo entorno digital,” *Rev. para el Análisis del Derecho*, vol. 1, pp. 1–44, 2016.
- [6] Ecommerce-nation, “Definición de marketplace,” 2018. [Online]. Available: <https://www.ecommerce-nation.es/que-es-un-marketplace-te-explicamos-todo-lo-que-debes-saber/>.
- [7] Eustat, “Definición de servicio doméstico.” [Online]. Available: http://www.eustat.eus/documentos/opt_0/tema_447/elem_5670/definicion.html.
- [8] J. Eduardo and H. Espinosa, “Colegio de Postgrados Jaime Eduardo Hidalgo Espinosa,” 2011.
- [9] I. Sommerville, Addison-Wesley es una marca de. 2011.
- [10] IBM, “Definición de servicio.” [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSMQ79_9.5.1/com.ibm.egl.pg.doc/topics/pegl_serv_overview.html.
- [11] gyires., “Definición de servicio web.” [Online]. Available: https://gyires.inf.unideb.hu/GyBITT/08/ch06.html?fbclid=IwAR0cU30Pj-cEWJFom52ATE69D6-8uR9r8KDU-NZzgrg9LLNKKrP_petlCw.
- [12] W3C, “Definición de WebService.” [Online]. Available: <https://www.w3.org/TR/ws-gloss/>.
- [13] Balsamiq Studios, “Balsamiq,” Balsamiq Wireframes. [Online]. Available: <https://balsamiq.com/wireframes/>.
- [14] Salesforce, “Heroku.” [Online]. Available: <https://www.heroku.com/platform>.
- [15] Datenschutz, “AGM.” [Online]. Available: <https://angular-maps.com/api-docs/>.
- [16] Bootstrap team, “Bootstrap.” [Online]. Available: <https://getbootstrap.com/>.
- [17] Hapi Family, “Joi,” 2019. [Online]. Available: <https://www.npmjs.com/package/@hapi/joi>.
- [18] MongoDB Documnetation, “MongoDB,” 2019. [Online]. Available: <https://www.mongodb.com/es/what-is-mongodb>.
- [19] 3T Software Labs, “Robomongo.” [Online]. Available: <https://robomongo.org/>.
- [20] C. Team, “Chart.js.” [Online]. Available: <https://www.chartjs.org/>.

- [21] Auth0, "JWT." [Online]. Available: <https://jwt.io/>.
- [22] AWS, "Amazon EC2 Definition." [Online]. Available: <https://aws.amazon.com/ec2/>.
- [23] Aws, "AmazonS3." [Online]. Available: https://docs.aws.amazon.com/es_es/AmazonS3/latest/dev/Welcome.html.
- [24] J. A. Cuartero, "Metodologías Ágiles: el nuevo paradigma empresarial," 2018. [Online]. Available: <https://www.americaeconomia.com/analisis-opinion/metodologias-agiles-el-nuevo-paradigma-empresarial-que-llega-para-querdarse>.
- [25] F. Hays, a Guide To the a Guide To the. 2015.
- [26] N. K. Rad and F. Turley, "The Scrum Master Training Manual," pp. 0–86, 2013.
- [27] J. Roche, "Definición de artefactos Scrum." [Online]. Available: <https://www2.deloitte.com/es/es/pages/technology/articles/artefactos-scrum.html>.
- [28] Mike Cohn, "Release planning." [Online]. Available: <https://www.mountaingoatsoftware.com/blog/release-planning-retiring-term-not-technique>.
- [29] Mountaingoatsoftware, "Sprint Backlog." [Online]. Available: <https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/sprint-backlog>.
- [30] K. Schwaber and J. Sutherland, "La Guía definitiva de Scrum: Las Reglas de Juego," Scrum.org, 2014.
- [31] SCRUM Colombia, "Incremento de Sprin," 2019. [Online]. Available: <https://scrumcolombia.org/incremento-de-producto/>.
- [32] Visual Paradigm, "Burndown." [Online]. Available: <https://www.visual-paradigm.com/scrum/scrum-burndown-chart/>.
- [33] D. Palminio, "Uso de release planning." [Online]. Available: <http://agilismoeningenieriadesoftware.blogspot.com/2016/08/scrum-release-planning-y-scrum.html>.
- [34] jmbeas, "Ventajas del release planning." [Online]. Available: <https://jmbeas.es/guias/release-plan/>.
- [35] R. Hughes, Agile Data Warehousing. .
- [36] Keh-Chyuan Tsai, " A Combined Phase and Force Compensation Method for Real-time Hybrid Testing," 15th World Conf. Earthq. Eng., vol. 13, no. 3, pp. 319–340, 2012.

6. Anexos

6.1 Anexo I: Historias de Usuario Épicas

Historia de Usuario épica	HUE01
Título: Gestión de la información del solicitante.	Prioridad: Alta
Descripción: Como usuario que solicita servicios (solicitante) requiero gestionar mi información, almacenar mi localización para posteriormente usar los servicios ofrecidos por la plataforma en mi casa u oficina.	

Tabla 25. Historia Épica HUE01.

Historia de Usuario épica	HUE02
Título: Gestión de la información del ofertante.	Prioridad: Alta
Descripción: Como persona que oferta servicios (ofertante) requiero gestionar mi información y las habilidades que ofrezco en la plataforma para aceptar o rechazar servicios.	

Tabla 26. Historia Épica HUE02.

Historia de Usuario épica	HUE03
Título: Gestión administrativa de la plataforma.	Prioridad: Alta
Descripción: Como administrador requiero registrar nuevos ofertantes, gestionar usuarios ofertantes y solicitantes de servicio además ver estadísticas de la plataforma.	

Tabla 27. Historia Épica HUE03.

Historia de Usuario épica	HUE04
Título: Proceso de solicitud de servicio.	Prioridad: Alta
Descripción: Como solicitante requiero adaptar un servicio a mis necesidades tanto en horario, frecuencia habilidades y ofertante seleccionado.	

Tabla 28. Historia Épica HUE04.

6.2 Anexo II: Historias de Usuario del Sprint 1

Historia de usuario	HU01
Título: Pre-registro de ofertante.	
Descripción: Como aspirante a ofertante de la plataforma requiero ingresar mi información para participar en el proceso de preselección.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
Conversación: Consta de una interfaz gráfica que permite ingresar los siguientes campos del aspirante: nombre, apellido, correo electrónico, género y fecha de nacimiento.	
En la parte superior debe visualizarse una lista donde se indique la información necesaria en caso de ser aceptado como ofertante. La lista debe especificar que	

el aspirante debe llevar: comprobante de domicilio, RUC, cédula de identidad, número de cuenta bancaria.
Criterios de Aceptación: <ul style="list-style-type: none"> • Comprobar validez del correo electrónico ingresado. Si no es válido, mostrar un mensaje de error al usuario. • Comprobar en la fecha de nacimiento, que el aspirante sea mayor de edad. Si es mayor de edad, permitir que se registre, caso contrario mostrar un mensaje que diga que solo se pueden registrar personas mayores de edad. • El botón de Aplicar debe estar habilitado solo si el formulario está completo y los campos ingresados son válidos. En caso de que el botón no esté habilitado, mostrar mensaje que diga que se deben llenar todos los campos y que deben ser válidos.

Tabla 29. Historia de usuario 1 (HU01).

Historia de usuario	HU02
Título: Registro ofertantes en la plataforma.	
Descripción: Como administrador requiero visualizar una lista con todos los nombres de los aspirantes (pre-registrados) para seleccionar a cualquiera de ellos y registrarlo como ofertante.	
Prioridad: Alta.	Esfuerzo: 21 puntos.
<p>Conversación: Consta de una interfaz gráfica que permite visualizar una lista de todos los aspirantes, y se podrá realizar una búsqueda por nombre en la parte superior. Una vez seleccionado un aspirante se observará la información básica y se tendrá un botón para continuar con el registro de esa persona en la plataforma.</p> <p>En el registro se ingresa los siguientes campos: Cédula de ciudadanía, Contacto extra en caso de emergencia, nombre de usuario, habilidades, fotografía, récord policial, plantilla de electricidad, agua o teléfono. Para el registro de domicilio se deberá seleccionar el punto exacto en un mapa de Google Maps y además los siguientes campos: zona de ubicación, calle principal y calle secundaria.</p>	
Criterios de Aceptación: <ul style="list-style-type: none"> • Búsqueda de usuarios pre-registrados por nombre. • El botón para registrar al ofertante se habilitará solo cuando el formulario está completo y todos los campos son válidos. Caso contrario se deberá mostrar un mensaje diciendo que todos los campos deben estar llenos y deben ser válidos. • Se debe enviar un mail al correo electrónico del ofertante, una vez finalizado el registro de forma automática. Este mail debe contener las credenciales de acceso a la plataforma. • La generación de la contraseña debe ser automática y solo debe ser vista por el ofertante en el correo enviado. 	

Tabla 30. Historia de usuario 2 (HU02).

Historia de usuario	HU03
Título: Registro de nuevo solicitante.	

Descripción: Como persona que solicita un servicio (usuario) requiero registrarme en la aplicación para solicitar servicios por medio de la plataforma.	
Prioridad: Alta.	Esfuerzo: 21 puntos.
Conversación: Consta de una interfaz gráfica en donde el usuario ingresa su nombre, apellido, correo electrónico, teléfono, nombre de usuario y contraseña para registrarse en la plataforma.	
Criterios de Aceptación.:	
<ul style="list-style-type: none"> • Verificar que el correo electrónico ingresado sea válido. • Verificar que los campos de contraseña y confirmación de contraseña coincidan. • Validar que el correo ingresado no se encuentre registrado en la plataforma previamente. • Validar que el nombre de usuario sea único en la plataforma. • Verificar que la contraseña ingresada tenga.: <ul style="list-style-type: none"> ○ Entre 8 y 16 caracteres. ○ Al menos un dígito. ○ Al menos una letra mayúscula. ○ Al menos una letra minúscula. • El botón “regístrate” debe estar habilitado solo cuando el formulario está completo y todos los campos son válidos. 	

Tabla 31. Historia de usuario 3 (HU03).

Historia de usuario	HU04
Título: Ingreso a la plataforma.	
Descripción: Como usuario registrado (ofertante, solicitante o administrador), requiero ingresar a la plataforma mediante mis credenciales personales para acceder a sus servicios.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
Conversación: Consta de una interfaz gráfica de ingreso o login a la plataforma donde se ingresa el usuario y la contraseña. Se impide el ingreso cuando las credenciales no son válidas.	
Criterios de Aceptación	
<ul style="list-style-type: none"> • El campo de la contraseña no debe mostrar el contenido. • El botón “Ingresar” debe habilitarse solo si el formulario esta completo. • Si al momento de ingresar, las credenciales no son válidas, mostrar el siguiente mensaje: “El usuario o contraseña está equivocado. Vuelve a intentarlo.” 	

Tabla 32. Historia de usuario 4 (HU04).

Historia de usuario	HU05
Título: Registro de domicilio.	
Descripción: Como solicitante requiero registrar un domicilio para indicar donde se realizará el servicio.	

Prioridad: Alta.	Esfuerzo: 13 puntos.
Conversación: Consiste en una interfaz gráfica que contiene un mapa donde se debe indicar la ubicación exacta del domicilio a través de un marcador que indicará la latitud y longitud del punto especificado. Además, se ingresarán los siguientes campos como información adicional para facilitar la ubicación al ofertante que acepte un servicio: zona de ubicación, calle principal y secundaria, piso o número de departamento e información extra.	
Criterios de Aceptación: <ul style="list-style-type: none"> • Si previamente el solicitante guardó uno de sus domicilios, se debe mostrar una lista con estos domicilios, y una opción para seleccionarlos y cargar su información en el formulario. • Integración de un mapa con Google Maps para seleccionar la ubicación del domicilio. • Debe existir una barra de búsqueda para encontrar con mayor facilidad la ubicación donde se realizará el servicio. • El botón “Continuar” debe estar habilitado cuando el formulario está completo y todos los campos son válidos. 	

Tabla 33. Historia de usuario 5 (HU05).

6.3 Anexo III: Historias de Usuario del Sprint 2

Historia de usuario	HU06
Título: Selección de habilidades.	
Descripción: Como solicitante requiero seleccionar las habilidades que necesito para el servicio que deseo contratar.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
Conversación: Consta de una interfaz gráfica que contiene una imagen y descripción por cada una de las habilidades: limpieza, cocina, lavado y planchado. Cada una puede ser marcada o desmarcada según sea necesario. Cuando el usuario escoge la habilidad de limpieza, aparecerá un formulario para recopilar información adicional del domicilio. Esta nueva información deberá tener los siguientes datos: número de habitaciones, número de baños y metros cuadrados.	
Criterios de Aceptación: <ul style="list-style-type: none"> • Marcar y desmarcar las habilidades requeridas. • El formulario de información extra del domicilio solo aparece cuando se selecciona la habilidad de limpieza. • Si previamente se seleccionó un domicilio ya guardado, debe mostrar la información adicional en el formulario, en caso de tenerla. • Dentro del formulario de información adicional del domicilio, restringir los campos numéricos a enteros positivos. • El botón “continuar” se debe habilitar solo si hay al menos una habilidad seleccionada. • Mostrar en pantalla el tiempo estimado para el servicio. 	

Tabla 34. Historia de usuario 6 (HU06).

Historia de usuario	HU07
Título: Ingresar horario de servicio.	
Descripción: Como solicitante requiero ingresar el horario del servicio para adaptarlos a mis necesidades.	
Prioridad: alta.	Esfuerzo: 21 puntos.

<p>Conversación: Consta de una interfaz gráfica, donde se muestra la semana actual, a partir del día actual en el que estoy usando la plataforma. En cada día se muestra un horario de 7 am a 5pm para que el usuario escoja el día y a la hora deseada del servicio. De igual forma escoge la frecuencia que puede ser de mínimo una semana (única vez) o máximo 4 semanas.</p>
<p>Criterios de Aceptación</p> <ul style="list-style-type: none"> • Permitir que el usuario solo pueda escoger una hora por día. • Se deben bloquear las horas del día actual que ya pasaron. • Si la duración estimada del servicio excede la hora máxima de trabajo para un ofertante (7 de la noche), desde la hora inicial, entonces se debe bloquear dichas horas. • Se debe tener la posibilidad de escoger el inicio de un servicio hasta 3 semanas después de la actual. • El botón de reservar debe estar habilitado solo si se ha seleccionado al menos un día y un horario, y una frecuencia para el servicio. En caso contrario el usuario no puede continuar con el proceso de contratación del servicio.

Tabla 35. Historia de usuario 7 (HU07).

Historia de usuario	HU08
Título: Empleados disponibles.	
Descripción: Como solicitante requiero visualizar una lista de empleados disponibles, junto con su perfil, para seleccionar el que sea de mi preferencia.	
Prioridad: Alta.	Esfuerzo: 21 puntos.
<p>Conversación: Es una interfaz gráfica en la que se muestra una lista de empleados que están cerca de la ubicación del solicitante y se cumplen con las necesidades del servicio solicitado. Cada empleado se mostrará como una tarjeta de identificación con su nombre, calificación, foto de perfil, número de servicios y opiniones. Dentro de cada tarjeta, también debe existir un botón para reservar a dicho empleado, y otro para mostrar su perfil completo.</p>	
<p>Criterios de Aceptación:</p> <ul style="list-style-type: none"> • Lista de ofertantes que se encuentren máximo a 4km a la redonda de la ubicación donde se solicitó el servicio. • La lista de ofertantes contiene solo ofertantes activos. • La lista de ofertantes debe contener solo ofertantes que cumplan las habilidades requeridas para el servicio. • Mostrar fotografía, calificación y los últimos 3 comentarios que ha recibido el ofertante. • Al dar clic en el botón de “Reservar”, continuar con el proceso de contratación del servicio con el empleado elegido. 	

Tabla 36. Historia de usuario 8 (HU08).

Historia de usuario	HU09
Título: Reserva del servicio.	
Descripción: Como solicitante requiero ver un resumen general del servicio solicitado para verificar que los datos sean correctos y proceder con el pago.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
<p>Conversación: Consta de una interfaz gráfica donde se muestra la imagen de perfil del</p>	

empleado seleccionado, su calificación y opiniones. Además, se muestra la fecha de inicio, horario, frecuencia, costo y tiempo de duración del servicio.
Criterios de Aceptación: <ul style="list-style-type: none"> • Realizar el pago a través de la plataforma Paypal. • El servicio se confirma solo si se realiza el pago exitosamente. Caso contrario, debe existir un botón que permita cancelar el servicio. • Si se guarda el servicio, se debe enviar una notificación a el empleado seleccionado, con el fin de que pueda aceptar o rechazar el mismo.

Tabla 37. Historia de usuario 9 (HU09).

Historia de usuario	HU10
Título: Lista de servicios contratados por el solicitante.	
Descripción: Como solicitante requiero visualizar en una lista todos mis servicios por estado (Aceptados, Iniciados o Finalizados) para acceder a sus detalles y verificar su progreso.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
Conversación: Consta de una interfaz gráfica que contiene una lista de los servicios del solicitante, la cual se puede filtrar por los siguientes estados: pendientes, aceptados y finalizados. Al final de la lista se encuentra un botón que permite ver los detalles del servicio seleccionado. Este botón se habilitará solo si se ha seleccionado algún servicio de la lista. Al dar clic en el botón, se mostrará la siguiente información del servicio: duración, ubicación, costo, fecha de inicio y fin, nombre del ofertante y un enlace para ver su perfil.	
Criterios de Aceptación <ul style="list-style-type: none"> • Filtro de servicios según su estado. • Animación de carga al aplicar un filtro a la lista de servicios. • Servicios ordenados de forma cronológica. • El botón “ver detalles” está activo solo cuando hay un servicio seleccionado. • Los detalles del servicio deben aparecer en la misma página. • Al dar clic en el enlace del perfil del empleado, se debe visualizar el perfil en una nueva página. 	

Tabla 38. Historia de usuario 10 (HU10).

6.4 Anexo IV: Historias de Usuario del Sprint 3

Historia de usuario	HU11
Título: Control de progreso del servicio.	
Descripción: Como ofertante requiero cambiar el estado de un servicio para controlar e informar al solicitante el progreso del servicio.	
Prioridad: Alta.	Esfuerzo: 21 pts.
Conversación: Dentro de la lista de servicios del ofertante debo tener un botón que permita cambiar el estado del servicio, dependiendo su estado actual. Los estados por los que puede pasar un servicio son: “pendiente”, “aceptado”, “iniciado”, “finalizado” o “cancelado”.	
Criterios de Aceptación: <ul style="list-style-type: none"> • Si la lista de servicios está filtrada por “Pendientes”, debe haber dos 	

<p>botones. Uno para aceptar el servicio, y otro para cancelar el servicio.</p> <ul style="list-style-type: none"> • Si la lista de servicios está filtrada por “Aceptados”, debe haber un botón para iniciar el servicio. • El botón para iniciar el servicio actualiza el estado a “iniciado” y guarda en la base de datos la hora exacta en la que comienza el servicio en el campo “hora de inicio”. • Si la lista de servicios está filtrada por “Iniciados”, debe haber un botón para finalizar el servicio. • El botón para finalizar el servicio actualiza el estado a “finalizado” y guarda en la base de datos la hora exacta en la que termina el servicio en el campo “hora de fin”. • Si la lista de servicios está filtrada por “Finalizados”, no debe haber ningún botón ya que es el último estado posible. • En cada cambio de estado se debe enviar una notificación al solicitante, informando el nuevo estado del servicio.

Tabla 39. Historia de usuario 11 (HU11).

Historia de usuario	HU12
Título: Actualización de información de domicilios.	
Descripción: Como solicitante requiero editar la información de mis domicilios para indicar los nuevos datos a la plataforma.	
Prioridad: Alta.	Esfuerzo: 21 puntos.
<p>Conversación: Consta de una interfaz gráfica donde se mostrará una lista con mis domicilios guardados. Al seleccionar uno, se cargará el detalle en un formulario que contendrá: el número de baños, de habitaciones y de metros cuadrados, un mapa con la ubicación, zona, calle principal y secundaria y piso o número de casa. En este formulario puedo registrar los nuevos cambios y se podrá guardar a través de un botón. También debe existir un botón para cancelar la operación actual.</p>	
<p>Criterios de Aceptación</p> <ul style="list-style-type: none"> • Se debe mostrar la lista de domicilios, solo en caso de que se haya guardado uno previamente. Los tipos de domicilio que se pueden guardar son casa u oficina. • El mapa debe mostrar la ubicación actual del domicilio y permitir seleccionar una nueva ubicación. El mapa debe tener una barra de búsqueda de ubicaciones. • El botón “guardar” se habilita solo cuando el formulario está completo y todos los campos son válidos, caso contrario debe mostrar un mensaje que informe al usuario que debe llenar todos los campos y que deben ser válidos. 	

Tabla 40. Historia de usuario 12 (HU12).

Historia de usuario	HU13
Título: Lista de los servicios de ofertante (empleado).	
Descripción: Como ofertante requiero visualizar en una lista todos mis servicios por estado (Pendientes, Aceptados, Iniciados o Finalizados) para acceder a sus detalles y verificar su progreso.	
Prioridad: Alta.	Esfuerzo: 21 puntos.
<p>Conversación: Consta de una interfaz gráfica que contiene una lista de servicios, los cuales pueden ser filtrados por su estado como: pendientes, aceptados, iniciados y finalizados. Al final de la lista se encuentra un botón que permite ver los detalles del servicio</p>	

<p>seleccionado. Este botón se habilitará solo si se ha seleccionado algún servicio de la lista.</p> <p>Al dar clic en el botón, se mostrará la siguiente información del servicio: duración, ubicación, costo, fecha de inicio y fin, nombre del solicitante y un enlace para ver su perfil.</p>
<p>Criterios de Aceptación:</p> <ul style="list-style-type: none"> • Filtro de servicios según su estado. • Animación de carga al aplicar un filtro a la lista de servicios. • Servicios ordenados de forma cronológica. • El botón “ver detalles” está activo solo cuando hay un servicio seleccionado. • Los detalles del servicio deben aparecer en la misma página. • Al dar clic en el enlace del perfil del solicitante, se debe visualizar el perfil en una nueva página.

Tabla 41. Historia de usuario 13 (HU13).

Historia de usuario	HU14
Título: Calificación de ofertante a solicitante.	
Descripción: Como persona ofertante requiero calificar al solicitante que recibió mi servicio y dejar mis comentarios y opiniones.	
Prioridad: Alta.	Esfuerzo: 13 puntos.
<p>Conversación:</p> <p>Consta de un popup que aparecerá cuando el ofertante termine el servicio. El popup debe tener la imagen de perfil del solicitante del servicio (usuario) seguida de 5 estrellas que representan la calificación a el usuario.</p> <p>Además, se presenta una caja de texto donde el usuario escribirá su comentario u opinión sobre el solicitante o la experiencia general del servicio.</p> <p>La calificación y el comentario no son obligatorios.</p> <p>Debe existir un botón para enviar la calificación y otro para cerrar el popup en caso de que el ofertante no quiera realizar la calificación.</p>	
<p>Criterios de Aceptación:</p> <ul style="list-style-type: none"> • Visualizar el nombre del solicitante (usuario). • Visualizar la foto de perfil. • En la calificación, se debe mostrar 5 estrellas de color blanco y cuando el empleado pulse sobre una deben marcarse de amarillo. • Los comentarios pueden ser opcionales, por lo que se debe poder enviar la calificación, aunque el comentario este vacío. • El botón enviar debe estar habilitado solo si se ha seleccionado un número de estrellas, caso contrario debe existir la opción de cerrar para omitir el paso de calificación. 	

Tabla 42. Historia de usuario 14 (HU14).

Historia de usuario	HU15
Título: Calificación de solicitante a ofertante.	
Descripción: Como persona solicitante requiero calificar al ofertante que realizó mi servicio y dejar mis comentarios y opiniones.	
Prioridad: Alta	Esfuerzo: 13 puntos
<p>Conversación:</p> <p>Consta de un popup que aparecerá automáticamente cuando el ofertante haya finalizado el servicio y el solicitante ingrese a la plataforma. El popup debe tener la imagen de perfil del ofertante del servicio (usuario) seguida de 5 estrellas que representan la calificación a el usuario y una lista de reconocimientos (Amable,</p>	

<p>Respetuoso/a, Puntual, Excelente Servicio) que se deseen dar al ofertante. Además, se presenta una caja de texto donde el usuario escribirá su comentario u opinión sobre el ofertante o la experiencia general del servicio. Debe existir un botón para enviar la calificación y otro para cerrar el popup en caso de que el solicitante no quiera realizar la calificación.</p>
<p>Criterios de Aceptación:</p> <ul style="list-style-type: none"> • Visualizar el nombre y foto de perfil del ofertante. • En la calificación, se debe mostrar 5 estrellas de color blanco y cuando el solicitante pulse sobre una deben marcarse de amarillo. • Deben aparecer popups por cada servicio que tiene el solicitante sin calificar, cada vez que ingrese a la plataforma. • Los comentarios y reconocimientos pueden ser opcionales, por lo que se debe poder enviar la calificación, aunque esos campos estén vacíos. • El botón enviar debe estar habilitado solo si se ha seleccionado un número de estrellas, caso contrario debe existir la opción de cerrar para omitir el paso de calificación. • Si el solicitante no desea calificar el servicio y da clic en cerrar, ya no debe volver a aparecerle el popup.

Tabla 43. Historia de usuario 15 (HU15).

6.5 Anexo V: Historias de Usuario del Sprint 4

Historia de usuario	HU16
Título: Cambio de contraseña	
Descripción: Como usuario (ofertante o solicitante) requiero cambiar mi contraseña para tener más seguridad en el acceso a mi perfil.	
Prioridad: Media.	Esfuerzo: 13 puntos.
Conversación: Es una parte de la interfaz gráfica de configuración, donde se presentarán los campos contraseña actual y contraseña nueva para que el usuario pueda cambiar.	
<p>Criterios de Aceptación:</p> <ul style="list-style-type: none"> • Si la contraseña actual no coincide con el registro en la base de datos que tiene el usuario mostrar el mensaje "La contraseña actual no es correcta". • Verificar que la nueva contraseña cumple con los siguientes requisitos: <ul style="list-style-type: none"> ○ Entre 8 y 16 caracteres. ○ Al menos un dígito. ○ Al menos una letra mayúscula. ○ Al menos una letra minúscula. • Mostrar un mensaje que recuerde al usuario las condiciones que debe tener la contraseña. • El botón "cambiar" se habilitará cuando los campos estén llenos y se haya verificado que la nueva contraseña cumpla con los requisitos indicados. 	

Tabla 44. Historia de usuario 16 (HU16).

Historia de usuario	HU17
Título: Restablecimiento de contraseña.	
Descripción: Como usuario (ofertante o solicitante) requiero restablecer mi contraseña para obtener una nueva en caso de haber olvidado la anterior.	
Prioridad: Media.	Esfuerzo: 13 puntos.
Conversación: Consta de una interfaz gráfica donde se solicitará el mail o nombre de usuario y un botón para enviar la nueva contraseña al correo electrónico.	
Criterios de Aceptación: <ul style="list-style-type: none"> • Si el usuario o correo electrónico ingresado no se encuentra registrado, mostrar un mensaje que diga “Usuario no encontrado”. • Si se encuentra el usuario o correo electrónico, se debe generar una nueva contraseña automáticamente y enviarla al mail del usuario. Además, debe mostrar un mensaje que diga al usuario que la contraseña fue reestablecida con éxito. 	

Tabla 45. Historia de usuario 17 (HU17)

Historia de usuario	HU18
Título: Suspensión usuarios.	
Descripción: Como administrador requiero suspender de la plataforma a usuarios solicitantes y ofertantes, que incumplan con las normas establecidas de comportamiento para asegurar la calidad y confianza de los servicios.	
Prioridad: Alta.	Esfuerzo: 8 puntos.
Conversación: Es una interfaz gráfica que contiene una lista de todos de los usuarios, contendrá un campo que permitirá la búsqueda por nombre. Al seleccionar un elemento de la lista y pulsa ver detalles se mostrará el nombre, rol, teléfono y correo electrónico además de un botón “dar de baja” que inhabilita al usuario (empleado o usuario seleccionado).	
Criterios de Aceptación: <ul style="list-style-type: none"> • Se visualiza una lista con todos los ofertantes y solicitantes registrados en la plataforma. • Debe existir una barra búsqueda por nombre de solicitante u ofertante. • Los usuarios suspendidos, no deben tener acceso a la plataforma. • Envío de un correo electrónico de forma automática al usuario, indicando que fue suspendido de la plataforma. 	

Tabla 46. Historia de usuario 18 (HU18).

Historia de usuario	HU19
Título: Edición de información de ofertantes.	
Descripción: Como administrador requiero editar la información de los ofertantes para tener la siempre su información actualizada.	
Prioridad: Media.	Esfuerzo: 13 puntos.
Conversación: Es una interfaz gráfica que permite editar información de ofertantes como la ubicación (zona, calle principal y secundaria y número de casa.), foto de perfil, documentos y número de cuenta bancaria.	
Criterios de Aceptación: <ul style="list-style-type: none"> • Se visualizará una lista que contiene a todos los ofertantes. 	

- Búsqueda por nombre de ofertante.
- Visualizar la ubicación actual mediante un mapa con la posibilidad de edición.
- Visualizar la información que esta almacenada actualmente acerca del ofertante.
- Actualizar imagen de perfil y documentos que incluyen una vista previa de la imagen y el pdf a subir.
- El botón "Guardar" se activa cuando el formulario esta completo.

Tabla 47. Historia de usuario 19 (HU19).

Historia de usuario	HU20
Título: Información estadística.	
Descripción: Como administrador, requiero visualizar información estadística del uso de la plataforma para obtener información que soporte futuras inversiones en la plataforma.	
Prioridad: Baja.	Esfuerzo: 13 puntos.
Conversación: Es una interfaz gráfica que se muestran gráficos de barras o pastel, de usuarios por género, servicios más requeridos y el número de usuarios y empleados.	
Criterios de Aceptación: Los siguientes cuadros estadísticos deben ser visualizados en forma de Barras, Rosquilla y Pie. <ul style="list-style-type: none"> • Visualizar información del número de usuarios y número de empleados en la plataforma activos en la plataforma. • Empleados en función del género. • Cual de todos los servicios es más requerido mediante la plataforma. 	

Tabla 48. Historia de usuario 20 (HU20).

6.6 Anexo VI: Diseño de interfaces de usuario final Sprint I

Pre-registro.

Figura 69. Mock de pre-registro.

Registro de ofertante en la plataforma.

A browser window showing a user profile page for 'Armando Tapia' on 'http://servicios.com/login'. The page includes a navigation bar with 'Servicios', 'Inicio', and the user's name. The profile details are as follows:

- Nombre:** Roberta
- Apellido:** Hernandez
- Correo:** roberto01@email.com
- UserName:** Roberto01
- Imagen Perfil:** A circular profile picture of a woman.
- Ubicación:** A map showing a location in a city.
- Documentos:** A blue document icon.
- Habilidades:** Four icons representing skills: 'Limpieza' (cleaning), 'Cocinar' (cooking), 'Lavar' (washing), and 'Planchar' (ironing).

Figura 70. Mock registro de ofertante.

Registro de nuevo usuario.

A browser window showing a registration form on 'http://Servicios.com/login'. The form includes a navigation bar with 'SERVICIOS', 'Inicio', 'Solicita Ya!', 'Aplica Ya!', and a 'Login' button. The registration fields are:

- Nombre:
- Apellido:
- Correo electrónico:
- Teléfono:
- Nombre de usuario:
- Contraseña:
- Confirmar Contraseña:

At the bottom, there is a 'Regístrate' button.

Figura 71. Mock registro nuevo usuario.

Ingreso a la plataforma.

A browser window showing a login page on 'http://Servicios.com/login'. The page features a navigation bar with 'SERVICIOS', 'Inicio', 'Solicita Ya!', 'Aplica Ya!', and a 'Login' button. The login form is overlaid on a background image of a couple sitting on a couch with a laptop. The form includes:

- Usuario:
- Password:
- Iniciar sesión:

Below the form, there is a promotional message: 'DEDICA TU TIEMPO A LO QUE TE GUSTA. En Recomendando sabemos que cuando la gente trabaja en equipo, en un ambiente inspirador, usando tecnología, las cosas salen mucho mejor.' and a 'SOLICITA YA' button.

Figura 72. Mock ingreso a la plataforma.

Ingreso de ubicación.

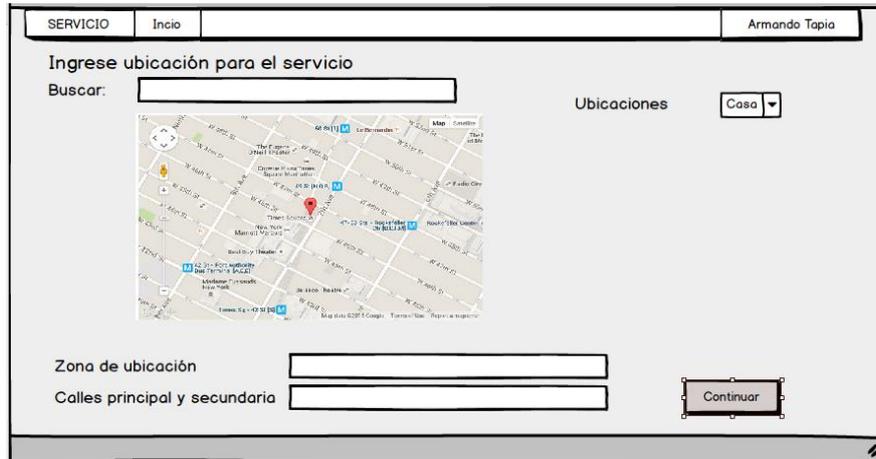


Figura 73. Mock ingreso de ubicación.

6.7 Anexo VII: Diseño de interfaces de usuario final Sprint II

Selección de habilidades.

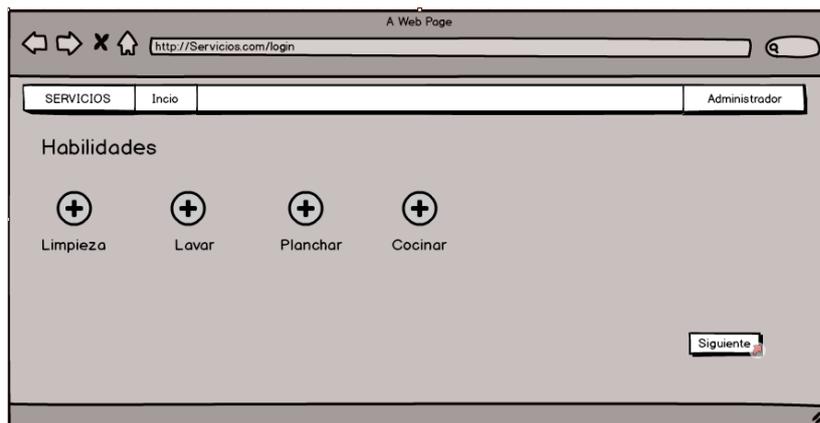


Figura 74. Mock selección de habilidades.

Ingresar horario de servicio.

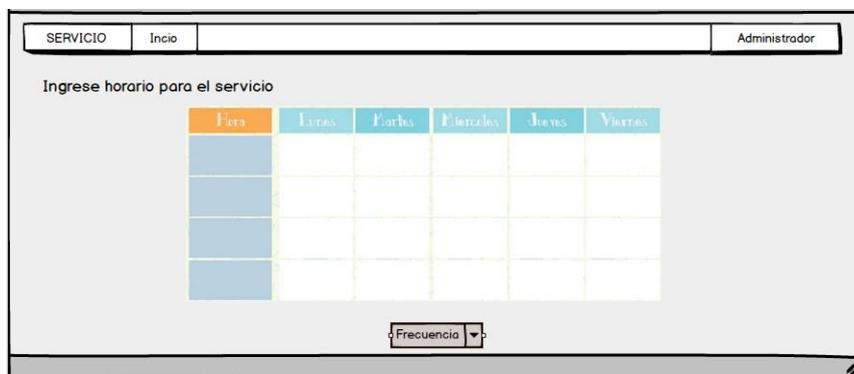


Figura 75. Mock ingreso de horario.

Ofertantes disponibles.

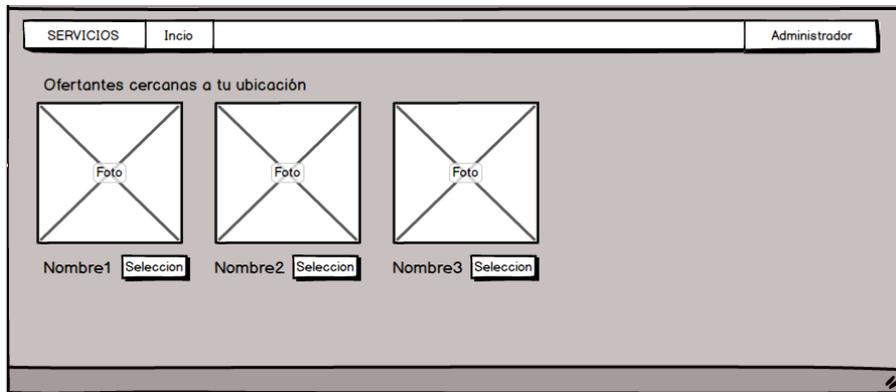


Figura 76. Mock de ofertantes disponibles.

Reserva del servicio.

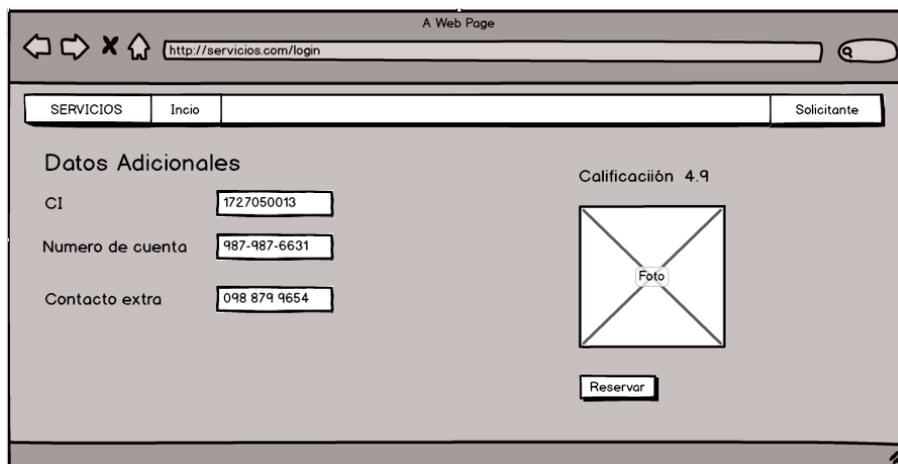


Figura 77. Mock de reserva de servicio.

Lista de servicios del solicitante.

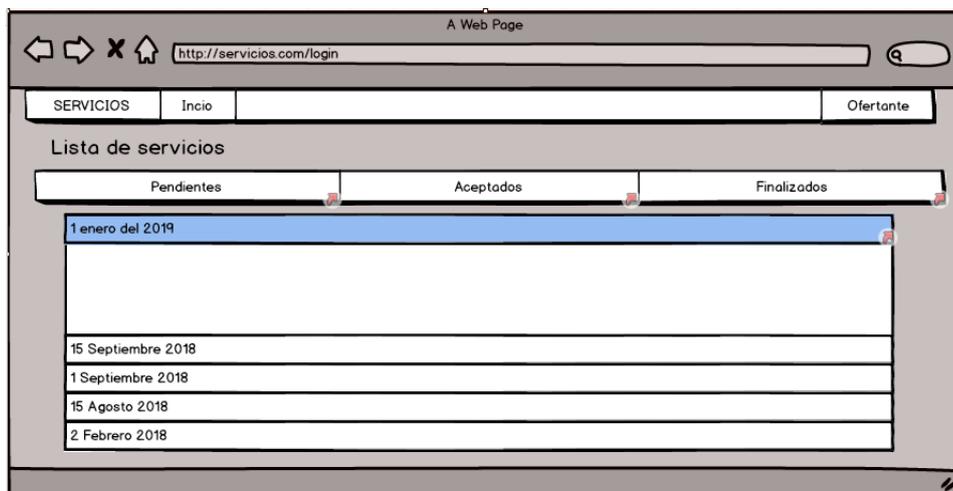


Figura 78. Mock de lista de servicios solicitante.

6.8 Anexo VIII: Diseño de interfaces de usuario final Sprint III

Edición de domicilio.



A Web Page
http://Servicios.com/login

SERVICIOS Inicio Solicitante

Edición de oficina o Selección de tu dirección

Sector Floresta

Calle principal Madrid

calle secundaria Lerida

Numero de casa oe3-654

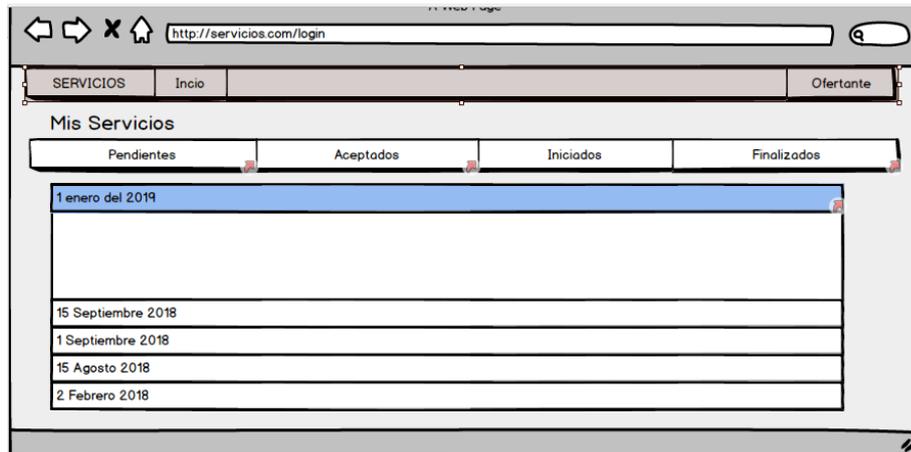
Información extra Diagonal a la EPN

Guardar

The mockup shows a web browser window with a navigation bar containing 'SERVICIOS', 'Inicio', and 'Solicitante'. The main content area is titled 'Edición de oficina o Selección de tu dirección'. On the left, there are input fields for 'Sector' (Floresta), 'Calle principal' (Madrid), 'calle secundaria' (Lerida), 'Numero de casa' (oe3-654), and 'Información extra' (Diagonal a la EPN). On the right, there is a map titled 'Selección de tu dirección' with a red location pin. A 'Guardar' button is located at the bottom right.

Figura 79. Mock edición de domicilio.

Lista de servicios del ofertante.



http://servicios.com/login

SERVICIOS Inicio Ofertante

Mis Servicios

Pendientes Aceptados Iniciados Finalizados

1 enero del 2019

15 Septiembre 2018

1 Septiembre 2018

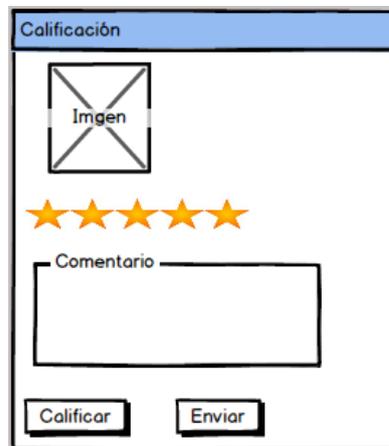
15 Agosto 2018

2 Febrero 2018

The mockup shows a web browser window with a navigation bar containing 'SERVICIOS', 'Inicio', and 'Ofertante'. The main content area is titled 'Mis Servicios'. Below the title, there are four tabs: 'Pendientes', 'Aceptados', 'Iniciados', and 'Finalizados'. A list of services is displayed, with the first item '1 enero del 2019' highlighted in blue. Other items include '15 Septiembre 2018', '1 Septiembre 2018', '15 Agosto 2018', and '2 Febrero 2018'.

Figura 80. Mock de lista de servicios del ofertante.

Calificación solicitante y ofertante.



Calificación

Imagen

★★★★★

Comentario

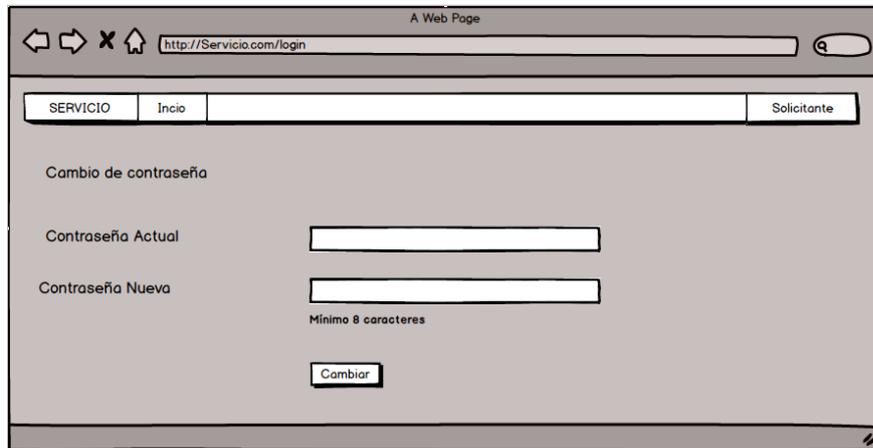
Calificar Enviar

The mockup shows a rating form titled 'Calificación'. It features a placeholder for an image labeled 'Imagen'. Below the image is a five-star rating system, represented by five yellow stars. Underneath the stars is a text input field labeled 'Comentario'. At the bottom, there are two buttons: 'Calificar' and 'Enviar'.

Figura 81. Mock de calificación.

6.9 Anexo IX: Diseño de interfaces de usuario final Sprint IV

Cambio de contraseña.



A Web Page
http://Servicio.com/login

SERVICIO Inicio Solicitante

Cambio de contraseña

Contraseña Actual

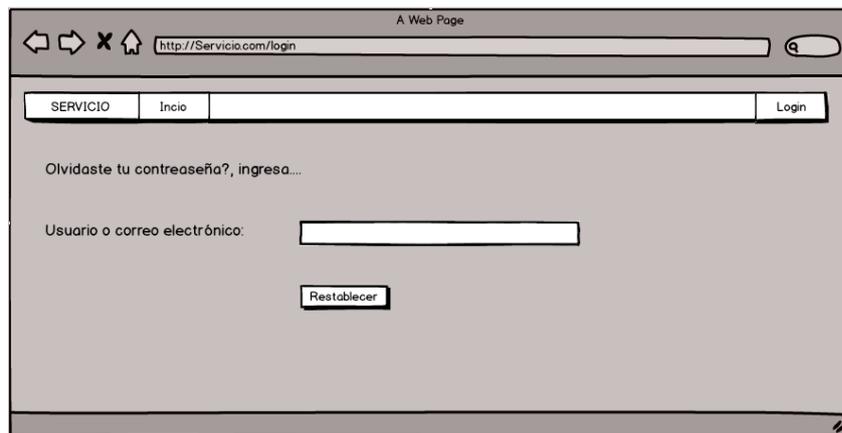
Contraseña Nueva

Mínimo 8 caracteres

Cambiar

Figura 82. Mock cambio de contraseña.

Restablecimiento de contraseña.



A Web Page
http://Servicio.com/login

SERVICIO Inicio Login

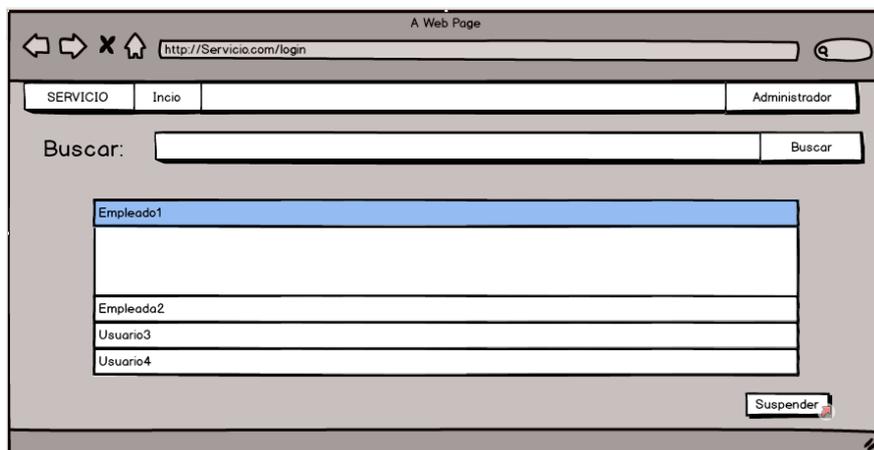
Olvidaste tu contraseña?, ingresa...

Usuario o correo electrónico:

Restablecer

Figura 83. Mock restablecimiento de contraseña.

Suspensión de usuario.



A Web Page
http://Servicio.com/login

SERVICIO Inicio Administrador

Buscar: Buscar

Empleado1
Empleado2
Usuario3
Usuario4

Suspender

Figura 84. Mock de suspensión de usuario.

Edición de información de ofertantes.

A Web Page
http://Servicios.com/login

SERVICIOS Inicio Administrador

Ofertante: Nombre del ofertante...

Sector: Floresta

Calle principal: Madrid

calle secundaria: Lerida

Numero de casa: oe3-654

Información extra: Diagonal a la EPN

Ofertante

Documento

Selecciona la dirección

Guardar

Figura 85. Mock de información de ofertantes.

Información estadística.



Figura 86. Mock de información estadística.

6.10 Anexo X: Encuestas realizadas

Cuestionario de utilidad de uso percibida.

Solicitante

Solicitante

Utilidad de uso percibida.

La aplicación le ayudó a encontrar ofertantes de servicios domésticos de forma más rápida? *

1 2 3 4 5

De ninguna manera Absolutamente

La aplicación le facilitó el proceso de contratación de servicios domésticos? *

1 2 3 4 5

De ninguna manera Absolutamente

Encuentra útil la aplicación para contratar servicios domésticos? *

1 2 3 4 5

De ninguna manera Absolutamente

Figura 87. Preguntas al solicitante sobre utilidad percibida.

Ofertante

Solicitante

Utilidad de uso percibida.

La aplicación le ayudó a encontrar ofertantes de servicios domésticos de forma más rápida? *

1 2 3 4 5

De ninguna manera Absolutamente

La aplicación le facilitó el proceso de contratación de servicios domésticos? *

1 2 3 4 5

De ninguna manera Absolutamente

Encuentra útil la aplicación para contratar servicios domésticos? *

1 2 3 4 5

De ninguna manera Absolutamente

Figura 88. Preguntas al ofertante sobre utilidad percibida.

Cuestionario de facilidad de uso percibida.

Solicitante

Solicitante

Facilidad de uso.

Le fue fácil contratar un servicio a través de la plataforma?

1 2 3 4 5

De ninguna manera Absolutamente

La interacción con la plataforma fue clara y entendible?

1 2 3 4 5

De ninguna manera Absolutamente

En general le resulto sencillo utilizar la plataforma?

1 2 3 4 5

De ninguna manera Absolutamente

Figura 89. Preguntas al solicitante sobre facilidad de uso.

Ofertante

Ofertante

Facilidad de uso

La aplicación le ayudó a encontrar oportunidades de trabajo de forma más rápida? *

1 2 3 4 5

De ninguna manera Absolutamente

La aplicación le facilitó el proceso de contratación del servicio? *

1 2 3 4 5

De ninguna manera Absolutamente

Encuentra útil la aplicación para ofertar y brindar servicios domésticos? *

1 2 3 4 5

De ninguna manera Absolutamente

Figura 90. Preguntas al ofertante sobre facilidad de uso.

6.11 Anexo XI: Resultados de las encuestas.

Utilidad percibida

Ofertante

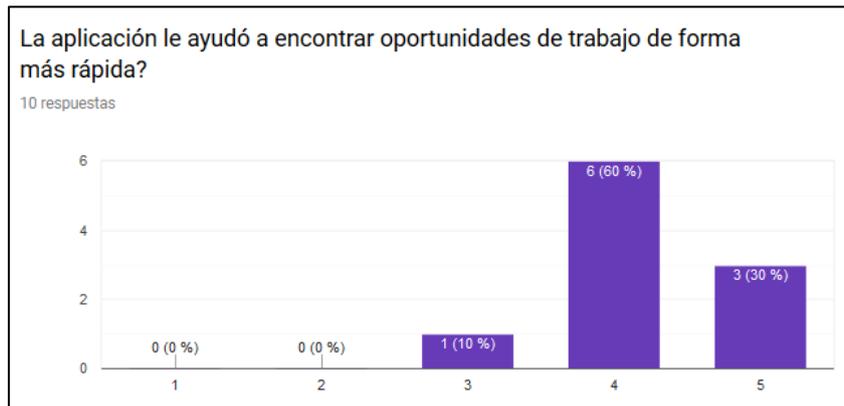


Figura 91. Respuestas de ofertante para pregunta 1 de utilidad percibida.



Figura 92. Respuestas de ofertante para pregunta 2 de utilidad percibida.



Figura 93. Respuestas de ofertante para pregunta 3 de utilidad percibida.

Solicitante

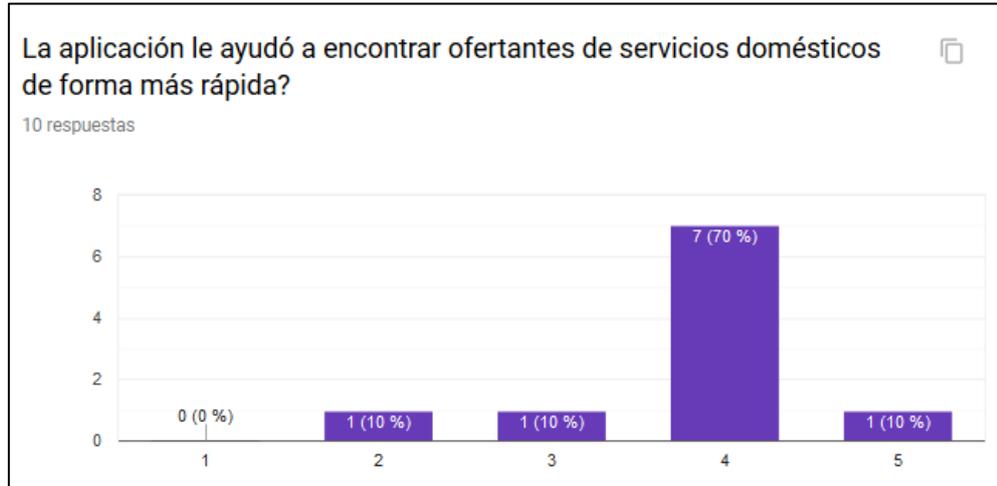


Figura 94. Respuestas de solicitante para pregunta 1 de utilidad percibida.

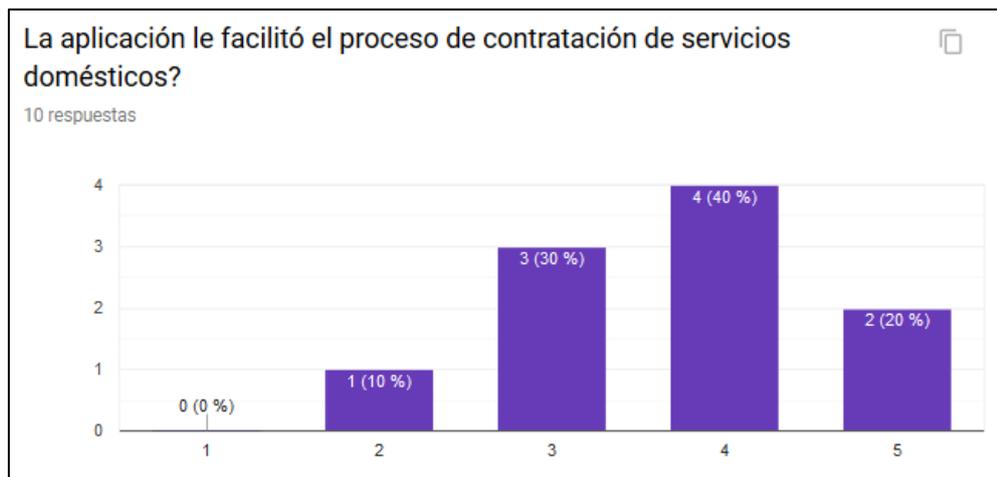


Figura 95. Respuestas de solicitante para pregunta 2 de utilidad percibida.

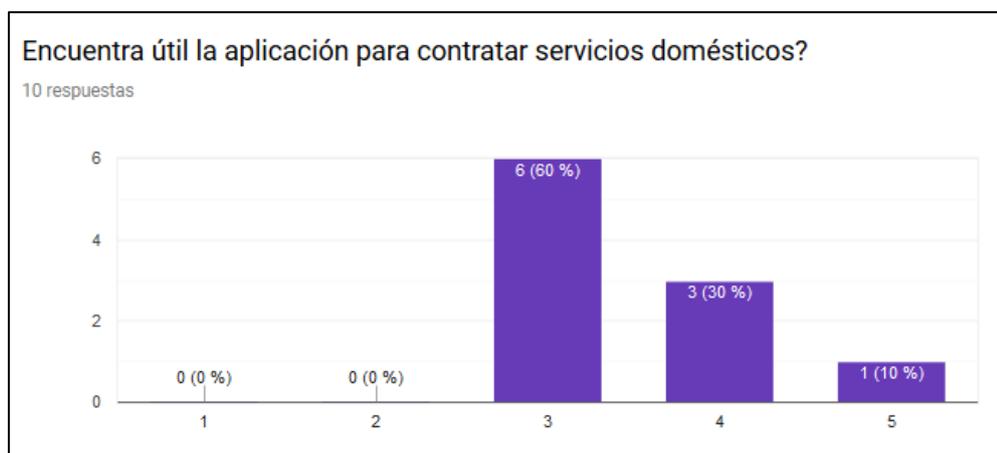


Figura 96. Respuestas de solicitante para pregunta 3 de utilidad percibida.

Facilidad de uso

Ofertante

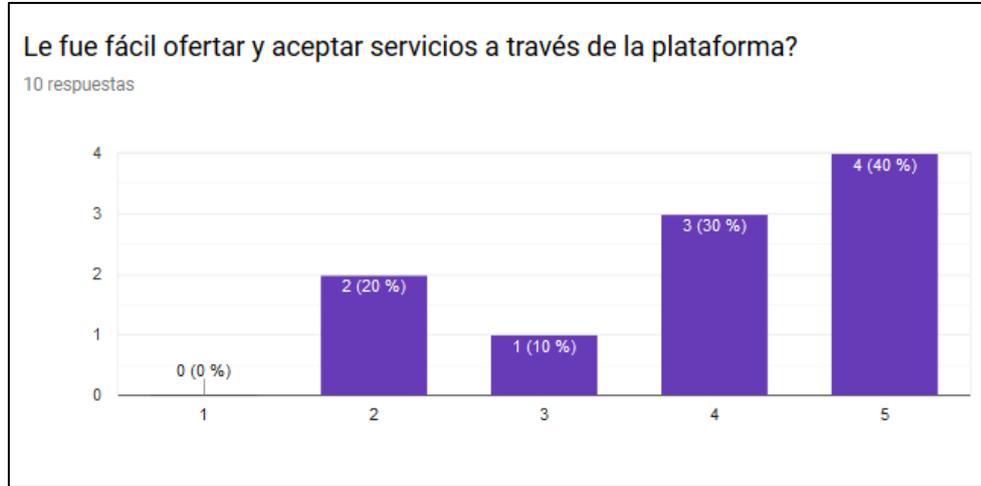


Figura 97. Respuestas de ofertante para pregunta 1 de facilidad de uso.

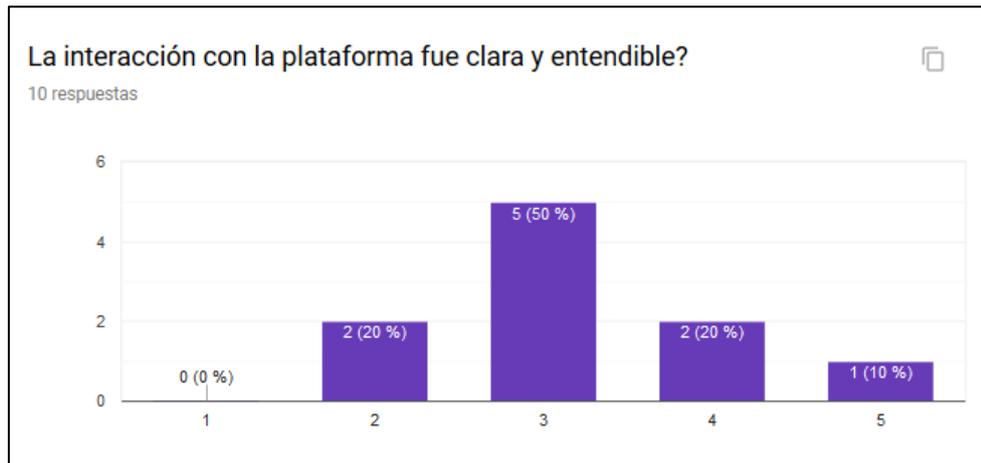


Figura 98. Respuestas de ofertante para pregunta 2 para la utilidad percibida.

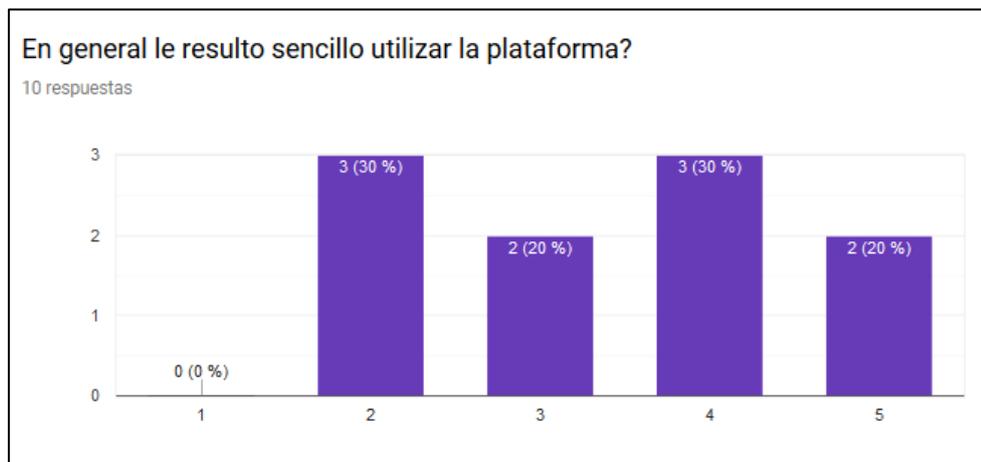


Figura 99. Respuestas de ofertante para pregunta 3 para la utilidad percibida.

Solicitante

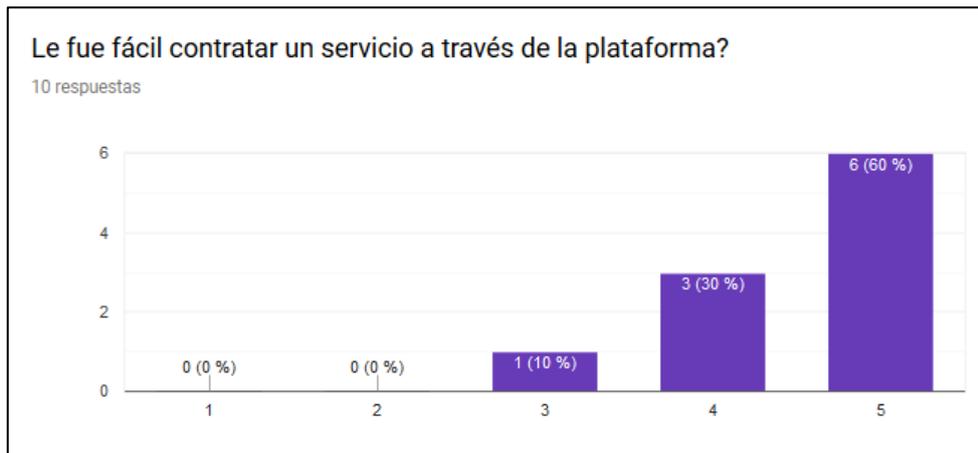


Figura 100. Respuestas de solicitante para pregunta 1 de facilidad de uso.

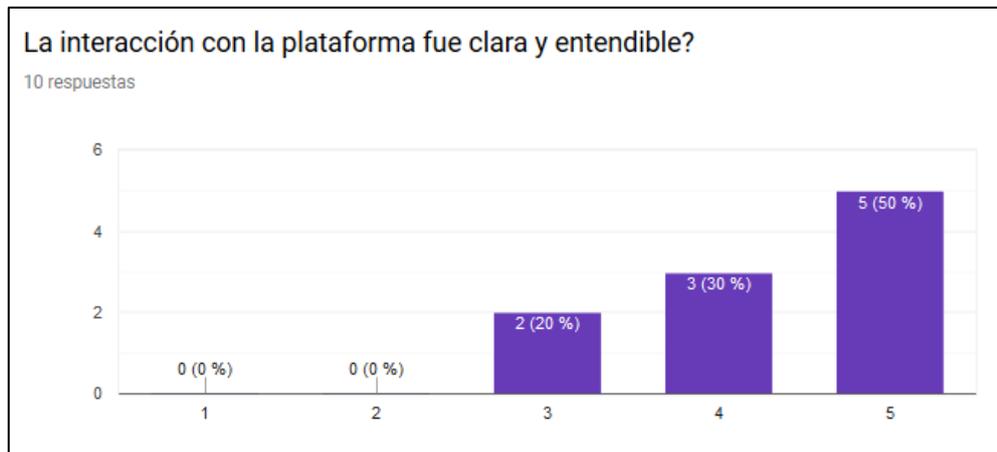


Figura 101. Respuestas de solicitante para pregunta 2 de facilidad de uso.

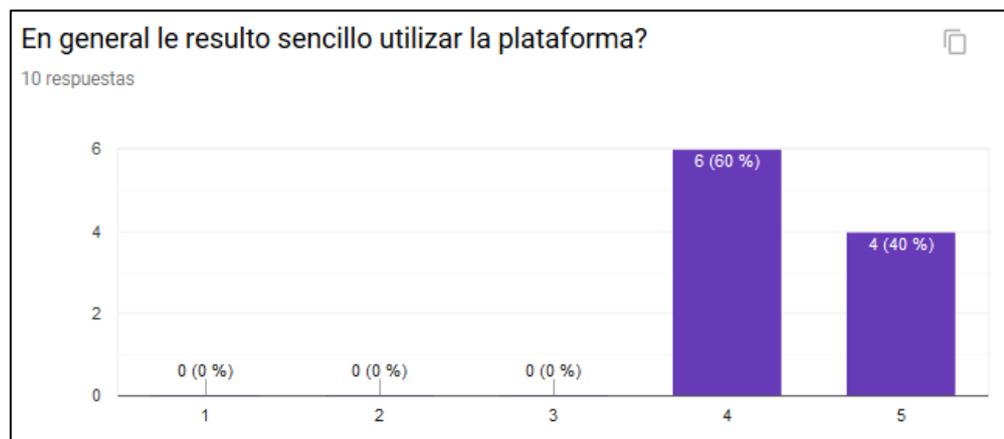


Figura 102. Respuestas de solicitante para pregunta 3 de facilidad de uso.