

# Automatización con aplicación móvil para agendamiento de cita en peluquerías

Fecha de recepción: 2021-10-25 • Fecha de aceptación: 2021-01-04 • Fecha de publicación: 2022-02-10

**Miguel Angel Pino Andrade<sup>1</sup>**

Investigador independiente, Ecuador

*lucariodrago@gmail.com*

<https://orcid.org/0000-0002-3044-7765>

**Arthur Wilson Chávez Mora<sup>2</sup>**

Investigador independiente, Ecuador

*arthurchavez2399@gmail.com*

<https://orcid.org/0000-0001-7600-345X>

**Juan Daniel Carrion Robles<sup>3</sup>**

Investigador independiente, Ecuador

*dann191812@gmail.com*

<https://orcid.org/0000-0001-6518-9105>

## RESUMEN

El agendamiento de citas en peluquerías es un proceso que la mayoría de los locales no ha automatizado, ya que el servicio se brinda de acuerdo a cómo lleguen los clientes, esto es un problema, ya que puede tomar mucho tiempo en recibir el servicio de peluquería. En este sentido, se propone una aplicación capaz de agendar citas para evitar la espera, con el fin de optimizar el mencionado proceso. Se idea la aplicación desde dos perspectivas, la de usuario; es decir, clientes de la peluquería y la de administrador, el cliente solamente elegirá la hora, día y la sucursal en la cual desea ser atendido, el sistema automáticamente asignará la cita con el peluquero indicado, reservará ese tiempo para que no haya problemas, además de eliminar la espera, a su vez, el próximo cliente acudirá al término de la atención del cliente en turno. El desarrollo de la aplicación se realizará bajo Visual Studio XAMARIN, mediante las directrices de la metodología c, los resultados iniciales

demuestran la efectividad de la aplicación.

**PALABRAS CLAVE:** Xamarin, Visual Studio, Scrum, Github, Open Source

## **ABSTRACT**

El agendamiento de citas en peluquerías es un proceso que la mayoría de los locales no ha automatizado, ya que el servicio se brinda de acuerdo con como lleguen los clientes, esto es un problema ya que puede tomar mucho tiempo en recibir el servicio de peluquería. En este sentido se propone una aplicación capaz de agendar citas para evitar la espera, con el fin de optimizar el mencionado proceso se idea la aplicación desde dos perspectivas, la de usuario, es decir clientes de la peluquería y la de administrador, el cliente solamente elegirá la hora, el día y la sucursal en la cual desea ser atendido, el sistema automáticamente asignará la cita con el peluquero indicado, reservará ese tiempo para que no haya problemas, además de eliminar la espera, a su vez el próximo cliente acudirá al término de la atención cliente en turno. El desarrollo de la aplicación se realizará bajo Visual Studio XAMARIN, mediante las directrices de la metodología Scrum, los resultados iniciales demuestran la efectividad de la aplicación.

**KEYWORDS:** Xamarin, Visual Studio, Scrum, Github, Open Source

## Introducción

En los últimos años se ha evidenciado un creciente desarrollo de sistemas computacionales e informáticos para automatizar procesos de distintos negocios que se realizan de forma manual, y resolver distintos problemas que afectan a los usuarios, dentro de estas aplicaciones se pueden mencionar aplicaciones web (Nagy et al., 2008), móviles (Reyes et al., 2019), de escritorio (Leung, 2015), entre otras.

Existen distintos procesos que realizan ciertos negocios, que por más sencillos que parezcan, deben ser automatizados para mejorar el servicio al cliente, la literatura demuestra que existen soluciones para distintos tipos de servicio, desde la solicitud de un servicio de autos puerta a puerta (Artieda Guachamin et al., 2021), servicios de compra y venta a domicilio (Reyes Rivera et al., 2020), servicios de localización de mascotas (Burbano Ulloa et al., 2020), servicios de reconocimiento facial (Nuñez & Nuñez, 2020), sistemas de realidad virtual (Florez et al., 2021), pero muy poco se ha considerado el agendamiento de uno de los servicios más utilizados por la mayoría de personas en el mundo, como lo es el servicio de peluquería.

Si bien existen trabajos que mencionan este tipo de negocio, no se identifica un sistema actual en Ecuador que pueda automatizar el proceso de agendamiento de citas en una peluquería, y que se respeten los tiempos de los clientes, inicialmente San Martín Galaz (2005) desarrolla un “prototipo de aplicación informática que permita el registro de la información de clientes para facilitar la toma de decisiones en los procesos de gestión administrativa y financiera de la microempresa Peluquería Nuevo Estilo”.

Por otro lado, Luzardo Cabrera (2014) propone “desarrollar una aplicación Android para pedir cita previa en peluquerías, tomando como ejemplo Peluquerías Naranja, una franquicia de peluquerías canaria, esta aplicación permite al usuario del dispositivo, reservar hora o pedir cita en la peluquería, la cita se reservará siguiendo una serie de pantallas, en las que la información introducida de forma táctil se almacenará posteriormente en una base de datos online, asimismo, la aplicación será capaz de gestionar las citas previas disponibles permitiendo al usuario cancelar su cita previa”.

En Ecuador, la automatización de estos procesos aún es precaria, uno de los autores de este trabajo acudió a recibir un corte de cabello en el local de su confianza, cuando arribó vio que el peluquero estaba con un cliente en turno, mantuvieron una conversación en la cual esclarecieron en el horario que se desocuparía y pudiera acudir para recibir el servicio, sin tener que esperar. El mismo regresó en el horario pactado, finalmente pudo recibir el servicio.

De la situación descrita anteriormente, se visualizó la idea de crear una aplicación móvil con el fin de no tener que esperar o tener que regresar para ser atendido por un peluquero. La idea se fue materializando en la aplicación llamada “Peluquéate”, la misma está diseñada para controlar el tiempo de sus peluqueros y tener a la disposición de todos los clientes, suscritos a la aplicación, con un horario organizado que desemboque en un tiempo menor al empleado, antes del uso de la aplicación para recibir un servicio, lo llamaremos “corte de cabello”.



# Metodología

La metodología aplicada en esta investigación es la observación directa, que se define como cuando “el investigador está ahí, en el lugar donde se desarrolla la acción y está preparado para registrar lo que está ocurriendo” (González, 2005), interpretando este concepto al contexto en que se desarrolla el presente estudio, lo que significa que los datos se obtienen directamente desde la fuente; es decir, los distintos negocios de peluquería ubicados en Quito, Ecuador.

En cuanto al desarrollo de software, se utiliza la metodología Scrum, a continuación, se detalla el proceso.

## 2.1 Modelo tradicional PDCA

El modelo para la elaboración de cada fase del proyecto, siguiendo la metodología Scrum, fue desarrollado por Walter Shewhart (Rigby et al., 2016), en los laboratorios de la empresa Bell y difundido por W. Edwards Deming. El modelo se centra en cuatro pasos: planificar, hacer, verificar, actuar (PDCA, por sus siglas en inglés). Cada acción se compone de una serie de acciones, las mismas que se describen a continuación de acuerdo a cada paso:

## 2.2 Definición de metodología

Se puede definir la palabra metodología como una disciplina que muestra los pasos a seguir para cada lineamiento que lleva la realización de un proyecto. Los elementos que tiene una metodología son: fases, control y evaluación, documentación, **métodos**, técnicas y herramientas (Montero et al., 2018).

Hay puntos en común de cualquier metodología como lo es el tiempo de aprendizaje, este no debe ser largo. A su vez, el apoyo en forma gráfica es muy importante, también se usa documentación estándar y se trabaja en pro de simplificar el trabajo y aumentar la productividad.

De acuerdo con su naturaleza, las metodologías son orientadas al flujo de información u objetivos, o son híbridas. Mientras que, de acuerdo con el grado de formalismo, pueden ser pesadas o ágiles.

Una metodología ágil se concentra en dar respuestas a falencias que han tenido lugar en las metodologías tradicionales. Se adapta a los requisitos que no se conoce, en lugar de predecirlos; es decir, no trabaja con dichos requisitos, una metodología tradicional sí y los predice, en este sentido Scrum es una metodología ágil (Amaya Balaguera, 2015).

## 2.3 ¿Qué es SCRUM?

La metodología ágil se caracteriza por trabajar en contextos que tienen incertidumbre, autoorganización, control moderado y transmisión del conocimiento (Sangama Oñate, 2020)

La metodología tiene como base crear ciclos breves para la realización del proyecto, las cuales son llamadas iteraciones, bajo la metodología mencionada se llama “Sprints”, la cual tiene como fases:

- Planificación del *backlog*: aquí se define un documento en el cual se plasman los requisitos del sistema, se mantiene una reunión llamada “Sprint 0”, en la que se definen los objetivos y el trabajo a realizar.
- Seguimiento del sprint: se hacen reuniones para evaluar el avance de las tareas en base a estas tres preguntas: ¿qué se hizo desde la reunión anterior?, ¿cuál es el trabajo hasta la próxima reunión? y ¿qué inconvenientes han surgido y qué hay que solucionar para continuar?
- Revisión del sprint: al finalizar el sprint se hace una reunión para revisar los avances y el incremento generado, además, se presentarán los resultados finales.

También se toman en cuenta los roles que tienen las personas comprometidas con el proyecto, estos pueden ser

- *Product owner*: es un intermediario con el cliente, toma las decisiones de los clientes, se encarga de colocar sus ideas en el *product backlog* y ordenarlas por su prioridad.
- Scrum Master: encargado comprobar la metodología y el modelo para que puedan ser funcionales. Elimina los inconvenientes para el proceso.
- Equipo de desarrollo: está conformado de 5 a 9 personas y tienen autoridad para tomar las decisiones que encaminen al proyecto a su objetivo.

A su vez, las personas no involucradas directamente, que vendrían siendo:

- Usuarios: es el cliente como tal.
- Stakeholders: las personas a las que el programa les producirá un beneficio.
- Managers: toman las decisiones finales participando en la selección de requisitos y objetivos.

También es importante tener en cuenta los elementos de Scrum, tales como:

- Product Backlog: es la lista de necesidades de los clientes.
- Sprint Backlog: lista de tareas que se realizan en un “sprint”.
- Incremento: parte añadida y totalmente operativa en un “sprint”.

## 2.4 Metodología en “peluquéate”

De acuerdo con las directrices de la metodología Scrum y los elementos en los cuales está basado, se ha decidido utilizar dicha metodología para el desarrollo del proyecto; sin embargo, al ser un proyecto desarrollado por visión e iniciativa de los autores, y no por petición de una

empresa o persona natural para fines comerciales, los roles se han tomado solamente de “Scrum Master” para Arthur Chávez y “equipo de desarrollo”, que está conformado por Miguel Pino y Daniel Carrión. Desde la perspectiva de las personas no involucradas, los roles de “usuarios” y “manager” serán realizadas por los tres nombres mencionados con anterioridad.

## Resultados

### 3. 1 Proceso de desarrollo y conceptualización

El gestor utilizado para la base de datos del proyecto es PostgreSQL (Emc2Net, s.f), un gestor *open source* de herramientas avanzadas. Este gestor puede equipararse a otros grandes gestores como Oracle, SQL Management (Microsoft). PostgreSQL tiene dos grandes ventajas, tiene capacidad para trabajar con grandes cantidades de datos y tiene una licencia dual, en caso de que el gestor sea usado para fines comerciales, se debe realizar un pago, si es un proyecto no comercial, no habría problema con su uso y se descarta un posible pago.

Las características que hacen a este gestor de base de datos tan robusto son:

- Alta concurrencia, tienen la capacidad de atender a varios clientes el mismo tiempo y entregar la información sin ningún tipo de bloqueos.
- Soporte para múltiples tipos de datos de manera nativa, a diferencia de otros gestores ofrecen tipos de datos como ip, mac, decimales con precisión configurable, figuras geométricas, entre otros.
- Soporte a triggers, generan acciones cuando se disparan.
- Vistas, se consultan datos de manera diferente a como se almacenan.
- Trabaja con sus datos como si fueran objetos.
- Soporta gran cantidad de lenguajes de programación.

Durante este desarrollo también se hizo la inclusión de Visual Studio, un programa realizado por Microsoft para el desarrollo de plataformas web, móvil y aplicaciones de escritorio. Parte de Visual Studio es Xamarin, y es la plataforma que se va a utilizar para la elaboración de la aplicación (Microsoft, s.f). Xamarin es una plataforma de gran rendimiento para creación de aplicaciones móviles en sistemas operativos de Apple (iOS), Google (Android) y Windows phone con “.net”. La aplicación puede ser desarrollada en Windows o Macintosh.

Por otra parte, Git es un repositorio y sistema de control de versiones, fue desarrollado por Linus Torvalds, la licencia de github es libre. Esta herramienta dispone de herramienta de línea de comandos llamada “git bash” y de escritorio (Blischak et al., 2016).

GitHub es un servicio comercial de alojamiento de repositorios Git, fue creado en el año 2008. GitHub tiene una interfaz web. Permite al usuario crear repositorios vacíos o por clonación de otro repositorio de GitHub, enviar solicitudes de cambio entre repositorios que estén en GitHub, los repositorios creados son de acceso público por defecto, solamente si se contrata la versión

de paga y/o si cumplen ciertos requisitos, se puede tener como privado el repositorio del código. La herramienta proporciona un completo sistema de gestión de comentarios a cada repositorio, cuadro de control con grafos sociales, inclusive una página web propia (Blischak et al., 2016).

Laravel es un *framework* generador de código donde se trabaja con modelo, controlador en donde se implementan rutas para la comunicación de red entre aplicación. Esto facilita la construcción del *backend* de la aplicación móvil “peluquéate”, dado que se manejan múltiples vistas y poder controlar cada una de ella, solo con un lenguaje nativo como Php representaría un incremento en el tiempo de desarrollo y de entrega (u/wjgilmore2014, 2015).

Entendiendo la importancia del manejo de todos los servicios que se requieren para poder implementar una aplicación de consumo a nivel usuario, se ha decidido trabajar en un entorno Linux (Loshin, 2021), en este caso, Ubuntu 20.04 *long time supports*, en lugar de un virtualizador o ejecutor Linux como XAMPP, para esto hay que recordar el IDE de la aplicación es Visual Studio 2022, el mismo que no corre en un servidor Linux, a solución de esto se implementó el uso de la última característica de Windows 10 y 11 llamada WLS – “windows subsystem for linux”.

Esto permite tener todo el Kernal de Linux mediante una terminal que comparte los recursos de Windows, sin tener afectación o choques de red ni de rendimiento.

Esta característica impulsada por Windows, permitió que se trabajara en dos entornos al mismo tiempo de forma armonizada, usando Linux para la gestión de servidores (apache 2) y Windows para el desarrollo de la aplicación.

Php, siendo un lenguaje de programación enfocado al desarrollo web, tiene características especiales que permite la gestión y control de los procesos de consulta entre el *frontend* y la base de datos, por ello se ha decidido utilizarlo como lenguaje principal de gestión *backend*, conectando entre controladores, modelo y, siendo un puente entre la base de datos y la aplicación móvil que actualmente está desarrollada en Xamarin - C# (PHP, s.f).

Como el flujo de desarrollo ideal es poder trabajar en paralelo lo apartado visual con el control, se ha optado por usar Postman para discriminar la necesidad de las vistas al momento del desarrollo del *backend*; es decir, esto permitió asegurar que las conexiones y los controles estén funcionando correctamente, sin la necesidad de depender del *frontend* y retrasar los sprints (Postman, s.f).

Entorno de pruebas de conexión con la base de datos:

Traer los datos (*Get*), registrar (*Post*), actualizar (*Put*), eliminar (*Delete*).

Por su parte, ZeroTier-One es una herramienta que permite crear una red privada virtual, ya que como no se tienen servidores remotos por el costo, se ha decidido montar una laptop como servidor, conectado a esta red virtual y así acceder a la aplicación en cualquier parte del mundo que tenga la aplicación y los permisos de acceso a esta red (Curman, 2021).



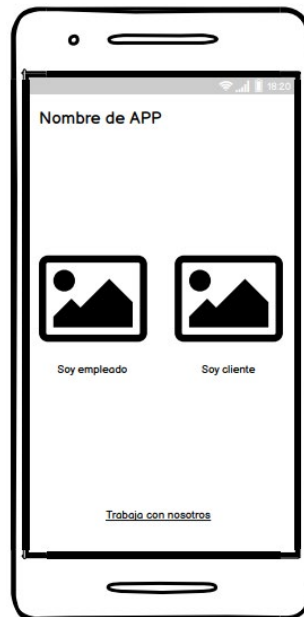
Esto permitió también realizar pruebas desde un teléfono Android físico, ya que ZeroTier es multiplataforma entre Linux, Windows, Android e IOS.

A continuación, se dará a conocer el proceso que se siguió para programar la aplicación. Esto se explicará en “Sprints”, de acuerdo con la metodología elegida para el desarrollo.

- **Sprint Cero:** se tuvo una breve reunión en la cual se dieron a conocer los objetivos, el proceso a realizar y la organización que se daría a lo largo del proyecto. Se definió el gestor de datos con el cual se trabajaría, la plataforma de desarrollo con sus respectivas librerías, el programa para la modulación de datos.
- **Primer Sprint:** se realizó el prototipo de las primeras vistas con Balsamiq (*Figura 1*), en paralelo se realizaba el diseño de la base de datos de las tablas que intervendrían en dichas vistas. El encuentro tuvo lugar digitalmente a través de la herramienta “meet” de Google en la semana del 23 de octubre de 2021.

### Figura 1

*Prototipo de la primera vista*



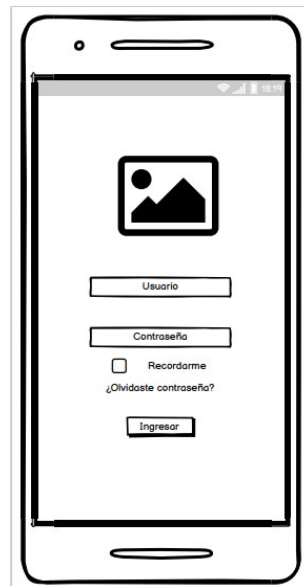
Fuente: Balsamiq

- **Segundo Sprint:** en el segundo encuentro se siguió con la definición de las tablas de la base de datos, siguiendo las pautas de las vistas (*Figura 2*). El encuentro tuvo lugar digitalmente a través de la herramienta “meet” de Google en la semana del 30 de octubre de 2021.



**Figura 2**

*Prototipo de la segunda vista*

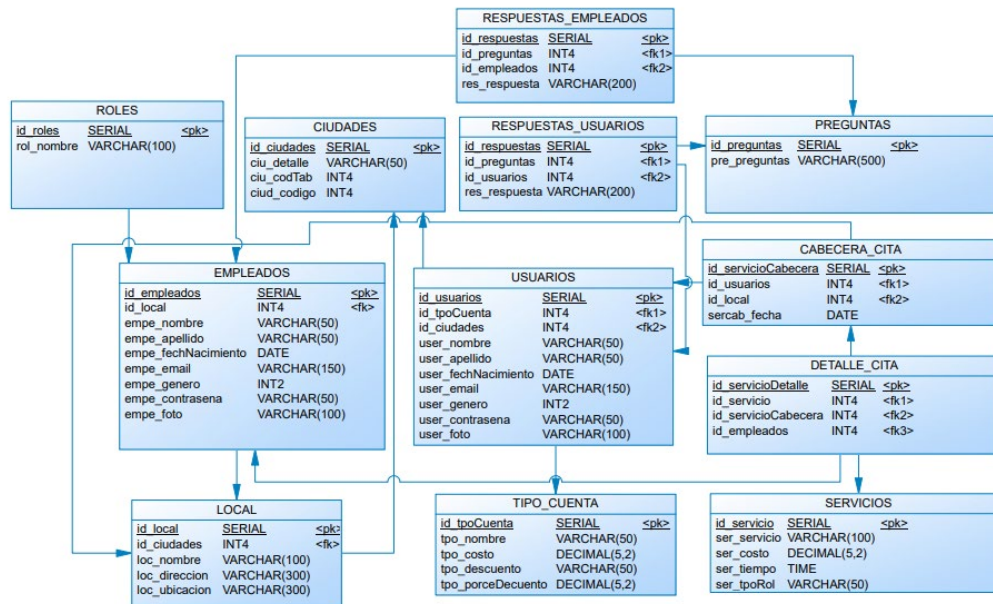


**Fuente:** Balsamiq

- **Tercer Sprint:** en esta reunión se finalizó el diseño de la base de datos, además del prototipo de todas las vistas para empezar el desarrollo en la plataforma Visual Studio (*Figura 3*). El encuentro tuvo lugar digitalmente a través de la herramienta “meet” de Google en la semana del 06 de noviembre de 2021.

**Figura 3**

Primera versión de la base de datos

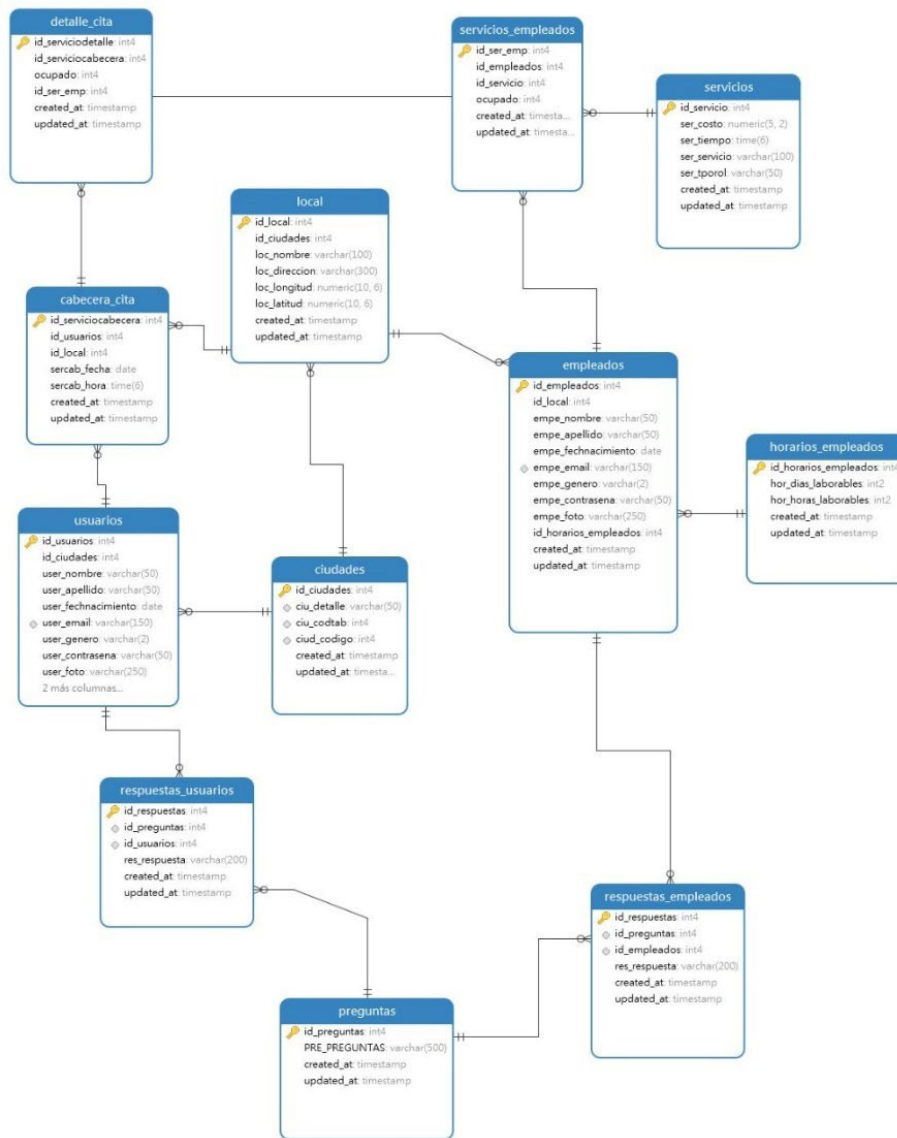


Fuente: Power Designer

- Cuarto Sprint:** desde el tercer Sprint, se realizó el *front end* de la aplicación, se estableció una reunión para definir los detalles del diseño y la conexión entre ventanas en la programación del *back end* y la base de datos desde la computadora de uno de los integrantes del equipo, la computadora actuará como servidor de la aplicación. El encuentro tuvo lugar digitalmente a través de la herramienta “meet” de Google en la semana del 13 de noviembre de 2021.
- Quinto Sprint:** en el penúltimo Sprint se corrigieron algunos detalles en el diseño del *front end* y la base de datos tuvo cambios. Finalmente, en la *Figura 4* se encuentra el diseño físico de la base de datos. El encuentro tuvo lugar digitalmente a través de la herramienta “meet” de Google en la semana del 20 de noviembre de 2021.

Figura 4

Versión definitiva del diseño de la base de datos



Fuente: PostgreSQL

- **Sexto Sprint:** se verificó el producto terminado y completamente funcional. Los pormenores de la aplicación se mostrarán en las gráficas siguientes.

### 3.2 Proceso de agendamiento

Desde la aplicación, para poder realizar el agendamiento de una cita, se hará lo siguiente:

1. El usuario descarga la aplicación en su dispositivo Android (la aplicación será desarrollada para el sistema operativo en mención, en su fase de prototipo).

2. Se debe registrar en la aplicación. Si está ya registrado, iniciar sesión con sus credenciales.
3. Para registrarse debe especificar su: nombre, apellido, correo, fecha de nacimiento, provincia y ciudad de residencia, género y crear una contraseña (el usuario será el correo electrónico ingresado). Por último, especificar tres preguntas de seguridad con sus respectivas respuestas, éstas servirán para generar una nueva contraseña en el escenario que el cliente la haya olvidado.
4. Una vez iniciada la sesión se podrá visualizar la sección destinada a publicidad y nos otorgará la opción para agendar la cita en la peluquería que se desee.
5. En la parte superior izquierda, se podrá ver tres líneas paralelas horizontales, a través de ésta se direccionará a un menú desplegable que contiene las siguientes opciones: mi perfil, mis citas, agendar citas, acerca de y cerrar sesión. A través de dicho menú también se accederá a la vista para agendar una cita.
6. Dentro de la vista “Nueva cita” se va a seleccionar la sucursal de su conveniencia. En la opción “Ver Mapa” se direccionará a la aplicación “Google Maps” para mostrar la ubicación exacta de la sucursal.
7. En la misma vista seleccionar la hora y la fecha en la que se desea agendar la cita.
8. En la vista siguiente se debe seleccionar el servicio al que se desea acceder y el colaborador con el cual se tendrá la cita. Se puede seleccionar cuantos servicios desee. Después de elegir cada servicio en la parte inferior se mostrará el valor de cada uno.
9. Para finalizar, se direccionará a la vista “Mis citas”, donde se mostrará la cita que se acaba de gestionar.
10. El peluquero recibirá una notificación a la hora de la cita que le mostrará la información detallada.

### 3.3 Realizado sin la aplicación por parte del usuario

Este escenario está sesgado a las personas que no utilizan la aplicación. El cliente llegará a la peluquería, solicitará una cita y la persona que se encuentre en la recepción de la sucursal agendará la cita mediante la página web, ya que la aplicación administrativa dispone de la opción de agendar citas sin pasar por la aplicación cliente. La cita se agendará a nombre del cliente.

### 3.4 Pantallas de la aplicación

A continuación, se verán las pantallas de la aplicación. El *splash screen* es una pantalla de carga que permite dar a conocer al usuario sobre la marca del aplicativo entre colores y logotipo (Figura 5).

**Figura 5**

*Pantalla de ingreso a la aplicación*



De acuerdo con la *Figura 6*, como en la construcción de la base de datos, se está discriminando los usuarios de los empleados, usando una pantalla de pre login.

**Figura 6**

*Pantalla para escoger el tipo de usuario*



La ventana de login permite el ingreso de los usuarios con su cuenta personalizada y en caso de no tener cuenta, se pueden registrar, como se muestra en la *Figura 7*.

**Figura 7**

*Pantalla de cliente para ingresar las credenciales del usuario*



Como se observa en la *Figura 8*, en el formulario de registro se ha controlado la selección de ciudades por provincia, esto permite que solo se muestren las ciudades, de acuerdo con la provincia seleccionada por el usuario, ahorrando tiempo de búsqueda.

**Figura 8**

*Pantalla de cliente para el registro en el sistema*



17:52 3.9KHz

PELUQUEATE

### ¡Regístrate!

Muestra Uisrael

Fecha de nacimiento: 23/11/2021

muestra.uisrael@google.com

Provincia: Guayas

Ciudad: Seleccione ciudad

Género: Seleccione género

Contraseña

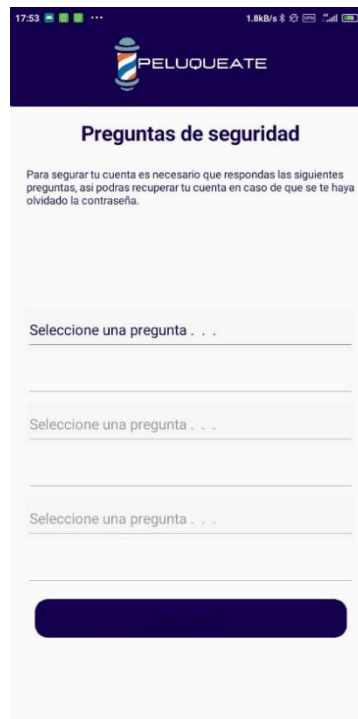
Repetir contraseña

SIGUIENTE

Con las preguntas y respuestas se asegura el método de recuperación de contraseña, esto seleccionando tres preguntas de un banco de seis (*Figura 9*).

**Figura 9**

*Pantalla de cliente para completar el registro en el sistema con las preguntas de seguridad*



17:53 1.8KHz

PELUQUEATE

### Preguntas de seguridad

Para asegurar tu cuenta es necesario que respondas las siguientes preguntas, así podrás recuperar tu cuenta en caso de que se te haya olvidado la contraseña.

Seleccione una pregunta . . .

Seleccione una pregunta . . .

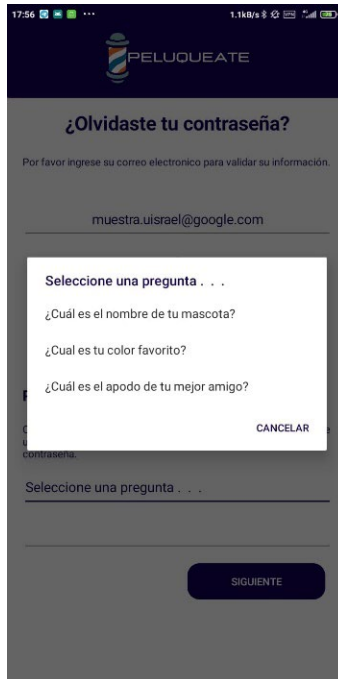
Seleccione una pregunta . . .

[Botón de confirmación]

Lo importante de estas vistas es que una vez seleccionado una pregunta (*Figura 10*), no se mostrará más para que escoja otra.

### Figura 10

*Pantalla de cliente generar una nueva clave ingresando el correo y respondiendo una pregunta*

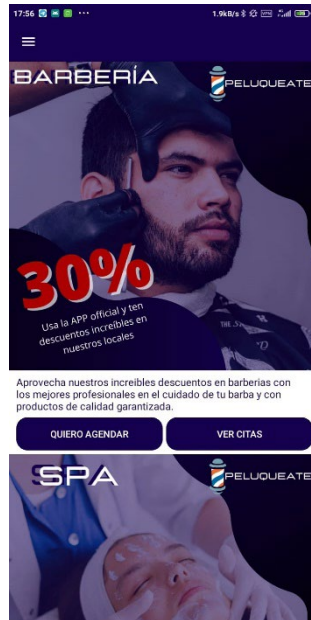


Una vez ingresadas las credenciales se muestra el “Home”, como se ve en la *Figura 11*, el cual fue pensando específicamente para promocionar cada uno de los servicios de las peluquerías.



**Figura 11**

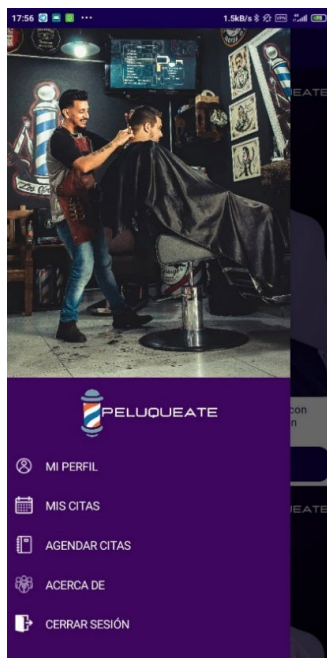
*Pantalla posterior del ingreso exitoso*



De acuerdo con la Figura 11, se cuenta con un menú desplegable, donde están las opciones de: mi perfil, mis citas, agendar citas, acerca de y cerrar sesión

**Figura 12**

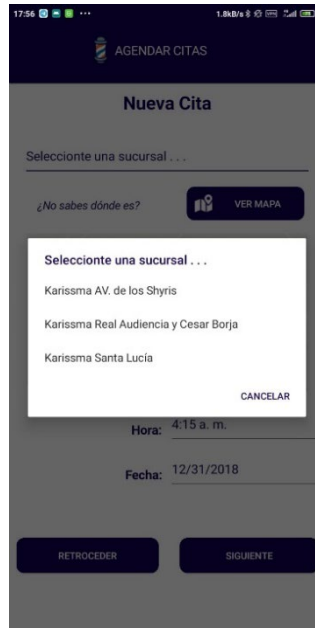
*Pantalla que muestra las opciones una vez iniciada sesión*



Para agendar una cita se cuenta un listado de todas las sucursales vigentes (*Figura 13*), dependiendo a la provincia y ciudad donde el usuario se había registrado, esto permite acceder al usuario al Google Maps, en caso no conocer cómo llegar.

**Figura 13**

*Pantalla que muestra las pautas para agendar una cita*



Como se observa en la *Figura 14*, esta vista lo que hace es mostrar la información registrada de cada usuario, teniendo la opción de modificar ciertos datos, en caso de ser necesario por el usuario, accediendo a la galería o cámara del celular, si es que el usuario desea colocar una foto de perfil, en vez del predeterminado un avatar.

**Figura 14**

*Pantalla que muestra el perfil del cliente y permite editarlo*



En caso de ser un empleado, cambiará la forma de presentación de los datos, mostrando, en este caso, la sucursal a la cual el empleado pertenece, esto permite que en un futuro este pueda ser reubicado en cualquier otra sucursal vigente, dentro del sistema

**Figura 15**

*Pantalla que muestra el perfil del empleado y permite editarlo*



## Conclusiones

El desarrollo móvil multiplataforma representa un gran crecimiento a nivel de mercado, por lo que se puede abarcar un mayor número de dispositivos de diferentes sistemas operativos, permitiendo que la aplicación desarrollada alcance un número considerable de usuarios, que validarán o rechazarán la aplicación.

Xamarin Form, siendo una tecnología *open source* soportada por Microsoft, muestra una increíble capacidad de desarrollo por la cantidad de paquetes y recursos que se pueden utilizar para desarrollar aplicaciones, de acuerdo con la necesidad del cliente o idea de negocio. La programación es una recopilación de varios conceptos afianzados a lo largo de la carrera, como programación orientada a objetos, ing. software y bases de datos, todo con el fin de construir un producto sostenible y con buenas prácticas en el entorno de la tecnología.

La aplicación “Peluquéate” se encuentra en su fase inicial, los resultados experimentales demuestran que los negocios (peluquerías) consolidados, y que manejan cantidad masiva de clientes, se podrían beneficiar con la automatización en su proceso de agendamiento de citas. No obstante, para complementar los resultados de la fase experimental, y conocer si es factible la implementación del software en mención para cada empresa, se debe realizar un estudio de viabilidad.

## Referencias

- Amaya Balaguera, Y. D. (2015). Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles. Estado actual. *Revista de Tecnología*, 12(2). <https://doi.org/10.18270/rt.v12i2.1291>
- Artieda Guachamin, C. A., Reza Morales, J. S., & Rojas Cisneros, H. D. (2021). Sistema móvil para la oferta de recorridos con rutas fijas. *REVISTA ODIGOS*, 2(1), 21–40. <https://doi.org/10.35290/ro.v2n1.2021.381>
- Blischak, J. D., Davenport, E. R., & Wilson, G. (2016). A Quick Introduction to Version Control with Git and GitHub. *PLOS Computational Biology*, 12(1), e1004668. <https://doi.org/10.1371/journal.pcbi.1004668>
- Burbano Ulloa, E. A., Constante Molina, M. A., Hidalgo Guamán, L. M., & Moya Chiluiza, F. A. (2020). Prototipo móvil para la geolocalización de mascotas callejeras. *REVISTA ODIGOS*, 1(3), 77–96. <https://doi.org/10.35290/ro.v1n3.2020.372>
- Curman, M. (2021). *Automatizirani sustav prikupljanja podataka o objektima održavanja primjenom IIoT senzora*. [Tesis doctoral, University of Zagreb]. <https://urn.nsk.hr/urn:nbn:hr:235:485182>
- Emc2Net. (s.f). PostgreSQL-ES. <https://e-mc2.net/es/postgresql-es>
- Florez, C., Quevedo, W., Galora, F., & Toasa, R. (2021). Performance of WebGL standard for displaying 3D applications on mobile devices. In *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1-6, IEEE. <https://doi.org/10.23919/CISTI52073.2021.9476391>
- González, E. (2005). La observación directa base para el estudio del espacio local. *Geoenseñanza*, 10(1), 101-105. <https://www.redalyc.org/pdf/360/36010107.pdf>
- Leung, T. (2015). Creating Desktop Applications. In *Visual Studio LightSwitch 2015*, pp. 73–109. Apress. [https://doi.org/10.1007/978-1-4842-0766-6\\_4](https://doi.org/10.1007/978-1-4842-0766-6_4)
- Loshin, P. (octubre, 2021). *Linux operating system*. <https://searchdatacenter.techtarget.com/definition/Linux-operating-system>
- Luzardo Cabrera, A. (2014). *Aplicación Android para pedir cita previa en peluquerías*. [Tesis de grado, Universidad de las Palmas de Gran Canaria]. [https://accedacris.ulpgc.es/bitstream/10553/20974/1/0727263\\_00000\\_0000.pdf](https://accedacris.ulpgc.es/bitstream/10553/20974/1/0727263_00000_0000.pdf)
- Microsoft. (s.f). *Visual studio*. <https://visualstudio.microsoft.com/es/>
- Montero, B. M., Cevallos, H. V., & Cuesta, J. D. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales Revista Multidisciplinaria de Investigación*, 2(17). <https://doi.org/10.31876/RE.V2I17.269>

- Nagy, P. G., Konewko, R., Warnock, M., Bernstein, W., Seagull, J., Xiao, Y., George, I., & Park, A. (2008). Novel, web-based, information-exploration approach for improving operating room logistics and system processes. *Surgical Innovation*, 15(1), 7–16. <https://doi.org/10.1177/1553350608316573>
- Nuñez, A. v., & Nuñez, L. N. (2020). Reconocimiento facial para el encendido automático de vehículos basado en Raspberry Pi. *REVISTA ODIGOS*, 1(2), 53–68. <https://doi.org/10.35290/RO.V1N2.2020.326>
- PHP. (s.f). ¿Qué es PHP? <http://php.net/manual/es/intro-what-is.php>
- Postman. (s.f). API Platform. <https://www.postman.com/>
- Reyes, J., Morales, L., Aldas, D., Reyes, R., & Toasa, R. (2019). Real Time Automatic Andon Alerts for Android Platforms Applied in Footwear Manufacturing. In *Computer and Communication Engineering*, pp. 43–56. [https://doi.org/10.1007/978-3-030-12018-4\\_4](https://doi.org/10.1007/978-3-030-12018-4_4)
- Reyes Rivera, A., Vallejos Cango., & Quintana García, D. (2020). Nueva realidad: compras en línea y a domicilio. *REVISTA ODIGOS*, 1(3), 47–60. <https://doi.org/10.35290/ro.v1n3.2020.370>
- Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). The Secret History of Agile Innovation. *Harvard Business Review*, 4.
- San Martin Galaz, M. (2005). *Sistema de gestión de clientes para Peluquería Nuevo Estilo*. [Tesis de grado, Universidad de Talca]. <http://dspace.otalca.cl/handle/1950/5230>
- Sangama **Oñate, A.** (2020). *Metodologías ágiles Scrum, XP, SLeSS, Scrumban, HME, Mobile-D y MASAN empleadas en la industria de dispositivos móviles: Un contraste en favor de la industria del desarrollo móvil*. [Tesis de grado, Universidad Peruana Union] <http://hdl.handle.net/20.500.12840/3906>
- u/wjgilmore2014. (23 de julio de 2015). *How Laravel 5 Prevents SQL Injection, Cross-Site Request Forgery, and Cross-Site Scripting*. EasyLaravelbook. <https://www.easylaravelbook.com/blog/how-laravel-5-prevents-sql-injection-cross-site-request-forgery-and-cross-site-scripting/>

Copyright (c) 2022 Miguel Angel Pino Andrade, Arthur Wilson Chávez Mora y Juan Daniel Carrion Robles



Este texto está protegido bajo una licencia internacional [Creative Commons](#) 4.0.

Usted es libre para Compartir—copiar y redistribuir el material en cualquier medio o formato — y Adaptar el documento — remezclar, transformar y crear a partir del material—para cualquier propósito, incluso para fines comerciales, siempre que cumpla las condiciones de Atribución. Usted debe dar crédito a la obra original de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios.

Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace de la obra.

[Resumen de licencia](#) – [Texto completo de la licencia](#)