

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Desarrollo de una aplicación móvil para la gestión del cobro de servicios en una empresa de reclutamiento internacional.

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Ciencias de la Computación

Presentado por:

Carlos Xavier Carvajal Villegas

Freddy Alberto Ponce Muzzio

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

El presente trabajo está dedicado a mis padres Fernando Carvajal, Paola Villegas y Freddy Toala que me enseñaron el valor del esfuerzo, la constancia y dedicación. También a la Ing. Mercedes Cabay que fue un apoyo para mí y formó parte de mi crecimiento.

De igual manera a mis hermanos Doménica Toala y Erick Carvajal y a mi abuela Mary Vera que me han apoyado incondicionalmente durante este tiempo, enseñándome que con esfuerzo y dedicación puedo alcanzar mis metas.

También a un gran amigo y mentor el Ing. Xavier Idrovo, que me enseñó amar la carrera, y a disfrutarla durante todo el proceso y que fue un gran pilar en mi desarrollo como profesional. Finalmente, a mi padawan Gabriela Ramos que fue parte importante de este proceso y mi vida universitaria.

Carlos Xavier Carvajal Villegas

DEDICATORIA

El presente proyecto lo dedico a mis padres Freddy Ponce y Luccetty Muzzio quienes me brindaron ese amor y apoyo incondicional para poder seguir adelante y cumplir mis objetivos. También a mis hermanos Boris, Génesis y Victoria Ponce que fueron una inspiración y un reflejo de lo que puedo llegar a ser, así como una motivación para algún día inspirarlos de la misma forma.

De igual manera a mis abuelos Bolívar Muzzio, Inés Molina y Lucrecia Pinargote quienes me criaron y guiaron desde pequeño, enseñándome buenos valores y brindándome esos valiosos consejos para ser una persona de bien. Son la razón por la cual cada día trato de ser una mejor versión de mí. También dedicárselo a mis tíos(as) y primos(as) que me alentaron a diario en cada una de mis etapas.

Sin ustedes no sería nada, esto se los dedico con mucho amor.

Freddy Ponce Muzzio

AGRADECIMIENTOS

Quisiera empezar agradeciendo a Calvin Sedao, Manuel Suárez y Johana Vásquez de Awana, ya que sin ellos este proyecto no hubiese sido posible.

Por otro lado, agradezco a todos los profesores de Computación especialmente al PhD. Carlos Mera y al MSc. Eduardo Cruz que fueron una inspiración para ser mejor estudiante y profesional. Adicionalmente, al MSc. Rodrigo Saraguro por su paciencia y dedicación en el presente trabajo como en otras etapas de la carrera.

Finalmente, solo me queda decir gracias a todas las personas, compañeros y amigos que hice durante mi vida universitaria, en especial a Karen Segovia, Dayana Huanca, Erick García, Linda Demera, Nicole García, Joel Jordan, Rodrigo Farinango, Camilo Gutiérrez, Freddy Ponce, los TAWS y a Lisbeth Sánchez, por brindarme todo su apoyo y amistad durante este tiempo en ESPOL.

Carlos Xavier Carvajal Villegas

AGRADECIMIENTOS

Quisiera empezar agradeciendo al equipo de Awana conformado por Calvin Sedao, Johana Vásquez y Manuel Suárez que sin ellos, este proyecto no hubiera sido posible.

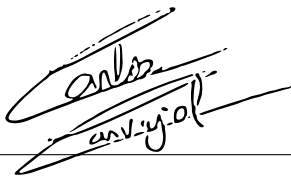
Por otro lado, quiero agradecer al PhD. Carlos Mera y al MSc. Rodrigo Saraguro por ser guías e inspiración a lo largo de mi proceso de formación universitaria, también a mis amigos más cercanos, Carlos Carvajal, Nicole García, Camilo Gutiérrez, Rodrigo Farinango y Gabriela Ramos que me ayudaron muchas veces cuando me sentía estancado, brindándome su amistad y apoyo incondicional.

Finalmente, dar las gracias a todas aquellas personas, compañeros y amigos que formaron parte de mi vida como estudiante, que me impulsaron a crecer y con quienes compartí momentos excepcionales.

Freddy Ponce Muzzio

DECLARACIÓN EXPRESA

”Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Carlos Xavier Carvajal Villegas y Freddy Alberto Ponce Muzzio damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

Handwritten signature of Carlos Carvajal Villegas in black ink, featuring a stylized 'C' and 'V'.

Carlos Carvajal Villegas

Handwritten signature of Freddy Ponce Muzzio in black ink, with a cursive style.

Freddy Ponce Muzzio

EVALUADORES

Erick Vicente Lavid Cedeño
PROFESOR DE LA MATERIA

Rodrigo Alexander Saraguro Bravo
PROFESOR TUTOR

RESUMEN

La gestión de cobros de servicios prestados por empresas internacionales de contratación de TI, al estar relacionado con un número considerable de clientes internacionales que disponen de determinados mecanismos con distintas formas de pago, conlleva por tanto una mayor carga de tiempo y esfuerzo para los encargados de esta tarea. Es por esto que construimos una aplicación móvil multiplataforma para facilitar el cobro de los diferentes servicios que ofrece la empresa para sus clientes. Hicimos uso de API como Stripe para administrar pagos y métodos de pago, también usamos servicios de AWS como Lambda, API Gateway y DynamoDB para crear un backend sin servidor y, finalmente, para crear la interfaz de usuario, usamos React Native con Expo para crear y entregar los MVP. a nuestro cliente Como resultado de este proyecto, nuestro cliente brindará a sus clientes una aplicación móvil donde podrán consultar información relacionada con los servicios que hayan adquirido, también les permitirá agregar múltiples métodos de pago para pagar esos servicios con una suscripción para que el proceso sería automatizado. Como conclusión, este sistema podría ser más eficiente para los encargados de esta tarea, ya que tendrán más tiempo para hacer más cosas relacionadas con su trabajo, haciéndolos más productivos.

Palabras Clave: React Native, AWS, Stripe, Ingeniería de Software, IT.

ABSTRACT

The management of collections of services provided by international IT recruitment companies, being related to a considerable number of international clients who have certain mechanisms with different payment methods, brings therefore a greater burden of time and effort for those in charge of this task. This is the reason why we built a multiplatform mobile application to facilitate the collection of the different services offered by the company for its customers. We made use of APIs as Stripe to manage payments and payment methods, also we use AWS services like Lambda, API Gateway and DynamoDB to create a serverless backend, and finally to create the UI we use React Native with Expo to create and deliver the MVPs to our customer. As a result of this project, our client will give their clients a mobile app where they can look up information related to the services they have acquired, it will also allow them to add multiple payment methods to pay for those services with a subscription so the process would be automated. As a conclusion, this system could be more efficient for those in charge of this task, as they will have more time to do more things related to their work, making them more productive.

Keywords: React Native, AWS, Stripe, Software Engineering, IT.

ÍNDICE GENERAL

RESUMEN	I
ABSTRACT	II
ABREVIATURAS	V
ÍNDICE DE FIGURAS	V
ÍNDICE DE TABLAS.	VII
CAPÍTULO 1	1
1 INTRODUCCIÓN	1
1.1 Descripción del problema	2
1.2 Justificación del problema	2
1.3 Objetivos	3
1.3.1 Objetivo General	3
1.3.2 Objetivos Específicos	3
1.4 Marco teórico	4
1.4.1 Outsourcing	4
1.4.2 Plataforma de pagos	5
1.4.3 Proyectos Relacionados	7
1.4.4 Herramientas Actualmente disponibles	7
CAPÍTULO 2	11
2 METODOLOGÍA	11
2.1 Análisis	11
2.1.1 Requerimientos	12
2.1.2 Backlog	13
2.1.3 Alcance y limitaciones de la solución	14
2.1.4 Riesgos y beneficios de la solución	15
2.1.5 Usuarios de la solución	16

2.2	Prototipado	16
2.2.1	Interfaces generales	16
2.2.2	Administrador	17
2.2.3	Cliente	19
2.3	Evaluación	21
2.4	Diseño de la solución	21
2.4.1	Arquitectura	21
2.4.2	Diagrama de casos de uso	22
2.4.3	Diagrama de clase	23
2.4.4	Diagrama de actividades	25
2.4.5	Plan de cronograma de desarrollo	26
CAPÍTULO 3	27
3 RESULTADOS Y ANÁLISIS	27
3.1	Plan de implementación	27
3.2	Entrega de avances	31
3.3	Plan de Testing	32
3.4	Productos Entregados	33
3.4.1	Prototipo	33
3.4.2	Aplicación Móvil	34
3.5	Análisis de costos	35
3.6	Cierre de Proyecto	38
CAPÍTULO 4	39
4 CONCLUSIONES Y RECOMENDACIONES.	39
4.1	Conclusiones	39
4.2	Recomendaciones	40
BIBLIOGRAFÍA	41
APÉNDICES	42

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
IT	Information Technology
EE. UU	Estados Unidos
CEO	Chief Executive Officer
MIS	Sistema de Información Gerencial
UI	User Interface
API	Application Programming Interface
AOT	Ahead of time
AWS	Amazon Web Services
GCP	Google Cloud Platform
PM	Project Manager
UML	Unified Modeling Language

ÍNDICE DE FIGURAS

Figura 1.1	Roles de IT [Autoría Propia]	5
Figura 2.1	Software ClickUp - Tablero SCRUM [Autoría Propia]	14
Figura 2.2	Interfaz de inicio de sesión [Autoría Propia]	17
Figura 2.3	Interfaz de opciones [Autoría Propia]	17
Figura 2.4	Dashboard del rol de administrador [Autoría Propia]	18
Figura 2.5	Interfaz de notificaciones [Autoría Propia]	18
Figura 2.6	Dashboard del rol de cliente [Autoría Propia]	19
Figura 2.7	Interfaz de métodos de pago [Autoría Propia]	20
Figura 2.8	Interfaz de ayuda e información de la empresa [Autoría Propia]	20
Figura 2.9	Arquitectura de la Solución [Autoría Propia]	22
Figura 2.10	Diagrama de casos de uso [Autoría Propia]	23
Figura 2.11	Diagrama de clases [Autoría Propia]	24
Figura 2.12	Diagrama de Actividades [Autoría Propia]	25
Figura 2.13	Cronograma de desarrollo [Autoría Propia]	26
Figura 3.1	Sprint 1 - Backlog [Autoría Propia]	28
Figura 3.2	Sprint 2 - Backlog [Autoría Propia]	29
Figura 3.3	Sprint 3 - Backlog [Autoría Propia]	29
Figura 3.4	Sprint 4 - Backlog [Autoría Propia]	30
Figura 3.5	Sprint 5 - Backlog [Autoría Propia]	30
Figura 3.6	Agenda de reuniones para presentación de Demos [Autoría Propia]	31
Figura 3.7	Evidencia de la presentación del producto al cliente [Autoría Propia]	31
Figura 3.8	Pruebas Unitarias en el Front-End [Autoría Propia]	32
Figura 3.9	Pruebas Unitarias como parte del CI en el Back-End [Autoría Propia]	33
Figura 3.10	Prototipo de la aplicación en Figma [Autoría Propia]	34
Figura 3.11	Vistas de la Aplicación Móvil [Autoría Propia]	34

Figura 3.12 Estimación mensual de costos por Servicio durante el Desarrollo
[Autoría Propia] 35

Figura 3.13 Costo por servicio [Autoría Propia] 36

ÍNDICE DE TABLAS

Tabla 1.1	Cuadro comparativo de pasarelas de pago [Autoría Propia]	6
Tabla 2.1	Requerimientos funcionales [Autoría Propia]	12
Tabla 2.2	Requerimientos no funcionales [Autoría Propia]	13
Tabla 3.1	Presupuesto para el desarrollo del proyecto [Autoría Propia]	36
Tabla 3.2	Costo estimado de operación anual del proyecto [Autoría Propia]	37

CAPÍTULO 1

1. INTRODUCCIÓN

Las agencias de talento digital que se especializan en la búsqueda, verificación y contratación de ingenieros calificados de todo el continente americano, crean comunidades remotas de innovadores tecnológicos, que apoyan el desarrollo de grandes ideas. El trabajo remoto ha experimentado un auge significativo en los últimos años, y según un reporte de Upwork menciona que, para el 2028, se prevé que el 73% de todos los equipos incluirán empleados remotos, lo cual muestra que el teletrabajo será aceptable en una gama aún más amplia de industrias [1]. Esto ayudó a catapultar soluciones beneficiosas tanto para las empresas de EE. UU. que tienen la oportunidad de conectarse con un grupo nuevo, talentoso y altamente calificado, como para los ingenieros de América Latina con una mayor experiencia, que pueden obtener grandes oportunidades en una industria que antes era inaccesible. Actualmente, este tipo de empresas suelen ofrecer algunos servicios a otras compañías que ya están posicionadas en el mercado o son emergentes, entre los cuales se encuentran: *Talent Placement* que consiste en agregar un miembro valioso al equipo de la empresa, *Contractor* el cual se basa en solicitar los servicios de contratistas latinoamericanos y manejar el pago de las remuneraciones, *Employee* que consiste en agregar empleados de tiempo completo en empresas del extranjero dando la posibilidad de manejar la nómina y otros requisitos del área de recursos humanos, y por último *Team Creation* el cual permite al cliente contratar talentos tecnológicos capacitados específicamente en el conjunto de herramientas y tecnologías que usan además de la cultura de la empresa.

Business Wire estimó que el tamaño del mercado global de subcontratación de procesos de contratación en 2021 sería de 8456,30 millones de dólares [2], dando a entender que la gestión de cobros de servicios provistos por empresas de reclutamiento *IT* internacional han captado el interés de empresas tecnológicas grandes, medianas y emergentes, tanto nacionales como internacionales que disponen de varios mecanismos con diferentes métodos de pago, lo cual tiene como consecuencia una carga de tiempo y esfuerzo mayor para los encargados de dicha tarea.

1.1 Descripción del problema

Las empresas de outsourcing enfocadas al reclutamiento *IT* internacional, se encuentran creciendo cada vez más, por lo que llevar sus procesos contables y finanzas se ha dificultado en los últimos meses debido al gran volumen de transacciones internacionales realizadas manualmente. En algunos casos, se estima que transcurren de 6 a 7 días laborables antes de la fecha de corte hasta que el pago se vea reflejado en las cuentas, generando inconvenientes e inconformidades, como el atraso de cancelación de sueldos, deudas y obligaciones por parte de la empresa.

1.2 Justificación del problema

Business Wire menciona que se proyecta que el mercado global de subcontratación de procesos de contratación alcance los 25379,40 millones de dólares para el 2028, exhibiendo una tasa de crecimiento anual compuesta de 17,00% durante el período de pronóstico [2].

Acorde a las entrevistas realizadas al personal de la compañía y al CEO de la empresa, se observa un malestar por parte de los involucrados, ya que se generan inconvenientes debido al retraso de la cancelación de cobros por los servicios adquiridos.

Esto se debe a que el proceso de cobro involucra una acción por parte de los clientes, los cuales están pendientes de mantener el crecimiento y rentabilidad de sus empresas, generando descuidos. Por otro lado, el hecho de eliminar los diversos problemas que se tienen al momento de realizar transacciones internacionales y locales mediante procesos automatizados para los cobros brinda una menor carga de tiempo y tareas a todas las partes involucradas en el proceso.

Al resolver este inconveniente, se tendrá una mejora en cuanto a optimización de procesos y solventará ciertos factores que intervienen en su realización como poder elegir métodos de pago entre tarjetas de crédito o débito o también una cuenta bancaria de Estados Unidos para así poder suscribirse a pagos recurrentes o únicos acorde a lo establecido con la compañía.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar una aplicación móvil multiplataforma que permita facilitar, visualizar y automatizar los cobros de los diferentes servicios que ofrece la compañía para sus clientes de manera recurrente.

1.3.2 Objetivos Específicos

1. Integrar el servicio de “Stripe” para que las compañías residentes en Estados Unidos puedan realizar el pago de los servicios que mantienen con la empresa.
2. Crear un módulo de notificaciones que comunique al cliente cuando los cobros de los servicios han sido efectuados.
3. Implementar un backend para manejar y almacenar información relacionada a la aplicación móvil, además de automatizar los cobros de servicios de forma mensual, utilizando un método de pago predeterminado.

1.4 Marco teórico

En esta sección se presenta una revisión literaria sobre el outsourcing, haciendo énfasis en el reclutamiento IT, además de ciertas tecnologías para el frontend y backend, utilizadas por la comunidad para implementar aplicaciones móviles multiplataformas. Finalmente, se referencian algunas plataformas y proyectos que permitirán destacar el valor agregado del proyecto.

1.4.1 Outsourcing

En la revisión literaria, podemos encontrar que algunos autores tienen diferentes conceptos, por ejemplo, en el año 2014 Rosero Manosalvas [3] lo definió como la subcontratación de servicios que pretende agilizar y reducir los costos que ocasionan los procesos internos de la empresa. En el año 2021 Anikin y Rudaya [4] lo describen como la situación en la que una empresa delega sus operaciones para que sean gestionadas por un tercero que puede hacerlo más barato, mejor y más rápido.

Daniel Minoli, define el outsourcing de IT como: "Entregar los sistemas de información y/o las funciones de comunicación, en su totalidad o en parte, a un contratista externo como solución al desafío, problema y gasto de crear y administrar una empresa de información corporativa" [5].

Actualmente la mayoría de negocios necesitan un departamento de IT para cumplir los requisitos de gestión de información. Los contratistas de IT realizan todo tipo de trabajo, incluido el desarrollo de software personalizado para sus clientes, tomando datos sin procesar de la empresa y generando todo tipo de informes MIS (Sistema de Información Gerencial). Esto le permite a la empresa reducir costos en software, espacio de servidor, mantenimiento de IT y costos de mano de obra [6].

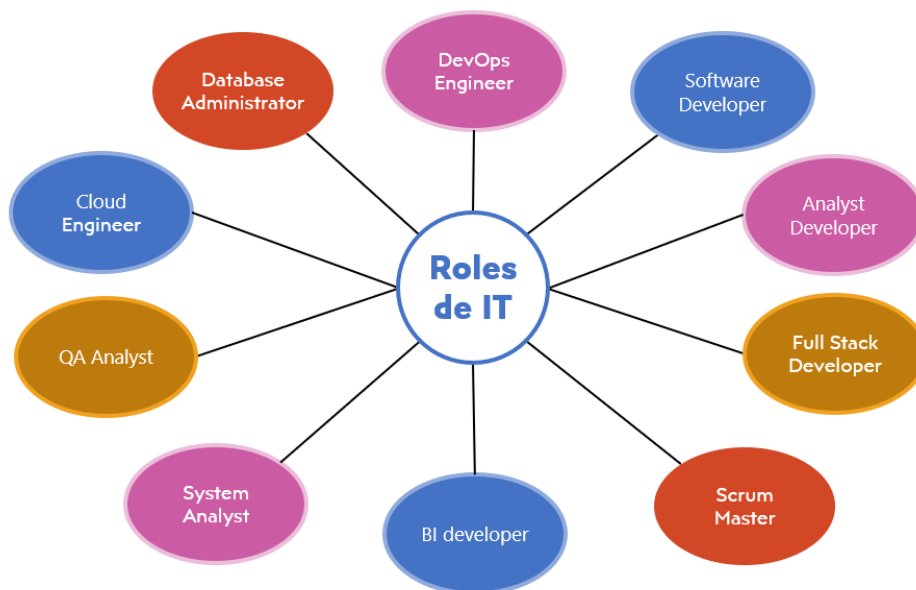


Figura 1.1: Roles de IT [Autoría Propia]

Como se puede observar en la Figura 1.1, son numerosos los roles que existen en IT por lo que el outsourcing permite mejorar los procesos de contratación y optimizar recursos de una empresa, delegando dichas tareas a personas externas capacitadas, permitiéndoles centrarse en funciones y tareas principales del negocio.

1.4.2 Plataforma de pagos

Un sistema para gestionar los cobros es de suma importancia para que un negocio crezca, estos sistemas han madurado en los últimos años dado al avance de tecnología y uso de nuevas herramientas digitales. Con esto nacen las plataformas de pago, que son herramientas que facilitan los pagos no presenciales, conectando el comercio con el cliente mediante internet. Según un estudio de *Americas Market Intelligence* con el título *Los pagos, el comercio y la vida después de la cuarentena de COVID-19* de MasterCard, donde se habla del uso de tarjetas durante la pandemia, se reportó que las tarjetas de crédito y débito se volvieron la forma de pago más utilizadas. El 54% de los encuestados informaron haber cambiado el

método de pago en efectivo por la tarjeta de débito o crédito [7].

Stripe

Stripe es el sistema de pago más reciente en el mercado de las pasarelas de pago, pudiendo conectar tu sitio web y/o aplicación móvil a una cuenta bancaria y empezar a cobrar por tus servicios o productos ofertados.

Además, es una de las API's más potentes y fáciles de usar del mundo, ya que abstrae mucha de la lógica para crear un intento de pago, realizar una transferencia, y agregar métodos de pago, ya que reúne todo lo necesario para desarrollar sitios web y aplicaciones móviles con una infraestructura de pagos para internet [8].

PayPal

Esta plataforma es una solución de pagos con mucha más historia que Stripe, por lo que es una marca reconocida, que aporta mayor confiabilidad a los usuarios que compran o venden por internet.

PayPal se diferencia en que en poco tiempo puedes tener una pasarela de pago segura y confiable, sin necesitar que los compradores tengan una cuenta en la misma, sin embargo, si existen escenarios donde para pagar con PayPal es necesario tener una cuenta registrada en la plataforma [9].

Tabla 1.1: Cuadro comparativo de pasarelas de pago [Autoría Propia]

	PayPal	Stripe
Ventajas	<ul style="list-style-type: none">- Proceso sencillo y rápido.- Es una plataforma ampliamente segura.- Es muy utilizado por los eCommerce.	<ul style="list-style-type: none">- Ofrecen un precio mucho más bajo.- Costos de mantenimiento muy bajos.- Aceptan casi todas las tarjetas bancarias.
Desventajas	<ul style="list-style-type: none">- Se requiere cuenta para poder realizar pagos.- Las comisiones PayPal cobradas son bastante altas.	<ul style="list-style-type: none">- Proceso de integración con eCommerce es un poco complicado.- Necesita conocimiento básico de Stripe.

1.4.3 Proyectos Relacionados

En el Centro de Investigación de Matemáticas, A.C. en el 2012, se llevó a cabo un reporte técnico enfocado al Sistema de Pago para Comercio Electrónico en donde se buscaba integrar un modelo de negocio con un cliente, para solventar la preocupación de los métodos de pago quitando la necesidad de tener una tarjeta de crédito, esto debido al auge del comercio electrónico y el alcance que tiene, ya que abarca el mercado tanto local, nacional como internacional. En primera instancia el alcance del proyecto es a nivel nacional, sin embargo, cuenta con una proyección a la integración con otros servicios como PayPal [10].

Este tipo de proyectos que están enfocados a los servicios de pagos internacionales suelen ser escasos, sin embargo, existen plataformas que solventan ciertas necesidades transaccionales como PayPal, que resulta ser un servicio que permite realizar pagos de forma segura a través de tu dispositivo móvil [11].

Sin embargo, estos proyectos o servicios no abarcan todas las necesidades que tiene una empresa de reclutamiento IT con clientes internacionales para cubrir la gestión de cobros con diferentes métodos de pago como tarjeta o cuenta bancaria de EE. UU., ya que involucra otras etapas, requerimientos y procesos.

1.4.4 Herramientas Actualmente disponibles

Por un lado, actualmente las tecnologías que se utilizan para el desarrollo de aplicaciones móviles multiplataformas a nivel de frontend son React Native y Flutter, siendo "React Native" uno de los frameworks más populares entre aquellos que tienen experiencia creando aplicaciones o páginas webs, ya que cuentan con experiencia en JavaScript. Mientras que Flutter, utiliza el lenguaje de programación Dart, haciéndolo menos atractivo dado que los lleva a tener que aprender un nuevo lenguaje. Sin

embargo, ambos comparten la filosofía de *Aprende una vez, escribe en cualquier lugar*, haciendo énfasis que con estas herramientas podemos crear aplicaciones para la web, PC's, celulares inteligentes y tablets [12] [13].

Por otro lado, las tecnologías que son más conocidas y utilizadas para el desarrollo de aplicaciones de backend en la actualidad son Amazon Web Services (AWS), Google Cloud Platform (GCP) y Microsoft Azure, las cuales son una colección de servicios de computación en la nube que nos permiten construir, probar, desplegar y administrar aplicaciones o servicios [14] [15] [16].

React Native

Lanzado en el año 2015 por Facebook como un framework UI de código abierto, con el objetivo de crear aplicaciones para cualquier plataforma permitiendo que los desarrolladores usen React junto con sus capacidades nativas.

React Native permite a los desarrolladores crear aplicaciones móviles híbridas, es decir que genera representaciones nativas para el sistema operativo iOS y Android [12], por lo que utilizarlo para el desarrollo de aplicaciones móviles nos asegura la reutilización de código y hasta un 70% de compatibilidad entre iOS y Android [17], además de contar con un Hot Reload para compilar nuestro código de manera rápida al momento de desarrollarlo.

Para el desarrollo de la interfaz de usuario, React Native utiliza componentes, que son porciones de código que describen cierta parte visual de la aplicación como botones, tarjetas, menús, etc., dado al tiempo que lleva en el mercado y la comunidad que tiene, podemos contar con una gran variedad de librerías con componentes específicos para cada plataforma, con su correspondiente animación minimizando así el tiempo de desarrollo [18].

Flutter

Lanzado por Google en el año 2017 como un framework UI creado con el lenguaje de programación Dart. Dado que ambos fueron desarrollados por Google esto asegura la compatibilidad entre ellos. Gracias a Dart, Flutter logró ser Ahead of time (AOT) [19], permitiendo la compilación a código nativo, esto elimina ciertas etapas innecesarias para poder obtener una aplicación en diferentes plataformas móviles. Por otro lado, Flutter contiene ciertos widgets que facilitan la implementación de la interfaz de usuario, ya que son clases que describen ciertos componentes generales de una aplicación como botones, imágenes, tablas, etc.]

Amazon Web Services

AWS provee una gran variedad de servicios entre los cuales tenemos, DynamoDB, Lambda y API Gateway que son las idóneas al momento de implementar una infraestructura debido a la abstracción, ya que el servicio de Lambdas te permite crear las funciones para tareas específicas, las cuales usualmente se relacionan con la gestión de las bases de datos como DynamoDB, dichas funciones son manejadas a través de la puerta de enlace representado por el servicio de API Gateway, teniendo como finalidad una API RESTful.

DynamoDB

Es un servicio de base de datos NoSQL provisto por Amazon que ofrece una perfecta escalabilidad y rendimiento logrando reducir la cantidad de tareas, ya que no disponemos de un centro de datos. Además de que provee una alta disponibilidad y durabilidad, gracias a que todos los datos almacenados se replican de forma automática en varias zonas disponibles de una región de AWS [20].

Lambda

Es un servicio informático en la nube que permite la ejecución de código en una infraestructura de alta disponibilidad sin la necesidad de tener servidores. Con Lambda, se puede ejecutar código para cualquier tipo de aplicación o servicio de backend, mediante la carga de una función escrita en uno de los lenguajes permitidos por el servicio [21].

API Gateway

Este servicio nos ayuda en la creación, publicación, mantenimiento, monitoreo y protección de las API REST, HTTP y de WebSocket. Suele trabajar en conjunto con Lambda para formar una infraestructura conocida como *serverless* dado que se utilizan las lambdas para ejecutar funcionalidades, además de otros servicios necesarios para la aplicación, las cuales son expuestas a través de los métodos de API que ofrece API Gateway [22].

Con estas premisas identificamos la falta de soluciones computacionales para realizar cobros internacionales en temas de servicios, esto nos lleva a proponer un aplicativo móvil que utilice las herramientas actualmente más utilizadas por parte de la comunidad de desarrolladores, como es el caso de React Native junto a servicios de AWS que hacen que la aplicación sea robusta. Es crucial tener una arquitectura flexible, es decir que permita integrar nuevas tecnologías o nuevos servicios a la misma, el resultado que se espera es que las empresas de reclutamiento de IT internacional aprovechen la solución que trae la facilidad y optimización de poder gestionar el cobro de servicios.

CAPÍTULO 2

2. METODOLOGÍA

La gestión y desarrollo de software ha estado en constante evolución, así como las metodologías ágiles que se caracterizan por la entrega continua de valor al cliente, la velocidad de desarrollo y la capacidad a responder mejor a las tendencias o cambios del mercado [23].

Según lo expuesto, en el presente proyecto se utilizó la metodología SCRUM, la cual se basa en *sprints* que son pequeñas iteraciones cortas de 1 o 2 semanas máximo, en los cuales se trabaja en una lista de pendientes o tareas a completar denominada *backlog* del sprint. En esta metodología, existen roles importantes como el "product owner" que es el encargado de conocer el producto y se centran en entender los requisitos empresariales de los clientes y del mercado para luego priorizar el trabajo que el equipo de desarrolladores debe cumplir [24].

El objetivo de nuestro proyecto es desarrollar una aplicación móvil multiplataforma, que automatice el proceso de cobros de servicios a una empresa de reclutamiento IT, reduciendo así el tiempo y la carga que conlleva cumplir dicha tarea. Este software requiere de características escalables y la capacidad de responder a las tendencias o cambios del mercado, además de la integración continua.

2.1 Análisis

Es necesario conocer las necesidades del cliente y el enfoque que desean tener para poder obtener un producto final que sea de su agrado, por lo que, a través de procesos iterativos, se utilizaron diferentes herramientas como entrevistas semi-guiadas e historias de usuario a clientes y trabajadores de la compañía para la elicitación de requisitos, obteniendo información valiosa

que nos permitió establecer los requerimientos funcionales y no funcionales finales, los roles de los usuarios finales como lo son **Company**, el cual se refiere a los clientes relacionados con la empresa de reclutamiento IT y el rol de **Admin** que engloba al área de recursos humanos. Como resultado se definió que la solución a desarrollar sea una aplicación móvil multiplataforma con diseño responsivo de tal forma que sea adaptable a diferentes modelos de dispositivos, que cumpla con los requerimientos descritos en la siguiente sección.

2.1.1 Requerimientos

Tabla 2.1: Requerimientos funcionales [Autoría Propia]

ID	Requerimiento
AP-01	Como Company, yo quiero realizar transferencias o pagar con mi tarjeta de débito o crédito desde Estados Unidos para hacer efectiva la paga de mis servicios que mantengo con Awana.
AP-02	Como Company, yo quiero poder agregar varios métodos de pago para poder pagar los servicios adquiridos con Awana.
AP-03	Como Company, yo quiero poder editar el método de pago predefinido y seleccionar otro para poder pagar los servicios adquiridos con Awana.
AP-04	Como Company, yo quiero tener la posibilidad de eliminar la información bancaria registrada como métodos de pago que no use para poder tener solamente los métodos de pago que servirán en pagos posteriores.
AP-05	Como Company, yo quiero recibir notificaciones de los pagos realizados por correo o push para tener un mejor control del valor cancelado.
AP-06	Como Company, yo quiero recibir notificaciones previo a los vencimientos de pagos por correo o push para tener un tipo de recordatorio de que debo cancelar mis valores pendientes.
AP-07	Como Company, yo quiero visualizar la información pertinente de los trabajadores para visualizar fácilmente información sobre ellos.
AP-08	Como Company, yo quiero visualizar el historial de pagos realizados, y los próximos a vencer, especificado por los servicios que mantengo con Awana para llevar un mejor control de los mismos.

AP-09	Como Company, yo quiero que la factura que se me envíe al momento del cobro tenga descripción del pago a realizar para tener una mejor idea de qué estoy pagando en el mes.
AP-10	Como usuario final, yo quiero elegir mi idioma nativo para usar la aplicación.

Tabla 2.2: Requerimientos no funcionales [Autoría Propia]

ID	Requerimiento
APNF-01	El sistema debe ser accedido únicamente por usuarios autorizados.
APNF-02	La aplicación debe estar integrada con los servicios de auth0 que maneja la empresa para añadir una capa de autorización a cada rol y seguridad a la aplicación.
APNF-03	La aplicación móvil debe operar tanto en iOS como en Android.
APNF-04	El sistema deberá ser escalable por lo que utilizará servicios en la nube de AWS que ofrece la empresa.
APNF-05	El sistema deberá integrar Stripe como servicio de pago para el cobro de servicios en los Estados Unidos.
APNF-06	La aplicación móvil debe desarrollarse con el framework React Native que pertenece al stack que maneja la empresa.
APNF-07	El sistema debe cumplir los requerimientos en la ley de protección de datos.

A partir de estos requerimientos logramos definir las tareas que conforman nuestro backlog. En el Apéndice C se agregaron los requerimientos mencionados de forma más detallada teniendo en cuenta los criterios de aceptación.

2.1.2 Backlog

Basándonos en las entrevistas realizadas a los diferentes involucrados en el problema de la compañía, se pudo evidenciar las necesidades que permiten tomar acciones en lo que respecta a la solución propuesta. Por tal razón, creamos un backlog Figura 2.1 utilizando ClickUp el cual es un software que nos permite organizar y asignar tareas a los miembros del equipo en un tablero SCRUM, de tal manera que se puedan actualizar el estado de la tarea dentro del tablero a medida que se vayan realizando.

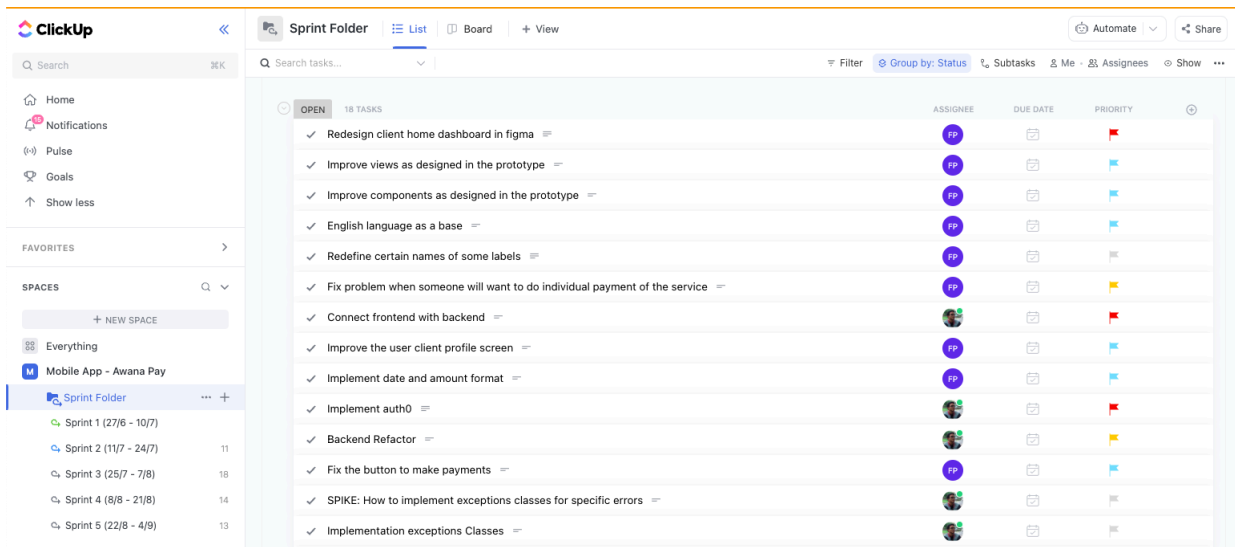


Figura 2.1: Software ClickUp - Tablero SCRUM [Autoría Propia]

Las tareas del backlog fueron organizadas por sprints de 2 semanas, con el objetivo de tener retroalimentación de las actividades que realiza el equipo y avance del producto, además de realizar cambios de ser necesario. Así mismo, esto nos permite planificar de forma adecuada las tareas que puedan causar bloqueos entre sí, como es el caso de las tareas de integraciones con servicios o con el backend.

2.1.3 Alcance y limitaciones de la solución

La aplicación móvil que se va a desarrollar tiene como finalidad agilizar el proceso de cobro, por lo que se debe tener en cuenta que las transacciones serán internacionales teniendo 2 puntos claros que son el manejo y la protección de datos, ya que se va a utilizar información sensible debido a que se manejarán cobros a través servicios externos y en cuanto a la experiencia de usuario implica que debe mostrar una adaptabilidad para los 2 principales idiomas inglés y español.

El sistema se encargará de proveer la información necesaria del valor que la empresa relacionada debe cancelar a modo de factura, además de simplificar el proceso mediante una suscripción la cual será un pago activo de forma periódica o única teniendo un método

de pago por defecto que será a elección del usuario junto con la posibilidad de agregar nuevos métodos. Esto implica un módulo de notificaciones para informar al usuario sobre la realización del pago.

2.1.4 Riesgos y beneficios de la solución

Riesgos

Al momento de implementar un sistema que reemplazará un proceso manual, suelen haber ciertos riesgos por parte de los usuarios como el período de adaptación y que los trabajadores se sientan amenazados por ser reemplazados. Entre los riesgos identificados en el desarrollo de la propuesta podemos observar también la dependencia de servicios externos como es el caso de AWS, que actualmente es uno de los mejores proveedores de servicios en la nube en el mercado. En el caso de que algún servicio utilizado en la aplicación deje de tener soporte, nos obligaría a realizar una migración de proveedor no contemplada. Por otra parte, los bugs o errores de API's de terceros como Stripe que es una robusta infraestructura de pagos para Internet en caso de que una de sus funciones falle o exista una interrupción, el usuario obtendrá retroalimentación mediante la aplicación.

Beneficios

La implementación de nuestra aplicación móvil conlleva grandes beneficios, no solo para la compañía, sino para sus clientes, como la disponibilidad en cualquier dispositivo iOS y Android, además de que el usuario estará 100% informado sobre los procesos de cobro con Stripe. Asimismo, el hecho de contar con un servicio de autenticación como Auth0, brinda una capa extra de seguridad y una mayor confiabilidad de que sus datos estarán seguros. Por último, una vez el sistema haya sido acogido por la empresa, se obtendrá una mejora en la ejecución del proceso de cobro, la disminución del tiempo invertido en dicha tarea, aumento en la productividad por parte del equipo y la disminución

de la probabilidad de inconvenientes ocasionados por retrasos.

2.1.5 Usuarios de la solución

Administrador

Encargados de la gestión de cobros, representan al personal de Recursos humanos de la empresa de outsourcing que provee los servicios. Podrán revisar la información de los clientes, empleados además de ver las facturas y el registro de pagos.

Cliente

Encargados de realizar la cancelación del cobro, quienes podrán tener una suscripción para realizarlo periódicamente además de tener la posibilidad de añadir diferentes métodos de pago ya sea mediante tarjeta (crédito o débito) o transferencia bancaria.

2.2 Prototipado

En base a los requerimientos funcionales obtenidos durante el proceso de elicitación, se diseñó un prototipo del producto que se desea obtener, a través del uso de la herramienta *Figma*, la cual permitió plasmar nuestras ideas de forma clara, ya que ofrece la posibilidad de simular el flujo de interacción del usuario.

2.2.1 Interfaces generales

Con el objetivo de cumplir ciertos requerimientos, se diseñaron algunas interfaces para solventar dichas necesidades tales como la interfaz de inicio de sesión de la Figura 2.2, la cual cubre la protección de datos y por otro lado tenemos la interfaz de opciones como se muestra en la Figura 2.3 que permite al usuario cambiar el idioma, consultar su información personal y entre otras.

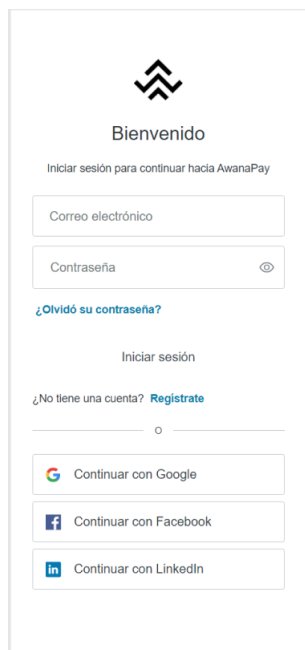


Figura 2.2: Interfaz de inicio de sesión [Autoría Propia]

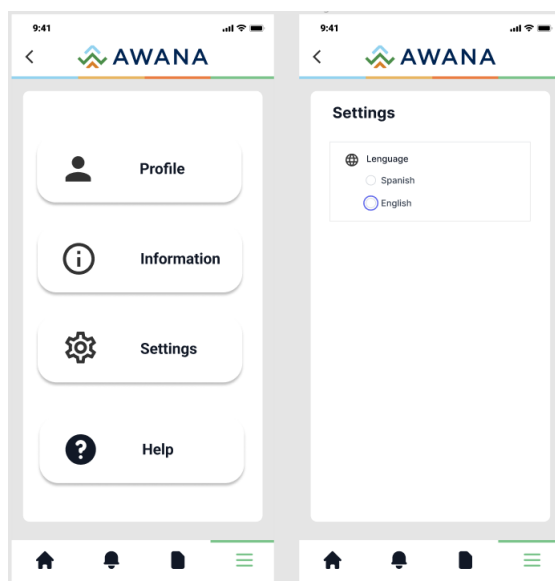


Figura 2.3: Interfaz de opciones [Autoría Propia]

2.2.2 Administrador

Para el rol de administrador, según las historias de usuarios recopiladas se diseñaron interfaces para satisfacer las especificaciones mencionadas de tal forma que le permita visualizar la lista de los clientes que posee la empresa junto con la nómina de empleados con los cuales están asociados como se

aprecia en la Figura 2.4, y por último en la Figura 2.5 se muestra el módulo de notificaciones para comprobación de cobros recibidos y realizados.

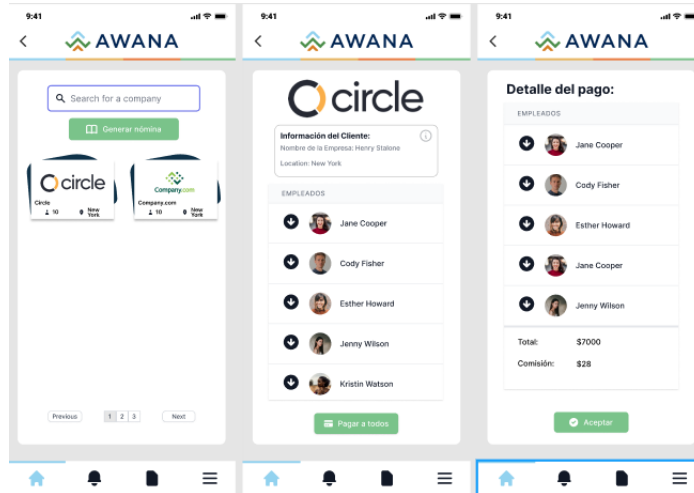


Figura 2.4: Dashboard del rol de administrador [Autoría Propia]

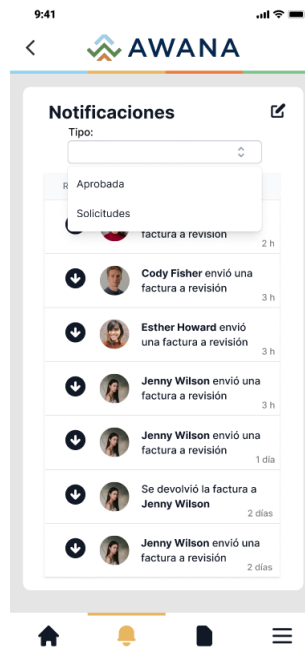


Figura 2.5: Interfaz de notificaciones [Autoría Propia]

2.2.3 Cliente

Por último, tenemos el rol de cliente, en donde podrá visualizar una interfaz principal que le dará acceso a todas las funcionalidades de la aplicación como tenemos en la Figura 2.6, como por ejemplo la posibilidad de agregar un método de pago cuyo flujo se especifica en la Figura 2.7, además de poder eliminar o editar dicho método. Y como última función específica, se presenta un formulario de ayuda como se indica en la Figura 2.8, para informar cualquier inconveniente ocurrido, lo que generará un correo que será recibido por la empresa.

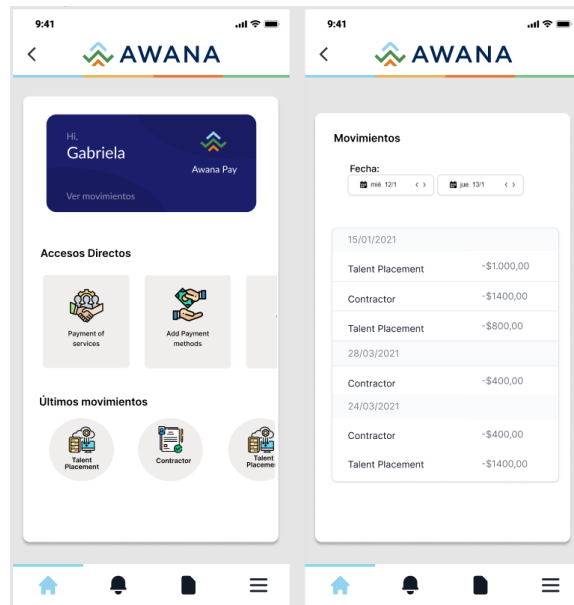


Figura 2.6: Dashboard del rol de cliente [Autoría Propia]

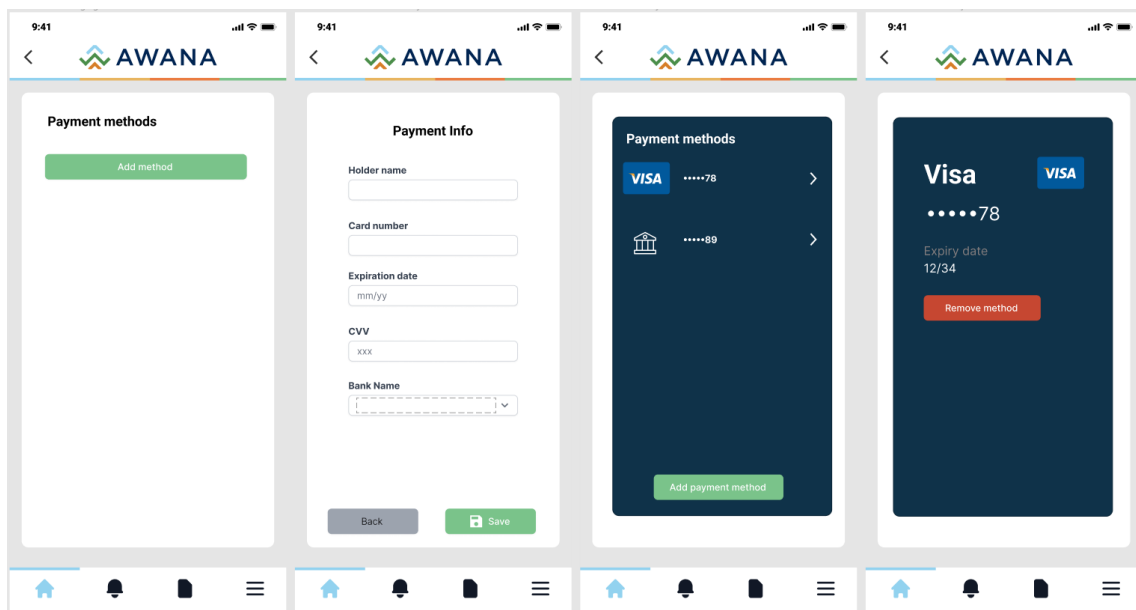


Figura 2.7: Interfaz de métodos de pago [Autoría Propia]

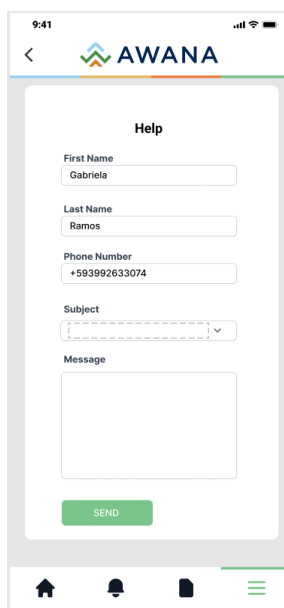


Figura 2.8: Interfaz de ayuda e información de la empresa [Autoría Propia]

2.3 Evaluación

Luego de la primera presentación del prototipo del sistema, se obtuvieron observaciones positivas por parte de los stakeholders y la Product Manager (PM) quien emitió una carta de aceptación del prototipo (ver Apéndice D) y requerimientos de la aplicación en nombre de la empresa, indicando el cumplimiento de cada uno de los requisitos solicitados y un diseño intuitivo, ya que la mayoría de las funcionalidades, están al alcance desde la vista principal. Por otro lado, también resaltaron ciertos cambios en cuanto al diseño del dashboard del cliente, ya que se presentaban ciertas opciones secundarias que aún no estaban del todo claras y podrían perjudicar a la experiencia de usuario, teniendo en cuenta que la funcionalidad principal de nuestra solución es la posibilidad de cobrar servicios por medio de suscripciones utilizando diferentes métodos de pagos.

2.4 Diseño de la solución

En base a las reuniones con los stakeholders junto a la PM y a los requerimientos establecidos, se obtuvieron ciertos criterios para poder llegar a definir puntos sustanciales para poder plasmar las especificaciones antes mencionadas, de tal forma que se consiguió un panorama más claro de la solución, teniendo en cuenta diversas variables y servicios para poder desarrollar un sistema acorde a lo solicitado, las cuales serán mencionadas a detalle a continuación.

2.4.1 Arquitectura

En el diseño de la aplicación se optó por una arquitectura que brinde flexibilidad, escalabilidad y disponibilidad de servicios con altos estándares. Como se puede apreciar en la Figura 2.9, se optó por una arquitectura serverless, el cual utiliza servicios de Auth0 para manejar la autenticación de usuarios, y de AWS para manejar

la lógica de la aplicación y el almacenamiento de datos, por último, la API de Stripe que permite gestionar las actividades comerciales en Internet.

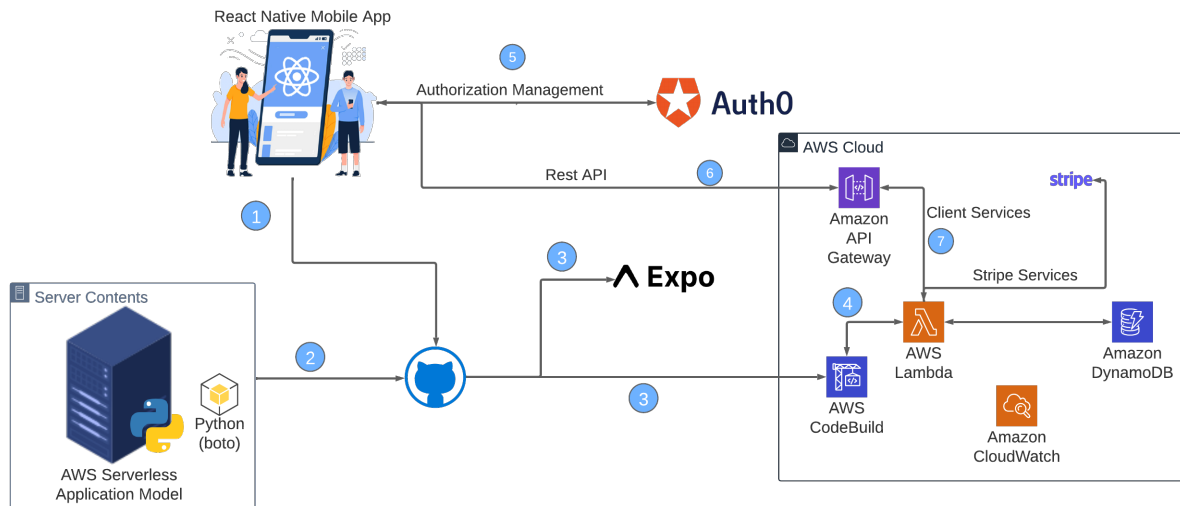


Figura 2.9: Arquitectura de la Solución [Autoría Propia]

2.4.2 Diagrama de casos de uso

En la Figura 2.10 se muestra los comportamientos que pueden tomar 3 actores del sistema como lo son: el administrador, el cual se encarga de la gestión del cobro como lo es crear la factura dependiendo del cliente que haya contratado algún(os) servicio(s), por otro lado tenemos al cliente cuyo rol es del receptor del cobro por lo que el principal uso que le daría al sistema, sería el de proveer la información necesaria para poder realizar la transacción que cubra el valor a pagar y por último tenemos el servicio externo llamado *Stripe* que estará integrado a la aplicación para realizar las transacciones internacionales.

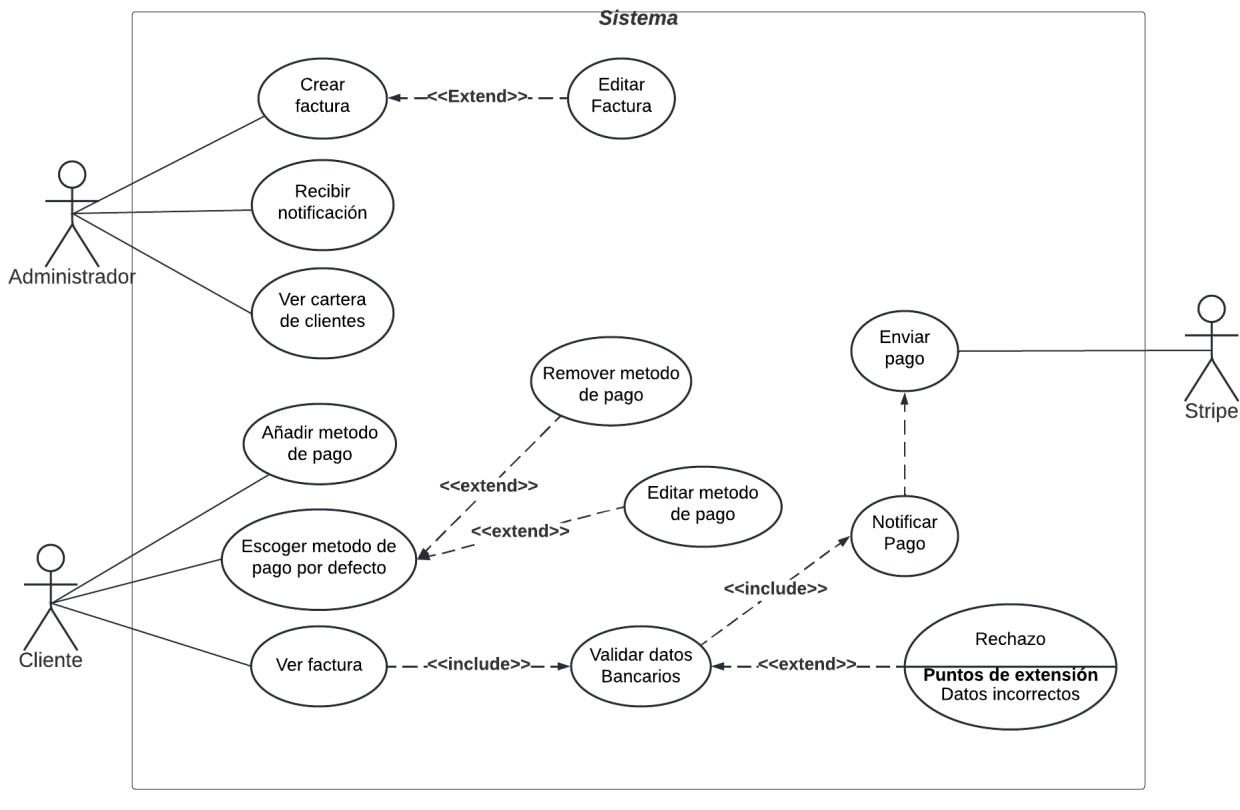


Figura 2.10: Diagrama de casos de uso [Autoría Propia]

2.4.3 Diagrama de clase

Para definir el comportamiento de la aplicación es necesario determinar el modelo de las entidades que participan en el proceso. Como se puede observar en la Figura 2.11 en la cual se define las clases que intervienen en el proceso actual del negocio y que son necesarias para realizar un cobro dentro del contexto de una empresa IT internacional usando un servicio de cobro como Stripe. Mediante el uso de las herramientas de UML logramos especificar las clases con sus atributos y métodos, además del comportamiento entre ellas lo cual ayudó a crear nuestro backlog.

2.4.4 Diagrama de actividades

El diagrama de la Figura 2.12 muestra el flujo entre el administrador, el servicio de *Stripe* y el cliente, donde se puede observar que el cliente añade un método de pago, luego Stripe se encarga de validar los datos ingresados para poder realizar las transacciones de forma periódica mediante una suscripción, para que finalmente el administrador sea notificado sobre cancelación del cobro a través de una notificación.

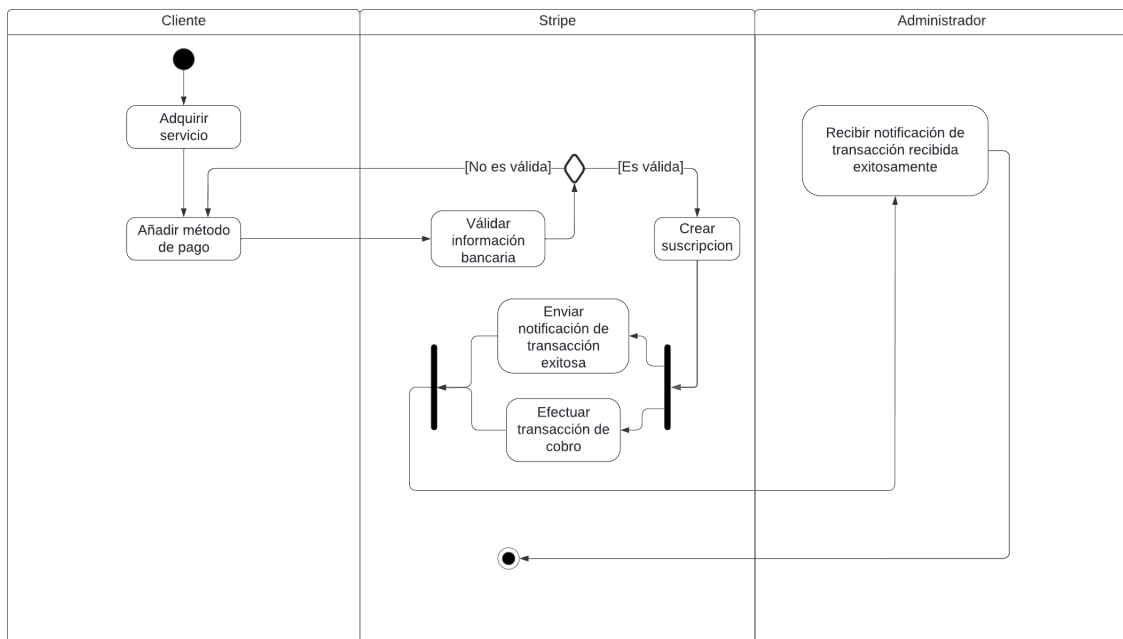


Figura 2.12: Diagrama de Actividades [Autoría Propia]

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

En el presente capítulo se describe la implementación de la solución basada en la metodología SCRUM, el análisis de costos de los servicios en la nube y las capacidades de las herramientas utilizadas, así mismo se evaluará las opciones de mejora que podrían agregarse a la aplicación para generar mayor valor y adopción efectiva.

3.1 Plan de implementación

La implementación de la solución se llevó a cabo de acuerdo con el marco de trabajo SCRUM, en la cual se logró identificar 3 etapas que se detallan a continuación:

- La primera etapa fue el levantamiento de información y requerimientos con el cliente, los stakeholders y demás afectados de la problemática; durante la misma se documentó las necesidades y problemas que habían al momento de realizar cobros a compañías extranjeras, así mismo mediante el uso de entrevistas semi-guiadas se identificaron los roles y actividades claves, los puntos de dolor de los usuarios y stakeholders. A partir de este análisis se realizó la definición de los requerimientos funcionales y no funcionales para que la solución propuesta pueda solventar dichas necesidades y genere valor al usuario final.
- La segunda etapa fue de prototipado, en esta se utilizó la herramienta de Figma para plasmar a algo tangible las ideas del producto final, cumpliendo los requerimientos funcionales y no funcionales. El

objetivo de esta fase fue obtener una retroalimentación de manera rápida e iterativa a través del usuario, aprender de él ajustándose cada vez más a la satisfacción de sus necesidades y tener un diseño final para generar un menor coste y tiempo de implementación.

- Una vez que fueron validados tanto los requerimientos como el prototipo se llevó a cabo la tercera y última etapa, la cual consistió en convertir el prototipo en una aplicación real, por lo que dentro del marco metodológico se definieron y asignaron las tareas en el backlog de cada sprint con sus respectivas fechas estimadas de entrega. La herramienta para llevar esto a cabo fue ClickUp, como se indican en las Figura 3.1 a 3.5, ya que ayuda a tener un seguimiento del avance de cada tarea y que haya una mejor comunicación con el cliente.

TAREA	PERSONA ASIGNADA	FECHA DE VENCIMIENTO	PRIORIDAD	TIEMPO ESTIMADO
Define Enum invoice statuses	FP	jun. 29	Normal	1h 3m
Add and/or update attributes to existing resources in the db	FP	jul. 3	Alta	2h 6m
Create the Invoice resource	FP	jun. 28	Alta	2h 6m
Define a date for payments to be regular each month.	FP	jun. 29	Normal	2h 6m
Generic button layout	FP	jul. 7	Baja	2h 6m
Window Development: Screen Structure	FP	jul. 4	Alta	2h 6m
Create the Service-Invoice resource	FP	jun. 29	Baja	1h 3m
Define the maximum collection time to determine that the payment is regular.	FP	jun. 29	Normal	2h 6m
Create the Tax resource	FP	jun. 29	Alta	1h 3m
Create the Service resource	FP	jun. 30	Alta	1h 3m
Create the Social Benefit resource	FP	jun. 30	Alta	1h 3m
Development of the customer card component	FP	jul. 5	Baja	2h 6m
Spike: Translations from English to Spanish and vice versa in the application and mockup	FP	jul. 3	Alta	4h 18m

Figura 3.1: Sprint 1 - Backlog [Autoría Propia]

Sprint 2 (11/7 - 24/7) + NUEVA TAREA OCULTAR CERRADOS

jul. 11 jul. 24

TAREAS	PERSONA ASIGNADA	FECHA DE VENCIMIENTO	PRIORIDAD	TIEMPO ESTIMA...
Development of the credit/debit card component	FP	jul. 20	🟢	2h 6m
Search bar component development	FP	jul. 21	🟢	2h 6m
Implementation of HR dashboard	FP	jul. 13	🔴	2h 6m
Define the names of services that can be contracted	👤	jul. 11	🟢	1h 30m
Obtain basic information on services that can be contracted	👤	jul. 11	🟢	1h 30m
Define the prices of services that can be contracted and their %	FP	jul. 12	🟢	1h
Create the service resource in Dynamo	👤	jul. 13	🔴	1h
Create a transaction module	👤	jul. 15	🟢	1h
Make a transaction within the USA using Stripe	👤	jul. 18	🔴	3h 30m
Spike of tables in react (React table or some other library, etc)	FP	jul. 15	🔴	4h 30m
Implementation of client dashboard	FP	Hoy	🔴	5h
UI design of the invoice	👤	jul. 21	🟢	3h
Development of employee display tables by company	FP	Ayer	🔴	4h

Figura 3.2: Sprint 2 - Backlog [Autoría Propia]

Sprint 3 (25/7 - 7/8) + NUEVA TAREA OCULTAR CERRADOS

jul. 25 ago. 7

TAREAS	PERSONA ASIGNADA	FECHA DE VENCIMIENTO	PRIORIDAD	TIEMPO ESTIMADO
Redesign client home dashboard in figma due to iterations	FP	jul. 25	🔴	10h 30m
Improve views as designed in the prototype	FP	jul. 25	🟢	10h 30m
Improve components as designed in the prototype	FP	jul. 27	🟢	6h 18m
English language as a base	FP	Hace 6 días	🟢	1h 3m
Connect frontend with backend	👤	jul. 25	🔴	24h
Improve the user client profile screen	FP	Hace 5 días	🟢	6h 18m
Implement date and amount format	FP	Hace 3 días	🟢	1h 3m
Implement auth0 integration	👤	Hace 4 días	🔴	8h
Backend improvements	👤	jul. 27	🟡	8h
Fix the button to make payments	FP	Hoy	🟢	6h 18m
SPIKE: How to implement exceptions classes for specific errors	👤	Hace 4 días	🟡	2h
Implementation exceptions Classes	👤	Ayer	🟡	2h
Add payment settings for all services considering commissions	👤	Hoy	🟢	6h 18m
SPIKE: Investigate if there is a way to do payments through bank debit.	👤	dom.	🟡	6h 18m
Implement stripe payments with bank debit	👤	sáb.	🔴	6h
Implement stripe payments with credit card payment	FP	sáb.	🟢	5h

Figura 3.3: Sprint 3 - Backlog [Autoría Propia]

Sprint 4 (8/8 - 21/8) + NUEVA TAREA ✓ MOSTRAR CERRADOS

ago. 8 - ago. 21

TODO 13 TAREAS

TAREA	PERSONA ASIGNADA	FECHA DE VENCIMIENTO	PRIORIDAD	TIEMPO ESTIMADO
Add new field into user database table	[Avatar]	lun.	Alta	6h 18m
Add functionalities to notification screen	[Avatar]	lun.	Alta	16h 48m
Visually implement the methods of payment that can be made	[Avatar]	ago. 11	Alta	15h 20m
Stripe integration with backend	[Avatar]	mar.	Alta	24h 35m
Design and implement the help screen	[Avatar]	ago. 13	Alta	10h 43m
Design and implement the information screen	[Avatar]	ago. 15	Alta	10h 30m
Implement functionality to enter bank information	[Avatar]	ago. 15	Alta	8h 25m
Make frontend and backend integration	[Avatar]	ago. 15	Alta	15h 26m
Save bank information in backend and in Stripe	[Avatar]	ago. 16	Alta	11h 9m
Notification payment successfully	[Avatar]	ago. 18	Alta	7h 48m
Payment retry notice / notification	[Avatar]	ago. 18	Alta	12h 48m
Document and API to process Invoice	[Avatar]	ago. 19	Alta	6h 24m
Manage payment methods	[Avatar]	sep. 20	Alta	7h 37m

Figura 3.4: Sprint 4 - Backlog [Autoría Propia]

Sprint 5 (22/8 - 4/9) + NUEVA TAREA ✓ MOSTRAR CERRADOS

ago. 22 - sep. 4

TODO 13 TAREAS

TAREA	PERSONA ASIGNADA	FECHA DE VENCIMIENTO	PRIORIDAD	TIEMPO ESTIMADO
Fixing errors with Date Picker in the Payments view	[Avatar]	ago. 22	Alta	3h
Reconfigure service charge details	[Avatar]	ago. 22	Alta	3h
Show payment terms and announce if there is no existing payment method	[Avatar]	ago. 23	Alta	1h
Improve Information screen	[Avatar]	ago. 23	Alta	1h 30m
Improve Profile screen	[Avatar]	ago. 23	Alta	1h 50m
Send form data to a specific email (Help Screen)	[Avatar]	ago. 23	Alta	2h
Announce that there is no payment method when there are pending values.	[Avatar]	sep. 24	Alta	1h
Show visually the values that will be discounted	[Avatar]	ago. 24	Alta	3h
Improve Report Screen by fixing errors with Date Picker	[Avatar]	ago. 25	Alta	3h
Define the information needed to make payments	[Avatar]	ago. 24	Alta	4h
Automate the excel process	[Avatar]	ago. 25	Alta	4h
Implementation of a WebHook using Stripe	[Avatar]	oct. 3	Alta	6h
Notification payment successfully	[Avatar]	oct. 3	Alta	2h

Figura 3.5: Sprint 5 - Backlog [Autoría Propia]

Con el backlog definido se procedió a la implementación de la aplicación móvil de la solución, así como el desarrollo de los funciones y servicios que proveen la información necesaria para que la aplicación pueda ser usada. En todos los sprints tuvimos un *sprint retrospective* para discutir lo que salió bien durante el sprint y lo que se puede mejorar para el próximo, además que, en cada entregable hubo correcciones basadas en la retroalimentación del cliente.

3.2 Entrega de avances

Al finalizar cada Sprint, se agendaron reuniones junto con el cliente y los stakeholders como se puede observar en la Figura 3.6, para realizar una demostración de las funcionalidades que iba cumpliendo el producto.

27	JUL 2022, MIÉ	● 4:30 – 5:30pm	Awana Pay - demo
29	JUL 2022, VIE	● 2:30 – 3pm	Awana Pay - Demo presentation (MVP1)
5	AGO 2022, VIE	● 1 – 1:45pm	Awana Pay - Demo
9	AGO 2022, MAR	● 4 – 4:30pm	Awana Pay - final demo MVP1

Figura 3.6: Agenda de reuniones para presentación de Demos [Autoría Propia]

Durante estas reuniones se presentaban los nuevos cambios añadidos, las mejoras, y el flujo normal del uso del producto para evidenciar que las funcionalidades fueron implementadas acorde a lo establecido y además sirvieron para ir obteniendo retroalimentación por parte de ellos.

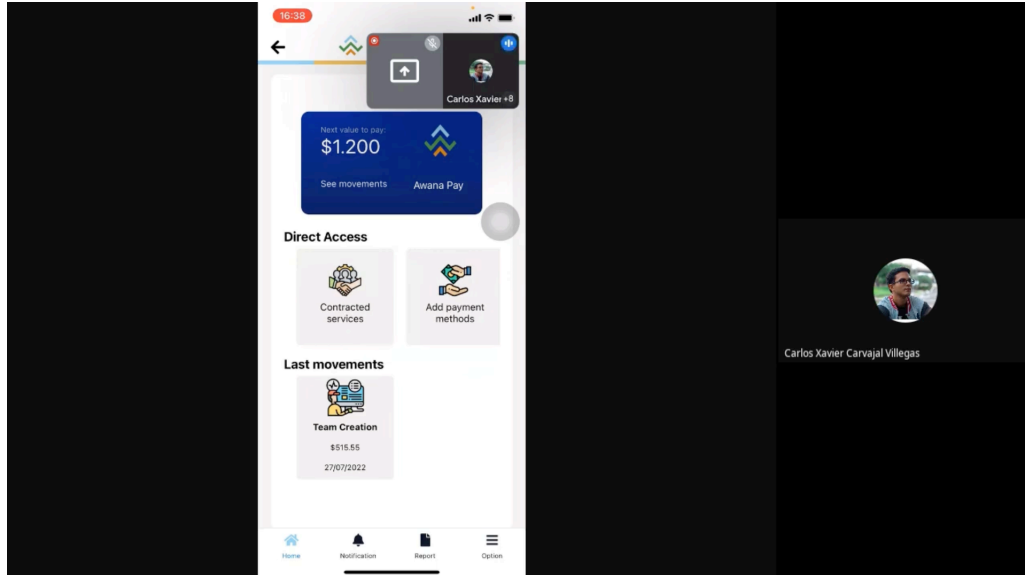


Figura 3.7: Evidencia de la presentación del producto al cliente [Autoría Propia]

3.3 Plan de Testing

Al finalizar cada sprint, los cuales fueron supervisados y aprobados por la PM que posteriormente comunicaba al cliente sobre el progreso del producto y las dificultades o bloqueos que se presentaban, los cuales solo surgieron en los 2 últimos sprints debido a cierta deuda técnica y al paso a producción por lo cual se procedió a extender el plazo, ya que se decidió refactorizar y realizar pruebas unitarias en componentes específicos logrando el 100% de cobertura en el caso del frontend como se indica en la Figura 3.8 y en el backend cada vez que se realice un cambio y se quiera integrar con cualquier ambiente de desarrollo se ejecutan las pruebas unitarias mediante una tarea en Github Action lo cual asegura una correcta integración con los servicios como se muestra en la Figura 3.9.

```
PASS tests /DirectAccess.test.js (16.236 s)
-----
File      % Stmts  % Branch  % Funcs  % Lines  Uncovered Line #s
-----
All files    100      100       100      100
Button     100      100       100      100
  ButtonAndroid.js  100      100       100      100
  ButtonIOS.js     100      100       100      100
  ButtonPayment   100      100       100      100
  ButtonPayAndroid.js 100      100       100      100
  ButtonPayIOS.js  100      100       100      100
Card        100      100       100      100
  CardAndroid.js  100      100       100      100
  CardIOS.js     100      100       100      100
  CardDescription 100      100       100      100
  CardPartner.js  100      100       100      100
CheckBox    100      100       100      100
  CheckBoxAwana.js 100      100       100      100
DirectAccess 100      100       100      100
  DirectAccess.js  100      100       100      100
  SearchBarController 100      100       100      100
  SearchBarController.js 100      100       100      100
Table       100      100       100      100
  DetailsTable.js  100      100       100      100
  NotificationTable.js 100      100       100      100
  ReportTable.js   100      100       100      100
-----
Test Suites: 13 passed, 13 total
Tests:       28 passed, 28 total
Snapshots:   0 total
Time:        17.314 s
Ran all test suites.
Done in 18.04s.
```

Figura 3.8: Pruebas Unitarias en el Front-End [Autoría Propia]

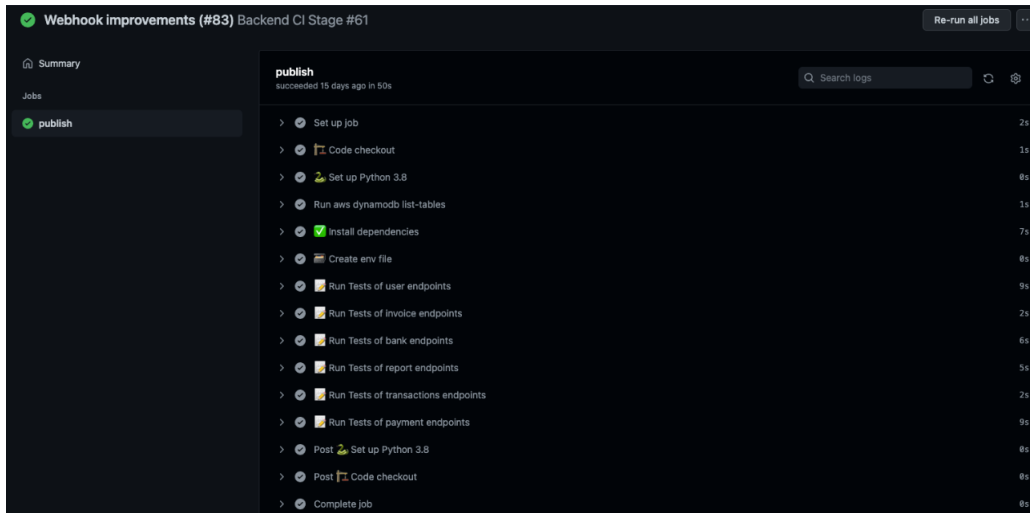


Figura 3.9: Pruebas Unitarias como parte del CI en el Back-End [Autoría Propia]

3.4 Productos Entregados

Como parte del acuerdo con el cliente, al finalizar la etapa de desarrollo e implementación se entregaron dos productos finales, uno de ellos fue el prototipo de la aplicación móvil en Figma y otro fue el código de la aplicación móvil los cuáles fueron aceptados por el cliente como se puede observar en el Apéndice D y E.

3.4.1 Prototipo

El plan de implementación contempló con una etapa de prototipado el cual fue desarrollado en Figma con el fin de obtener un producto tangible de cómo debería lucir y cómo debe ser la interacción y el flujo para tener una idea visual de cómo construirlo, además de crear recursos que más adelante permitirían que la construcción de la aplicación sea más rápida, además de que cuando un nuevo cambio visual era solicitado este servía para realizar pruebas de conceptos en donde se pudo corroborar que el producto final cumplía con las expectativas del cliente.

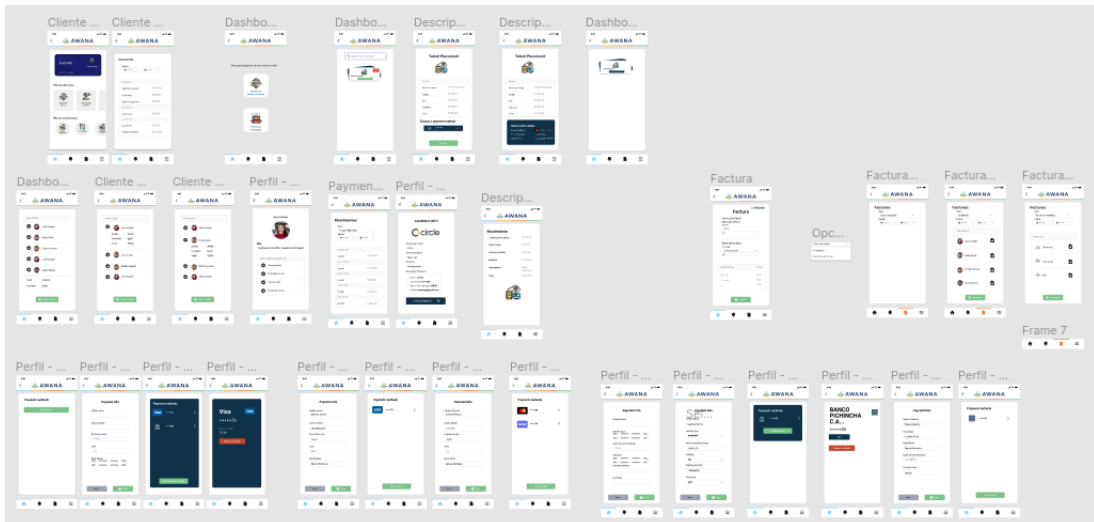
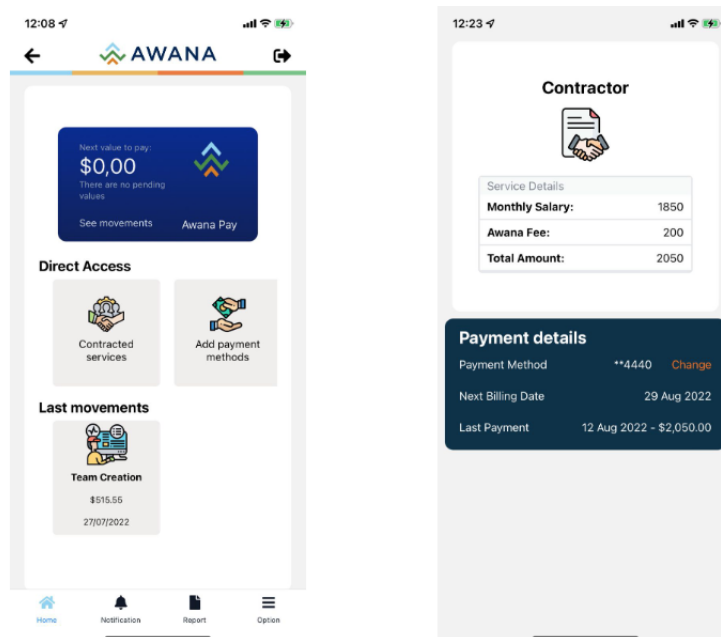


Figura 3.10: Prototipo de la aplicación en Figma [Autoría Propia]

3.4.2 Aplicación Móvil

Una vez culminado el tiempo estimado de desarrollo e integrado las pruebas unitarias para validar el correcto funcionamiento de los componentes y endpoints creados, se entregó el código tanto del frontend como del backend en sus respectivos repositorios luego de una última presentación en donde se corroboró que cumplía con todos los requerimientos que se definieron.



(a) Ventana principal

(b) Ventana de métodos de pago

Figura 3.11: Vistas de la Aplicación Móvil [Autoría Propia]

3.5 Análisis de costos

En el desarrollo de la solución se utilizaron varias tecnologías de uso gratuito por lo cual no representaron costos para el desarrollo de la aplicación. De igual manera, para el despliegue de la aplicación y manejar información bancaria mediante Stripe, se adquirieron **servicios on demand** también conocidos como **servicios bajo demanda**, los cuales traen beneficios en tema de reducción de costos para la empresa ya que, solo se paga por los servicios usados en el proyecto a medida que estos se vayan utilizando.

El código fuente de los diferentes desarrollos se alojaron dentro de la plataforma de GitHub el cual provee servicios gratuitos en el uso de los repositorios para el alojamiento del código.

El despliegue del frontend de la aplicación se realizó dentro del ambiente de Expo, el cual nos brinda un conjunto de servicios que hacen posible usar la aplicación desde nuestros dispositivos de forma nativa sin la necesidad de herramientas como Android Studio o Xcode. Por otro lado, para el despliegue del backend serverless se realizó dentro de un ambiente de pago mediante los servicios de AWS.

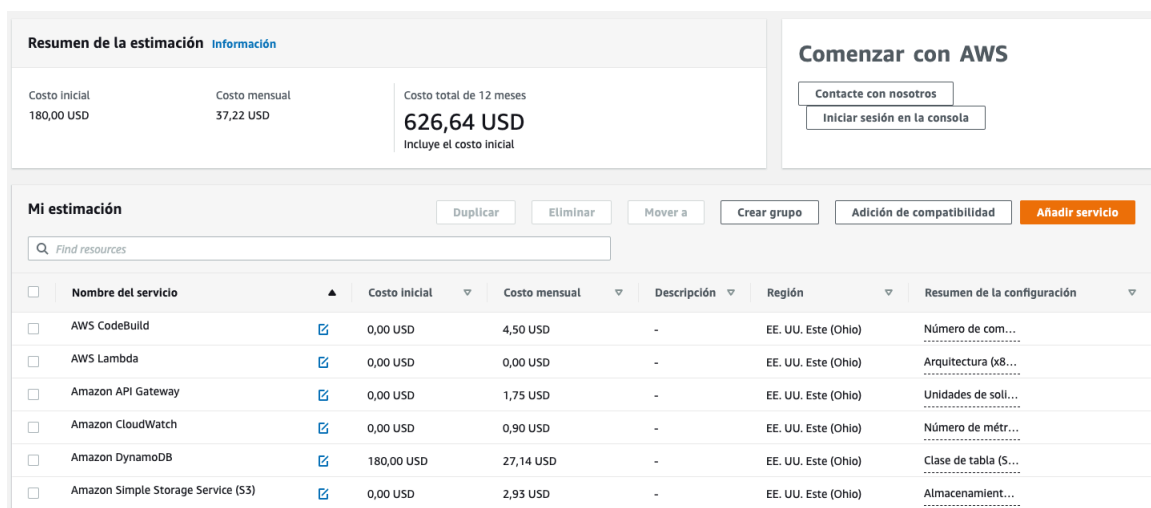


Figura 3.12: Estimación mensual de costos por Servicio durante el Desarrollo [Autoría Propia]

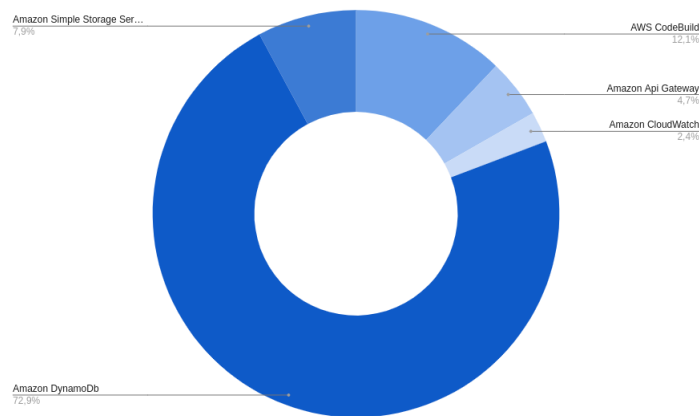


Figura 3.13: Costo por servicio [Autoría Propia]

Para el análisis de costo del desarrollo del proyecto, se contrastó la implementación a nuestro cargo en comparación con los costos asumidos al contratar desarrolladores externos en donde se obtuvo que la única diferencia surge debido al sueldo del desarrollador porque depende de su rango profesional ya que los servicios usados presentan precios fijos. Por lo que al diseñar la tabla de costos dado que el cliente es de EE. UU. se tiene como asunción que sí se desea contratar un desarrollador frontend especializado en React Native y uno especializado en el backend de dicho país con un nivel inicial, cobran alrededor de \$8,288 y \$8,362 al mes respectivamente, obteniendo así la Tabla 3.1.

Tabla 3.1: Presupuesto para el desarrollo del proyecto [Autoría Propia]

Descripción	Costo por mes
Sueldo de un Programador Front-End de nivel junior	\$8,288.00
Sueldo de un Programador Back-End de nivel junior	\$8,362.00
AWS CodeBuild	\$4.50
AWS Lambda	\$0.00
Amazon Api Gateway	\$1.75

Amazon CloudWatch	\$24.05
Amazon DynamoDb	\$43.42
Amazon Simple Storage Service (S3)	\$2.93
Auth0	\$23.00
Total	\$16,749.65

Con esta información de costos y teniendo en cuenta los gastos que se han presentado a lo largo del desarrollo y lo que podría costar hacer esta misma aplicación de manera externa además de sus gastos operacionales, se pudo estimar los costos de mantenimiento y operación de la aplicación móvil por año. Cabe recalcar que se pueden reducir costos eliminando ambientes de desarrollos, ya que en este caso se poseen tres ambientes, uno para desarrollo, otro para mostrar las nuevas características al cliente, y finalmente uno para producción, además de contratar desarrolladores de Latinoamérica.

Tabla 3.2: Costo estimado de operación anual del proyecto [Autoría Propia]

Descripción	Costo por año
Desarrollador FullStack con residencia en Latinoamérica	\$9,600.00
AWS CodeBuild	\$54.00
AWS Lambda	\$0.00
Amazon Api Gateway	\$21.00
Amazon CloudWatch	\$10.80
Amazon DynamoDb	\$325.68
Amazon Simple Storage Service (S3)	\$35.16
Stripe	\$24,000.00
Auth0	\$276.00
Despliegue en Apple	\$99.00
Despliegue en Android	\$25.00
Total	\$120,811.48

Stripe es un servicio que cobra a través de comisiones tomando en cuenta la cantidad y el tipo de transacciones que realiza, por lo que, si se realiza una transacción mediante una tarjeta de crédito, el valor de la comisión sería del 2.9% del monto de la transacción más 30 centavos y si es una transacción bancaria, el cobro será del 0.8% del monto.

Por petición del cliente se decidió manejar únicamente transacciones bancarias por un tiempo establecido a pesar de tener habilitada las transacciones a través de tarjetas de débito y crédito, por lo que para tener una estimación del costo operacional de Stripe se tiene como asunción que cuentan con 15 clientes, los cuales crean un ingreso aproximado de \$16,666.67 al mes de lo que Stripe recibe el 0.8% de ese monto lo que es equivalente a \$133.36, obteniendo como resultado un costo total estimado de \$2,000.00 al mes en Stripe por todos los pagos de sus clientes.

Teniendo en cuenta el número de compañías y la asunción de los ingresos netos estimados que recibe al mes, tenemos un total de \$250,000.00 mensuales, y un total de ingresos de \$3'000,000.00 anuales netos, de lo cual se invierte en Stripe un total de \$24,000.00 al año teniendo aún un margen de ganancias de \$2'976,000.00.

3.6 Cierre de Proyecto

Para finalizar el proyecto, se dispuso de una reunión con el cliente para realizar la presentación final del producto de tal forma que se corrobore su satisfacción quedando como evidencia un acta de aceptación firmada, la cual se presenta en el Apéndice E. Los entregables de este proyecto fueron:

- Prototipo en Figma.
- Producto final con su respectivo código fuente.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

La integración con el servicio de Stripe, facilitó el manejo de bancos y transacciones internacionales, abarcando la conexión con bancos americanos como *Wells Fargo* y el *Bank of America* lo cual se evidencia en la UI al momento de agregar los métodos de pago, que luego de ser validados es posible tener el registro de aquellos que están asociados al usuario, aumentando así la seguridad de los datos y robustez de la aplicación.

El módulo de notificaciones añadió mayor confiabilidad al sistema, ya que el usuario puede estar informado acerca de los eventos que ocurran, ya sea con las suscripciones, plazos a vencer o estados de pagos, restándole una preocupación extra y posibles inconvenientes, además de la incertidumbre al no recibir una verificación de alguna acción que realice.

Los servicios bajo demanda que provee AWS y Stripe usados en el backend permitieron un manejo eficiente y seguro de los datos financieros provistos por la compañía, además de poder configurar y administrar un servidor en la nube, disminuyendo cerca de un 70% de los gastos al desplegar un servidor físico.

4.2 Recomendaciones

En las pruebas de usabilidad, se pudo evidenciar la mejora de agregar un servicio extra para realizar la validación del propietario de la cuenta bancaria ingresada, ya que Stripe demora un periodo de 1 o 2 días hábiles en hacerlo, para lo que se recomienda utilizar un servicio de apoyo como *Plaid* para agilizar este proceso de verificación.

Como se menciona en el Capítulo 3 solo se realizaron pruebas unitarias, sin embargo, una forma de aumentar la confiabilidad del sistema entre los desarrolladores y el cliente es implementar pruebas *End to End* que verifican el correcto flujo e interacción y el sistema como un todo.

Otro punto de importancia que no se abordó en el presente trabajo es la posibilidad de agregar en futuras entregas la implementación para el manejo de pagos a los trabajadores de la compañía, lo cual agilizaría más el proceso que realizan actualmente además de generar un mayor valor al producto.

BIBLIOGRAFÍA

- [1] Upwork, “Third Annual “Future Workforce Report” Sheds Light on How Younger Generations are Reshaping the Future of Work,” tech. rep.
- [2] R. a. M. ltd, “Global Recruitment Process Outsourcing Market Research Report 2022 (Status and Outlook).”
- [3] S. Y. Rosero Manosalvas, “Estudio de factibilidad para la creación de una empresa de selección de personal para las pequeñas y medianas empresas (Pymes) en la ciudad de Quito,” 2014.
- [4] B. Anikin and I. Rudaya, *Outsourcing and outstaffing: high management technologies*. ru: INFRA-M Academic Publishing LLC., 0 ed., Nov. 2021.
- [5] D. Minoli, *Analyzing outsourcing: reengineering information and communication systems*. New York: Mcgraw-Hill, 1995.
- [6] V. Saini, D. C. Yen, and D. C. Chou, “Information Technology Outsourcing: Issues and Future Analyses,” 2016, vol. no. October, pp. 400–407.
- [7] A. Denadai, L. Lehr, and M. Gil, “Payments, commerce, and life post-COVID-19 quarantine,” 2020.
- [8] “Stripe® Official | Plataforma de procesamiento de pagos por Internet.”
- [9] “Paypal vs Stripe. ¿Cuál es la mejor pasarela de pago?,” Apr. 2018.
- [10] S. M. Monreal Mendoza, *Sistemas de Pago para Comercio Electrónico*. PhD thesis, Centro de Investigación en Matemáticas A.C., México.
- [11] “PayPal: la forma fácil y segura de hacer y recibir pagos en línea..”
- [12] “React Native · Learn once, write anywhere.”
- [13] “Flutter - Build apps for any screen.”
- [14] “AWS | Cloud Computing - Servicios de informática en la nube.”
- [15] “Servicios de cloud computing.”
- [16] “Servicios de informática en la nube | Microsoft Azure.”
- [17] D. |. F. S. D. Company, “Cross-platform Mobile Development: Flutter Vs React

Native.”

[18] “React Native vs Flutter: Which one is better for 2022?,” Apr. 2020.

[19] “Flutter FAQ’s.”

[20] “¿Qué es Amazon DynamoDB? - Amazon DynamoDB.”

[21] “¿Qué es AWS Lambda? - AWS Lambda.”

[22] “¿Qué es Amazon API Gateway? - Amazon API Gateway.”

[23] J. P. Zumba, “Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software,” *INNOVA Research Journal*, vol. 3, pp. 20–33, Oct. 2018.

[24] Atlassian, “Scrum: qué es, cómo funciona y por qué es excelente.”

APÉNDICES

Apéndice A

Entrevistas A Perfiles De Usuarios

Figura A.1: Entrevista a Perfiles de Usuarios [Autoría Propia]

PROPÓSITO GENERAL DE LAS ENTREVISTAS:

El propósito de las entrevistas es recopilar información que permita conocer los distintos problemas que se afrontan en el proceso de cobros de servicios. Al conocer los pensamientos, puntos de vista y opiniones de cada uno de los involucrados se logra comprender con mayor profundidad la problemática, conociendo las dificultades que se presentan, así como también conocer las facilidades que desean tener los involucrados en este proceso.

TIPO DE ENTREVISTAS A REALIZAR:

Semiestructurada ya que esta me permite interactuar de mejor manera con la persona entrevistada porque tiene preguntas subjetivas y objetivas.

Perfil del Cliente/Usuario 1: Clientes

1. ¿Qué dificultades afronta al momento de realizar el pago a sus trabajadores Latinoamericanos?
2. ¿Qué método le resulta más eficiente al momento de realizar los pagos? ¿Por qué?
3. ¿Cómo maneja el tema de impuestos locales para los diferentes empleados Latinoamericanos que trabajan para usted?
4. ¿Cuánto tiempo se demora en calcular los impuestos locales en caso de hacerlo y mandar los pagos para un empleado?
5. ¿Cómo maneja la elección del día de corte de pago? ¿Cada cuánto tiempo es el corte de pago?
6. ¿Qué significa para usted pagar atrasado a un empleado?
7. ¿Estaría de acuerdo con utilizar un método de pago recurrente? este sería tarjeta de crédito/debito, cuenta bancaria
8. ¿Qué requisitos necesita del cliente para realizar los pagos?
9. ¿Qué herramientas utiliza para llevar el control de sus empleados y sus respectivos pagos?
10. ¿Cómo sabe usted si los pagos han llegado correctamente al cliente?
11. ¿Qué datos de la factura son necesarios para usted a la hora de emitir un pago?
12. ¿En qué forma reciben las facturas actualmente?

Perfil del Cliente/Usuario 2: Empleados

1. ¿Qué dificultades afronta al momento de recibir los pagos de parte de sus empleadores?
2. ¿Qué método le resulta más eficiente al cobrar los pagos?
3. ¿El proceso de recibir el pago suele tardar? ¿Cuánto tiempo ha tenido que esperar?
4. ¿Cómo es el proceso al momento de emitir facturas? ¿Qué dificultad(es) presenta?
5. ¿Cómo se le notifica que su pago ha llegado a su cuenta?
 - a. ¿Por cuál medio le gustaría recibir estas notificaciones?
6. ¿Qué acciones ha tomado con el tema de pagos atrasados?
7. ¿Cómo ha procedido cuando no ha podido emitir una factura para cobrar su remuneración?
8. ¿Qué significa para usted recibir tu pago atrasado?

Perfil del Cliente/Usuario 3: RRHH

1. ¿Cómo es el proceso que realizan para el registro de empleados o clientes?
2. ¿Qué obligaciones presentan los clientes o empleados con el actual modo de pago?
3. ¿Ha recibido quejas al momento de pagar a sus empleados? ¿Cuáles han sido los motivos?
4. ¿Cómo es el proceso que ustedes realizan para enviar los pagos a los diferentes empleados que constan en la nómina?
5. ¿Qué procesos son repetitivos para usted?, ¿Cuáles cree que se puedan automatizar?
6. ¿Qué herramientas utilizan para la administración de la nómina de pagos?
7. ¿Qué herramientas utilizan para realizar la cobranza y los pagos?
8. ¿Cómo se notifica a los empleados la realización de los pagos?

Apéndice B

Actas

Figura B.1: Entrevista 1 [Autoría Propia]

ACTA DE ENTREVISTA

Entrevistado	Samuel Pilay	Cargo	Empleado
Entrevistador	Carlos Carvajal		
Lugar de entrevista	Zoom		
Hora de entrevista	10:30	Fecha	21/06/2022

TEMAS TRATADOS:

“Detalles del proceso de cobro para los perfiles de usuarios”.

Preguntas

1. ¿Qué dificultades afronta al momento de recibir los pagos de parte de sus empleadores?
2. ¿Qué método le resulta más eficiente al cobrar los pagos?
3. ¿El proceso de recibir el pago suele tardar? ¿Cuánto tiempo ha tenido que esperar?
4. ¿Cómo es el proceso al momento de emitir facturas? ¿Qué dificultad(es) presenta?
5. ¿Cómo se le notifica que su pago ha llegado a su cuenta?
6. ¿Por cuál medio le gustaría recibir estas notificaciones?
7. ¿Qué acciones ha tomado con el tema de pagos atrasados?
8. ¿Cómo ha procedido cuando no ha podido emitir una factura para cobrar su remuneración?
9. ¿Qué significa para usted recibir tu pago atrasado?

Desarrollo

Sistema financiero complicado del país.

Transferencia directa.

La empresa que envía tiene que controlar los valores que se asumen, le restan lo que tiene que restar y envía el dinero, eso dura como 6 días y en llegar el dinero al banco a lo mucho 3 días.

Depende mucho de la empresa por la que trabajas en este caso ocurre que pide factura autorizada por el SRI (laboratorio LA) les interesa que todo sea legal. En otra empresa, solo esperan 4 meses, no era tan relevante emitir factura dado los valores.

Depende de la empresa, incluso hay bancos que te avisan por correo que te ha llegado remesas del exterior.

Prefiero notificaciones por correo.

Me tocó ponerme en contacto con el banco, depende de la atención al cliente en Banco del Pacífico fue muy rápido, pero por lo general la persona que no tiene esa facilidad tiene que hacer un trámite largo. El dinero llegaba incompleto, y no sabía que era, era el banco del exterior que retiene esos valores y la empresa demora días en contactar el banco y ya luego llegaban los valores completos. Solo me ha tocado esperar, también depende del desarrollador como quiera proceder.

Opto siempre por lo burocrático

Como algo que puede pasar, estamos en una época donde la digitalización pasa el 100% problemas con hackers ya no depende de la empresa.

Figura B.2: Entrevista 2 [Autoría Propia]

ACTA DE ENTREVISTA

Entrevistado	Gerson Mayer	Cargo	Empleado
Entrevistador	Freddy Ponce		
Lugar de entrevista	Zoom		
Hora de entrevista	11:30	Fecha	21/06/2022

TEMAS TRATADOS:

“Detalles del proceso de cobro para los perfiles de usuarios”.

Preguntas

1. ¿Qué dificultades afronta al momento de recibir los pagos de parte de sus empleadores?
2. ¿Qué método le resulta más eficiente al cobrar los pagos?
3. ¿El proceso de recibir el pago suele tardar? ¿Cuánto tiempo ha tenido que esperar?
4. ¿Cómo es el proceso al momento de emitir facturas? ¿Qué dificultad(es) presenta?
5. ¿Cómo se le notifica que su pago ha llegado a su cuenta?
6. ¿Por cuál medio le gustaría recibir estas notificaciones?
7. ¿Qué acciones ha tomado con el tema de pagos atrasados?
8. ¿Cómo ha procedido cuando no ha podido emitir una factura para cobrar su remuneración?
9. ¿Qué significa para usted recibir tu pago atrasado?

Desarrollo

Hasta ahora solo tuvimos un retraso, pero el mayor problema es que no podemos cobrar hasta los 2 o 3 primeros días del mes, y hubo problemas con la transferencia una vez, son las únicas veces.

Recibirlo por una transferencia bancaria, por cheque imposible para mí es obsoleto, sería bueno por PayPal, pero es problemático.

Cuando ha sido directamente de Ecuador a mi cuenta, de 3 a 4 horas, cuando viene de USA a (Bco. pichincha) demora más de 24 horas y cobraron demasiado, 6,5% del total, por otro banco (Produbanco) recibí y no demoro ni 3 horas, pero cobraron 3,5% de tasa

No estoy recibiendo facturas, sino notas de venta.

Mediante un grupo de WhatsApp y toca esperar a que el pago se haga efectivo.

Por WhatsApp está bien, pero considero que por un correo es más privado.

Fue a todos, y simplemente tuvimos que esperar, un amigo ayudó a arreglar ese problema con el banco, porque el problema fue con el banco de la empresa.

Es feo tener que hacer tantas cosas, estuve sin internet 2 días y no pude trabajar, fallar en el presupuesto es doloroso.

Figura B.3: Entrevista 3 [Autoría Propia]

ACTA DE ENTREVISTA

Entrevistado	Jairo Suarez	Cargo	Recursos Humanos
Entrevistador	Carlos Carvajal		
Lugar de entrevista	Zoom		
Hora de entrevista	16:00	Fecha	22/06/2022

TEMAS TRATADOS:

“Detalles del proceso de cobro para los perfiles de usuarios”.

Preguntas

1. ¿Cómo es el proceso que realizan para el registro de empleados o clientes?
2. ¿Qué obligaciones presentan los clientes o empleados con el actual modo de pago?
3. ¿Ha recibido quejas al momento de pagar a sus empleados? ¿Cuáles han sido los motivos?
4. ¿Cómo es el proceso que ustedes realizan para enviar los pagos a los diferentes empleados que constan en la nómina?
5. ¿Qué procesos son repetitivos para usted?, ¿Cuáles cree que se puedan automatizar?
6. ¿Qué herramientas utilizan para la administración de la nómina de pagos?
7. ¿Qué herramientas utilizan para realizar la cobranza y los pagos?
8. ¿Cómo se notifica a los empleados la realización de los pagos?

Desarrollo

No registro los clientes sino a los empleados, como los reclutadores que buscan información en las empresas y contactan a la persona que necesita el trabajo. Yo voy cuando el empleado tiene muchas preguntas, de cualquier país, tienen muchas dudas.

Me envían un correo, me presento y les explico quiénes somos, si la persona esta contratada, se da información del banco, se da la opción de entregar facturas al 12% o de entregar recibo digital, donde envío salario por medio del banco internacional y próximamente estaremos con el bolivariano.

Los pagos vienen directos de USA, es mejor tener diferentes bancos en Ecuador porque su tarifa varía, tenemos dos tipos de contrato, formales donde le pagamos todo lo indicado, o un contrato de servicios prestados, con un salario mensual o quincenal.

Al momento de enviar recibos digitales lo quiero 8 días antes de fin de mes, ya que no conozco a las personas no sé cuánto tiempo trabajó, entonces lo envío a Javier o Calvin y ellos me confirman cuanto trabajo legalmente la persona, pasa más que la persona tiene más horas trabajadas de las que cree, tenemos un grupo de empleados por WhatsApp, donde informo cuando ya he pagado, los de contrato formales solo les envié su payroll, a servicios prestados los colocamos en un Excel.

Tenemos casi 3 años, hemos tenido problemas por parte del banco donde se retrasó con los pagos, fue en un feriado, fue la única vez que ha pasado, los empleados normalmente están felices con nosotros de recibir su pago al día.

Con la nómina enviamos el pago de 5 en 5 y se nos hace más fácil.

Quiero creer que el banco bolivariano será mejor que el internacional, ya que este está debiendo un poco.

Hacer la nómina cada mes es repetitivo porque son los mismos datos y solo cambia el salario.

De USA a Ecuador tarda un día, y aquí en Ecuador tarda unos 10 a 30 minutos y llega a su banco, donde el tiempo ahí ya depende del mismo.

Apéndice C

Especificación de Requerimientos

Figura C.1: Requerimientos Funcionales [Autoría Propia]

1. Requerimientos Funcionales

Table de Importancia/Urgencia de requerimientos

Alta	1
Neutra	2
Baja	3

RF-1	Infraestructura de pagos para Internet
Versión	1.0
Tipo	Desarrollo
Dependencias	
Historia de Usuario	Como <i>Company</i> , yo quiero <i>realizar transferencias o pagar con mi tarjeta de débito o crédito desde Estados Unidos</i> para <i>hacer efectiva la paga de mis servicios que mantengo con Awana</i> .
Descripción	El sistema deberá integrar un servicio de pagos por internet, que acepte pagos por tarjeta o transferencias bancarias.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para evaluar los diferentes métodos de pago que usan para realizar el pago de sus servicios desde USA.

RF-2	Métodos de pago
Versión	1.0
Tipo	Desarrollo
Dependencias	RF-1
Historia de Usuario	Como <i>Company</i> , yo quiero <i>poder agregar varios métodos de pago</i> para <i>poder pagar los servicios adquiridos con Awana</i> .
Descripción	El sistema permitirá a los clientes ingresar su banco o tarjetas de crédito o débito para realizar el cobro de sus servicios.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que el rol Company tenga la libertad de agregar métodos de pago para pagar sus servicios.

RF-3	Edición del método de pago predefinido
Versión	1.0
Tipo	Desarrollo
Dependencias	RF-2
Historia de Usuario	Como <i>Company</i> , yo quiero <i>poder editar el método de pago predefinido y seleccionar otro</i> para <i>poder pagar los servicios adquiridos con Awana</i> .
Descripción	El sistema permitirá a los clientes elegir su método de pago por defecto por el cual se realizará el cobro.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que el rol Company tenga la libertad de seleccionar el método de pago pagar sus servicios.

RF-4	Anulación de la información bancaria
Versión	1.0
Tipo	Mejora
Dependencias	
Historia de Usuario	Como <i>Company</i> , yo quiero <i>tener la posibilidad de eliminar la información bancaria registrada como métodos de pago que no use</i> para <i>poder tener solamente los métodos de pago que servirán en pagos posteriores</i> .
Descripción	El sistema permitirá la anulación de la información bancaria que el cliente desee borrar de su lista de métodos de pago.
Importancia	2
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que el rol Company puedan eliminar su información bancaria.

RF-5	Notificaciones de pagos realizados
Versión	1.0
Tipo	Desarrollo
Dependencias	
Historia de Usuario	Como <i>Company</i> , yo quiero <i>recibir notificaciones de los pagos realizados por correo o push</i> para <i>tener un mejor control del valor cancelado</i> .
Descripción	El sistema permitirá enviar notificaciones a los clientes cuando se les haya realizado el cobro.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders y el personal de Administración para evaluar la situación de notificar a los clientes del cobro de los servicios.

RF-6	Notificaciones previo a vencimientos de pagos
Versión	1.0
Tipo	Desarrollo
Dependencias	
Historia de Usuario	Como <i>Company</i> , yo quiero <i>recibir notificaciones previo a los vencimientos de pagos por correo o push</i> para <i>tener un tipo de recordatorio de que debo cancelar mis valores pendientes</i> .
Descripción	El sistema permitirá enviar notificaciones a los clientes previo a la fecha de pago como especie de recordatorio.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders y el personal de Administración para evaluar la situación de notificar a los clientes previo a la fecha de pago.

RF-7	Visualizar información de los trabajadores
Versión	1.0
Tipo	Mejora
Dependencias	
Historia de Usuario	Como <i>Company</i> , yo quiero <i>visualizar la información pertinente de los trabajadores</i> para <i>visualizar fácilmente información sobre ellos</i> .
Descripción	El sistema permitirá a los clientes visualizar la información pertinente de sus trabajadores.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que el personal de Administración y los clientes puedan revisar la información de los trabajadores.

RF-8	Visualizar información bancaria
Versión	1.0
Tipo	Mejora
Dependencias	RF-3
Historia de Usuario	Como <i>Company</i> , yo quiero <i>visualizar el historial de pagos realizados, y los próximos a vencer, especificado por los servicios que mantengo con Awana</i> para <i>llevar un mejor control de los mismos</i> .
Descripción	El sistema permitirá a los usuarios visualizar su información bancaria y el historial de pagos.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que los usuarios con el rol Company , puedan visualizar el historial de pagos realizados, y los próximos a vencer, especificado por los servicios que mantienen con Awana de forma que se le facilite llevar un control de los mismos.

RF-9	Detalle Factura
Versión	1.0
Tipo	Desarrollo
Dependencias	RF-1
Historia de Usuario	Como <i>Company</i> , yo quiero <i>que la factura que se me envíe al momento del cobro tenga descripción del pago a realizar para tener una mejor idea de qué estoy pagando en el mes.</i>
Descripción	La factura que genere el sistema debe contener un detalle del pago, para que el personal de Administración sepa si se trata de un pago del sueldo mensual, o algún otro pago como el del décimo.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevista con el personal de Administración para conocer y determinar los campos necesarios para la elaboración de una factura.

RF-10	Configurar idioma
Versión	1.0
Tipo	Mejora
Dependencias	
Historia de Usuario	Como <i>usuario final</i> , yo quiero <i>elegir mi idioma nativo para usar la aplicación.</i>
Descripción	El sistema permitirá a los usuarios escoger el idioma en el cual visualizar la página.
Importancia	2
Urgencia	2
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para corroborar que el usuario pueda elegir el idioma de preferencia.

Figura C.2: Requerimientos No Funcionales [Autoría Propia]

2. Requerimientos No Funcionales

2.1. De Producto

2.1.1. Seguridad.

RNF-1	Gestión de confidencialidad de la información
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	El sistema debe ser accedido únicamente por usuarios autorizados.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para conocer como otorgar el respectivo acceso a cada uno de los usuarios del sistema.

2.1.2. Confiabilidad.

RNF-2	Capas de autorización
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	La aplicación debe estar integrada con los servicios de auth0 que maneja la empresa para añadir una capa de autorización a cada rol y seguridad a la aplicación
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Uso de prototipo para verificar que la capa de autorización propuesta sea correcta y adecuada

2.2 Organizacionales

2.2.1 De Ambiente.

RNF-3	Portabilidad móvil multiplataforma
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	La aplicación móvil debe operar tanto en iOS como en Android.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para evaluar si la portabilidad de la aplicación móvil cumple con los requerimientos de desarrollo.

RNF-4	Uso de Cloud Computing
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	El sistema deberá ser escalable por lo que utilizará servicios en la nube de AWS que ofrece la empresa.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Casos de prueba para comprobar que los servicios sean óptimos y cubran las necesidades del sistema.

2.2.2 Desarrollo.

RNF-5	Uso de Stripe para el cobro seguro en Estados Unidos
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	El sistema deberá integrar Stripe como servicio de pago para el cobro de servicios en los Estados Unidos.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para verificar que el servicio pueda satisfacer de manera óptima el cobro de dinero dentro de los Estados Unidos.

RNF-6	Uso del framework React Native
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	La aplicación móvil debe desarrollarse con el framework React Native que pertenece al stack que maneja la empresa.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para evaluar la factibilidad del framework escogido en el desarrollo de la aplicación móvil.

2.3 Externos

2.3.1 Legislativo.

RNF-7	Protección de datos
Versión	1.0
Tipo	Desarrollo
Dependencias	
Descripción	El sistema debe cumplir los requerimientos en la ley de protección de datos.
Importancia	1
Urgencia	1
Estado	
Comentarios	Ninguno
Criterio de validación	Entrevistas con los stakeholders para acordar los requerimientos en cuanto lo establezca la ley.

Apéndice D

Carta de Aceptación del Prototipo

Figura D.1: Carta de Aceptación del Prototipo [Autoría Propia]

Conformity Certificate

Guayaquil, June 22, 2022.

To whom it may concern,

I, Johana Vásquez Cielo identified with the identification number 1757691645 of the city of Guayaquil, in my capacity as Legal Representative of the company **AWANA** with the role of *Company Operations Manager*, I allow myself to state that I am satisfied with the mobile prototype presented for this project, I also certify that I have reviewed and approved the new changes to the prototype. In addition, I have read and approved the *Specification of Requirements* document.

Sincerely,



Firm
Company Operations Manager

Apéndice E

Carta de Aceptación del Producto Final

Figura E.1: Carta de Aceptación del Producto Final [Autoría Propia]

Conformity Certificate

Guayaquil, August 26, 2022.

To whom it may concern,

I, Calvin Smith _____ identified with the identification number 84-4675014 _____ of the city of New York, in my capacity as Legal Representative of the company **AWANA**, I allow myself to state that I am satisfied with the delivered product called **Awana Pay**, system that has been developed for **Awana** by the students Carlos Carvajal and Freddy Ponce to obtain their Computer Science degree at ESPOL.

Sincerely,

Calvin Smith

Calvin Smith - CEO of Awana

c.c. *MAS*

Manuel Suarez
Tech Lead

c.c. *JVásquez*

Johana Vásquez Cielo
Company Operations Manager