



UNIVERSIDAD TECNOLÓGICA DE CHETUMAL

**PROYECTO:**

***“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO”***

**MEMORIA DE ESTADÍA PROFESIONAL  
PARA OBTENER EL TÍTULO DE:**

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA INFORMACIÓN  
ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

**PRESENTA:**

**JESÚS ALBERTO GÓMEZ HOIL  
GENERACIÓN 2019-2021**

**EMPRESA: TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO**

**ASESOR DE LA ORGANIZACIÓN:  
LIC. ISRAEL DAMIÁN UC AVILÉS**

**ASESOR ACADÉMICO:  
LIC. CARLOS ALBERTO VILLANUEVA  
CERVERA**

**CHETUMAL, QUINTANA ROO, AGOSTO 2021**



UNIVERSIDAD TECNOLÓGICA DE CHETUMAL

**PROYECTO:**

**“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL SUPERIOR DE  
JUSTICIA DE QUINTANA ROO”**

**MEMORIA DE ESTADÍA PROFESIONAL  
PARA OBTENER EL TÍTULO DE:**

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA  
INFORMACIÓN ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

**PRESENTA:**

**JESÚS ALBERTO GÓMEZ HOIL  
GENERACIÓN 2019-2021**

**EMPRESA: TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO**

**ASESOR DE LA ORGANIZACIÓN:  
LIC. ISRAEL DAMIÁN UC AVILÉS**

**ASESOR ACADÉMICO:  
LIC. CARLOS ALBERTO  
VILLANUEVA CERVERA**

**CHETUMAL, QUINTANA ROO, AGOSTO 2021**

## OFICIO DE APROBACIÓN DEL EMPASTADO

Chetumal, Quintana Roo, a 13 de agosto de 2021

**C. JESUS ALBERTO GOMEZ HOIL**  
ALUMNO DE LA CARRERA DE TÉCNICO SUPERIOR UNIVERSITARIO EN  
TECNOLOGÍAS DE LA INFORMACIÓN ÁREA DESARROLLO DE SOFTWARE  
MULTIPLATAFORMA  
GENERACIÓN 2019-2021  
PRESENTE

El informe de Estadía (MEMORIA), con el título “***App móvil Tribunal Familiar Digital del Tribunal Superior de Justicia de Quintana Roo***” que se desarrolló en la empresa “**Tribunal Superior de Justicia de Quintana Roo**” cumple con lo señalado en el Art. 12, fracción XV del Reglamento de Visitas y Estadías de la Universidad Tecnológica de Chetumal, por lo que se le autoriza el empastado de dicho informe teniendo como plazo de entrega 10 días hábiles a partir de la recepción de este documento.

**ATENTAMENTE**

**M.T.I. MARTÍN ANTONIO SANTOS ROMERO**  
**DIRECTOR DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y**  
**COMUNICACIÓN**

C.C.P. DIRECCIÓN ACADÉMICA  
C.C.P. EXPEDIENTE

Universidad Tecnológica de Chetumal Teléfono: (983) 12917 65 Ext. 1014 Correo Electrónico: [vinculacion@utchetumal.edu.mx](mailto:vinculacion@utchetumal.edu.mx)  
Camino antiguo a Santa Elena, S/N, Colonia Centro, C.P. 77000, Chetumal Quintana Roo.  
Referencia: Carretera Chetumal-Bacalar Km 5.3

**Organismo Público Descentralizado del Gobierno del Estado de Quintana Roo**  
**F-VIN-009 REV.05**  
**09/01/2018**



## CESIÓN DE DERECHOS

Con fundamento en los artículos 21 y 27 de la Ley Federal del Derecho de Autor y como titular de los derechos moral y patrimoniales de la obra titulada:

**“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL SUPERIOR DE  
JUSTICIA DE QUINTANA ROO”**

que se presenta a la Universidad Tecnológica de Chetumal (UT Chetumal) como defensa para obtener el título de Técnico Superior Universitario en Tecnologías de la Información Área Desarrollo de Software Multiplataforma, otorgo de manera gratuita y permanente a dicha institución y a las bibliotecas de sus sedes, autorización para que fijen la obra en cualquier medio, incluido el electrónico, y la divulguen entre sus usuarios, profesores y estudiantes, sin que pueda percibir por tal divulgación una contraprestación.

Chetumal, Quintana Roo, a 04 de mayo de 2021

**C. JESÚS ALBERTO GÓMEZ HOIL**

## DEDICATORIA

Dedico esta memoria de estadía profesional a mi madre, Liliana Rubí Hoil Alonzo, y a mi abuela, Ligia Yolanda Alonzo Briceño, porque han sido las personas que más apoyo me han brindado a lo largo de mi vida para salir adelante y cumplir todos mis sueños, y las cuales nunca han dejado de confiar en mí, sin importar las circunstancias en las que me he encontrado.

A mi madre porque es una de las personas más importantes en mi vida y de no ser por ella yo no estaría aquí y estoy muy agradecido por todo el cariño y amor que me ha dado. Mi madre ha sido de las personas que me ha aconsejado sobre las decisiones que debo de tomar en mis estudios y gracias a dichos consejos estoy en un punto clave de mi vida personal y profesional.

Por otro lado, a mi abuela porque ella me ha apoyado tanto en la parte emocional como en la parte económica, y gracias a ella pude continuar con mis estudios, estuvo acompañándome en mis noches de desvelo, se preocupó de mi salud y de mi sueño, y estuvo al pendiente de mí y de mi mamá. Es una persona maravillosa a la cual le debo todo.

A las dos las amo con todo el corazón y siempre estaré muy agradecido por todo lo que han hecho por mí.

## AGRADECIMIENTOS

Agradezco a mi novia, Aline Aide Suárez Sosa, que ha sido de las personas que me ha brindado su apoyo en todo momento, con la que paso momentos increíbles cada vez que nos vemos y sé que siempre podré contar con ella para todo. Gracias por lo que me has enseñado, a nunca dejar todo y a nunca rendirme en mi carrera, te amo demasiado.

A mi hermana Alejandra Guadalupe Gómez Hoil, a quien admiro por ser el mayor ejemplo de lo que es ser una persona trabajadora y responsable, referente para poder terminar mi carrera de TSU. Gracias a ella he podido dar mi mejor esfuerzo en las cosas que hago y sé que puedo contar con ella cada vez que tenga algún problema.

A mis tíos(os) y primas(os), que son importantes en mi vida, porque sé que ellos me dan apoyo en los momentos que lo necesito y siempre me sacan una sonrisa al momento de realizar mis cosas.

Agradezco a mis mejores amigos David Bustillos y Brayan Espinoza, por brindarme su apoyo cuando más lo necesitaba, por ayudarme con algunas ideas para mi proyecto cuando no podía pensar en nada, por estar a mi lado y no dejarme solo en ningún momento de mi vida.

También, agradezco a todos los amigos que he conocido a lo largo de mi vida, por brindarme su amistad y por darme el apoyo que necesito en todos los tropiezos que doy, sé muy bien que puedo confiar en ellos en cualquier momento. De igual forma, agradezco a todos los maestros que he tenido por brindarme todo su conocimiento y consejos que me pueden servir para toda la vida, ya sea personal o profesional. Personalmente, agradezco al profesor Carlos Villanueva por todo el apoyo y conocimiento que me ha aportado durante mis estadías.

## ÍNDICE

<b>INTRODUCCIÓN .....</b>	<b>1</b>
---------------------------	----------

### **CAPÍTULO I PLANTEAMIENTO DEL PROBLEMA**

#### ***“LA IMPORTANCIA DEL DESARROLLO MÓVIL”***

1.1 Situación problemática.....	3
1.2 Objetivos.....	5
1.2.1 Objetivo general.....	5
1.2.2 Objetivos específicos.....	5
1.3 Justificación.....	5
1.4 Limitaciones.....	7

### **CAPÍTULO II MARCO CONTEXTUAL**

#### ***“TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO”***

2.1 Antecedentes.....	9
2.2 Descripción general de la organización.....	10
2.2.1 Organigrama.....	11
2.3 Ubicación.....	11
2.4 Misión.....	12
2.5 Visión.....	12
2.6 Valores.....	12
2.7 Objetivo general.....	13
2.8 Objetivos específicos .....	13

### **CAPÍTULO III MARCO TEÓRICO**

#### ***“FUNDAMENTOS PARA EL DESARROLLO MÓVIL”***

3.1. Concepto de desarrollo de aplicaciones móviles.....	15
3.2. La evolución de las aplicaciones móviles en las empresas...	16

3.3. Diferencia entre aplicaciones web y móviles.....	17
3.4. Tecnologías para el desarrollo móvil.....	17
3.4.1. Android Studio.....	18
3.4.2. Apache.....	18
3.4.3. API.....	19
3.4.4. Dart.....	19
3.4.5. Flutter.....	20
3.4.6. JSON.....	20
3.4.7. MySQL.....	20
3.4.8. Postman .....	21
3.4.9. Programación SQL.....	22
3.4.10. Visual Studio Code.....	22
3.4.11. Web Services.....	23
3.4.12. XAMPP.....	24
3.5. Metodologías.....	24
3.5.1. Metodología Scrum.....	25
3.5.2. Metodologías para el desarrollo móvil.....	26
3.6. Casos de éxito.....	27
3.6.1. Tribunal Virtual Móvil.....	27
3.6.2. Tribunal Móvil BCS.....	28

#### **CAPÍTULO IV DESARROLLO DEL PROYECTO**

##### **“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO”**

4.1. Actividades realizadas.....	31
4.2. Cronograma de actividades.....	32
4.3. Metodología aplicada.....	33
4.4. Material, equipo y herramientas utilizadas.....	35
4.5. Propuesta.....	38
4.5.1. Requerimientos.....	38

4.5.2. Diseño del sistema.....	39
4.5.3. <i>Sprints</i> .....	40
4.5.4. Diagrama PAD de la aplicación móvil.....	46
4.5.5. Diagrama de caso de uso de la aplicación móvil....	47
4.5.6. Diseño de la base de datos.....	48
4.5.7. Mapa de sitio de la aplicación móvil.....	48
4.5.8. Bosquejos del sistema.....	49
4.5.9. Diagrama tecnológico de la aplicación móvil.....	64
4.5.10. Capturas de pantalla de la aplicación móvil.....	64
<b>CONCLUSIÓN.....</b>	<b>79</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>81</b>
<b>ANEXOS.....</b>	<b>84</b>

## **LISTA DE TABLAS Y FIGURAS**

<b>Figura 1.</b> Ubicación del Tribunal Superior de Justicia de Quintana Roo..	11
<b>Figura 2.</b> Procesos de la metodología Scrum.....	25
<b>Figura 3.</b> Interfaz de la aplicación móvil Tribunal Virtual Móvil.....	28
<b>Figura 4.</b> Interfaz de la aplicación móvil Tribunal Móvil de BCS.....	29
<b>Tabla 1.</b> Cronograma de actividades para el desarrollo del proyecto.....	32
<b>Tabla 2.</b> Etapas de la metodología Scrum en el proyecto.....	34
<b>Tabla 3.</b> Características de la laptop HP Notebook.....	36
<b>Tabla 4.</b> Características del dispositivo móvil Oppo A12.....	36
<b>Tabla 5.</b> Características de las herramientas de <i>software</i> utilizadas.....	37
<b>Tabla 6.</b> Requerimientos de la app móvil Tribunal Familiar Digital.....	38
<b>Figura 5.</b> Diagrama de clases de la app móvil Tribunal Familiar Digital	40

<b>Tabla 7.</b> Sprint 1 del proyecto.....	41
<b>Tabla 8.</b> Sprint 2 del proyecto.....	42
<b>Tabla 9.</b> Sprint 3 del proyecto.....	44
<b>Figura 6.</b> Diagrama PAD de la app móvil Tribunal Familiar Digital.....	46
<b>Figura 7.</b> Diagrama de caso de uso de la app móvil Tribunal Familiar Digital.....	47
<b>Figura 8.</b> Modelo relacional de la base de datos en MySQL.....	48
<b>Figura 9.</b> Mapa del sitio de la app móvil Tribunal Familiar Digital.....	49
<b>Figura 10.</b> Mockup de la pantalla animada.....	50
<b>Figura 11.</b> Mockup de la pantalla de acciones de ingreso.....	51
<b>Figura 12.</b> Mockup de la pantalla de registro de usuarios.....	52
<b>Figura 13.</b> Mockup de la pantalla de identificación del usuario.....	53
<b>Figura 14.</b> Mockup de la pantalla de confirmación de registro de usuarios.....	54
<b>Figura 15.</b> Mockup de la pantalla de inicio de sesión.....	55
<b>Figura 16.</b> Mockup de la alerta de inicio de sesión.....	56
<b>Figura 17.</b> Mockup de la pantalla del dashboard de la aplicación móvil	57
<b>Figura 18.</b> Mockup de la pantalla del módulo directorio.....	58
<b>Figura 19.</b> Mockup de la pantalla del módulo sentencias públicas.....	59
<b>Figura 20.</b> Mockup de la pantalla del módulo portal.....	60
<b>Figura 21.</b> Mockup de la pantalla del módulo expedientes electrónicos	61
<b>Figura 22.</b> Mockup del menú desplegable.....	62
<b>Figura 23.</b> Mockup de la alerta de cierre de sesión.....	63
<b>Figura 24.</b> Diagrama tecnológico de la aplicación móvil.....	64
<b>Figura 25.</b> Pantalla de presentación animada.....	65
<b>Figura 26.</b> Pantalla de acciones de ingreso.....	66
<b>Figura 27.</b> Pantalla de inicio de sesión.....	67
<b>Figura 28.</b> Pantalla de alerta de inicio de sesión.....	68
<b>Figura 29.</b> Pantalla de registro de usuarios.....	69
<b>Figura 30.</b> Pantalla para tomar foto a la identificación del usuario.....	70

<b>Figura 31.</b> Pantalla de confirmación de registro.....	71
<b>Figura 32.</b> Pantalla del <i>dashboard</i> de la aplicación móvil.....	72
<b>Figura 33.</b> Pantalla del menú desplegable.....	73
<b>Figura 34.</b> Pantalla de alarma de cierre de sesión.....	74
<b>Figura 35.</b> Pantalla del módulo directorio.....	75
<b>Figura 36.</b> Pantalla del módulo sentencias públicas.....	76
<b>Figura 37.</b> Pantalla del módulo portal de citas.....	77
<b>Figura 38.</b> Pantalla del módulo expedientes electrónicos.....	78

## INTRODUCCIÓN

Actualmente, las empresas han tenido la necesidad de implementar tecnologías que faciliten la realización de tareas manuales, empleando dispositivos como *smartphones*, *tablets* o computadoras.

El proyecto "App móvil Tribunal Familiar Digital del Tribunal Superior de Justicia de Quintana Roo" fue realizado como un proyecto de estadía profesional en el periodo mayo-agosto 2021. Este proyecto se llevó a cabo bajo las necesidades de la institución gubernamental que lleva por nombre "Tribunal Superior de Justicia de Quintana Roo". El proyecto tiene como finalidad desarrollar una aplicación móvil que agilice los trámites, procesos y servicios que se encuentren disponibles en la institución de gobierno, con el fin de evitar que las personas realicen sus procesos manualmente. Para el desarrollo del mismo, se realizó un planteamiento del problema, marco contextual, marco teórico y desarrollo del proyecto, en cuatro capítulos de la memoria.

En el capítulo 1, el planteamiento del problema, se describe la situación problemática que se detectó bajo datos estadísticos, al igual que describir los objetivos por los que surge este proyecto, se describen las justificaciones y finalmente, están las limitaciones, por las cuales el proyecto no se podría concluir. El capítulo 2 trata sobre el marco contextual, en donde se describe la información correspondiente a la empresa o institución de gobierno, como lo son antecedentes, descripción de la empresa, organigrama, misión, visión, valores, objetivo general y objetivos específicos.

Por otra parte, el capítulo 3 es el marco teórico donde se incluyen las teorías que soportan la propuesta de solución de la problemática detectada y descrita en el capítulo 1, siendo toda información importante que ayude a realizar el proyecto.

Y, por último, el capítulo 4 es el desarrollo del proyecto, donde se describen las actividades que se llevaron a cabo para la ejecutar el proyecto, así como la metodología aplicada en el proyecto, la descripción de los elementos utilizados para el desarrollo de este y la propuesta con capturas de pantalla de la app móvil.

**CAPÍTULO I**  
**PLANTEAMIENTO DEL PROBLEMA**  
***“LA IMPORTANCIA DEL DESARROLLO MÓVIL”***

**1.1 SITUACIÓN PROBLEMÁTICA**

**1.2 OBJETIVOS**

**1.2.1 OBJETIVO GENERAL**

**1.2.2 OBJETIVOS ESPECÍFICOS**

**1.3 JUSTIFICACIÓN**

**1.4 LIMITACIONES**

**CAPÍTULO I**  
**PLANTEAMIENTO DEL PROBLEMA**  
**“LA IMPORTANCIA DEL DESARROLLO MÓVIL”**

### **1.1. Situación problemática**

Actualmente, el uso de aplicaciones o sistemas informáticos ha evolucionado, desarrollándose para satisfacer las necesidades humanas, prueba de ello es que la mayoría de las empresas cuentan con aplicaciones web (las aplicaciones más comunes dentro de las TIC), pero, para avanzar en esta nueva revolución digital, siempre hay que tener a la mano una aplicación móvil, ya sea nativa, híbrida o multiplataforma, pues se calcula que en México hay 92.01 millones de usuarios de internet (71% de la población), y el dispositivo que más se utiliza para conectarse a la red es el *smartphone* (56.1%), sobre otros dispositivos como *laptops* y ordenadores (42%), *tablets* (1.8%) y consolas de videojuegos, entre otros (0.10%) (Alvino, 2021).

Las aplicaciones móviles son importantes para una empresa o institución, pues:

En un mundo donde el usuario exige inmediatez, simplificación e hiper conectividad, las aplicaciones móviles tienen la capacidad de cubrir las necesidades del cliente actual en cualquier nicho del mercado. De igual modo, son programas tan versátiles que permiten a las empresas desarrollar formas disruptivas de presentar sus servicios al público (Fernández, 2019, párr. 1).

No solo las empresas buscan competir en la revolución digital, sino que las instituciones gubernamentales tienen la necesidad de desarrollar nuevas tecnologías para realizar tareas y trámites de manera sencilla.

Lamentablemente, el uso de aplicaciones móviles realizadas por instituciones gubernamentales en México es bajo, debido a que principalmente se centran en desarrollar páginas web que necesitan de un navegador para ser usadas, limitando su uso principalmente a equipos de cómputo. Esto conlleva una

desventaja, porque las aplicaciones web podrían ser visualizadas fácilmente, no solo desde un navegador web, sino desde un dispositivo móvil o tableta.

Una institución que tiene una problemática relacionada es el Tribunal Superior de Justicia de Quintana Roo, institución gubernamental cuyo propósito es ejercer la jurisdicción y representar al poder judicial en Quintana Roo.

La problemática surge dentro de una de las salas del Tribunal Superior de Justicia de Quintana Roo: La segunda sala familiar, especializada en la materia familiar y materia familiar oral y encargada de servir con base a las apelaciones jurídicas a través de la impartición de justicia (Uc Avilés, comunicación personal, mayo 2021). Su prioridad es brindar certeza a los justiciables, y velar por sus derechos buscando siempre lograr el bienestar y también proteger a los infantes desde el principio del interés superior del menor.

Esta sala cuenta con un micrositio donde se pueden consultar los servicios del Poder Judicial de Quintana Roo, pero el sistema web tiene poca usabilidad, puesto que hoy en día muchos prefieren usar dispositivos móviles en vez de emplear una computadora para acceder a cualquier portal web y servicios. Aunque se puede visualizar la aplicación web en dispositivos móviles usando el navegador, no es tan fácil de emplear y acceder a todas las funciones. Desafortunadamente, esta institución no cuenta con una aplicación móvil donde el usuario pueda entrar y realizar de forma fácil y cómoda todas las consultas y acciones que sean necesarias para este.

La institución es consciente de ello y, de acuerdo con una entrevista hecha al asesor empresarial, el Lic. Israel Damián Uc Avilés, el proyecto de la aplicación se intentó realizar desde el año pasado, pero debido a que no se ha podido encontrar a un desarrollador móvil y por circunstancias como la pandemia y del virus COVID-19, no se pudo llevar a cabo. Sin embargo, aclaró que las ideas siguen siendo las mismas que se plantearon hace un año, como el nombre del proyecto “Tribunal Familiar Digital” (comunicación personal, mayo, 2021). Considerando lo anterior y que el tribunal carece de una actualización de sus sistemas informáticos, se pretende desarrollar el presente proyecto.

## 1.2. Objetivos

### 1.2.1. Objetivo general

Desarrollar la aplicación móvil "Tribunal Familiar Digital", aplicando fundamentos del diseño y desarrollo móvil para hacerla sencilla y funcional, con el fin de realizar trámites a distancia y facilitar el acceso a los servicios que ofrece la segunda sala familiar del Tribunal Superior de Justicia de Quintana Roo.

### 1.2.2. Objetivos específicos

- Definir el proceso de funcionamiento de la aplicación móvil a través del consumo de servicios.
- Diseñar el proceso de funcionamiento de la aplicación móvil en pantallas, utilizando elementos de diseño móvil mediante programación.
- Gestionar usuarios para que puedan iniciar sesión dentro de la aplicación móvil mediante un inicio de sesión o realizando el registro de usuario mediante un formulario.
- Implementar un módulo de reportes que permita al usuario descargar la información de los servicios que solicite.
- Publicar la aplicación móvil para que todos los usuarios con dispositivos móviles que cuenten con el Sistema Operativo Android lo utilicen y realicen trámites vinculados a la institución.

## 1.3. Justificación

Hoy en día, las personas poseen más teléfonos móviles (98.9%) que equipos de cómputo (74.2%) (Alvino, 2021), esto hace que el uso de aplicaciones móviles para satisfacer sus necesidades de forma fácil y rápida sea más común.

Tribunal Familiar Digital será una herramienta que le permitirá a los usuarios

realizar sus trámites y usar los servicios que ofrece la segunda sala familiar del Tribunal Superior de Justicia de Quintana Roo, los cuales son orientados a la materia familiar y a la materia familiar oral, todos estos servicios estarán al alcance de la mano gracias a la implementación de la aplicación móvil de este, haciendo que los procesos sean más ágiles para los usuarios, pues los servicios que ofrecerá esta sala del Tribunal Superior de Justicia de Quintana Roo serán los mismos que se verán reflejados en la aplicación móvil. Además, se pretende agregar funcionalidades como:

- Inicio de sesión y registro de usuarios.
- Directorio del Tribunal Superior de Justicia de Quintana Roo.
- Apartado de portal de citas.
- Apartado de sentencias públicas.
- Apartado de expedientes electrónicos y vinculación de expedientes.

Este proyecto beneficiará tanto al equipo de trabajo de la segunda sala familiar del tribunal, como a los ciudadanos que deseen realizar sus trámites.

Para los ciudadanos, el uso de la aplicación móvil agilizará los procesos que se quieran realizar por parte del ciudadano, permitiendo que los trámites se realicen a distancia (situación conveniente actualmente por la pandemia de COVID-19), ahorrando tiempo y esfuerzo al usuario, pues ya no tendrá que desplazarse hasta la institución para atender sus necesidades.

A la segunda sala familiar le beneficiará porque podrán actualizar su manera de trabajar con los sistemas informáticos dentro de esta era tecnológica, permitiendo la innovación. Pero no solo esto, sino que podrán tener la certeza de que su ritmo de trabajo y la manera de llevar sus procesos será más ágil y rápida, ya que podrían atender una mayor cantidad de usuarios en menos tiempo, eficientando los procesos e incluso usando menos recursos para atender las necesidades de los usuarios.

Asimismo, la aplicación puede servir para difundir los servicios de la institución, sirviendo no solo para promoción de esta, sino para difusión y que así más personas puedan acercarse al tribunal para ser atendidas de forma eficiente.

#### 1.4. Limitaciones

Las siguientes limitaciones podrían poner en riesgo la finalización de este proyecto:

- Tiempo: El tiempo de las estadías es de 4 meses, esto es una limitación importante porque desarrollar una aplicación móvil funcional, eficiente, intuitiva y rápida puede llegar a ser una labor difícil, en cuanto al entorno de desarrollo que se vaya a manejar y a las herramientas de *software* que se utilizarán en estos 4 meses.
- Entorno de desarrollo: Desarrollar una aplicación móvil mediante el uso de un *framework* puede ser para muchos una ventaja a la hora de querer reducir la carga de trabajo en la programación de aplicaciones. No obstante, trabajar con un *framework* puede ser también una desventaja por los requerimientos que la institución necesitará para la funcionalidad de un aplicativo móvil.
- Contingencia COVID-19: Un factor significativo para que el proyecto no se concluya sería la pandemia que azota a todo el mundo, el COVID-19. Esta limitación se justifica porque este virus hace difícil salir del hogar sin exponerse. Es claro que el proyecto se podría realizar desde casa, pero sería complicado tener comunicación con los asesores y de las observaciones que se deben de hacer para la aplicación móvil por parte de estos.

**CAPÍTULO II**  
**MARCO CONTEXTUAL**  
***"TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO"***

- 2.1 ANTECEDENTES**
- 2.2 DESCRIPCIÓN DE LA EMPRESA**
  - 2.2.1 ORGANIGRAMA**
- 2.3 UBICACIÓN**
- 2.4 MISIÓN**
- 2.5 VISIÓN**
- 2.6 VALORES**
- 2.7 OBJETIVO GENERAL**
- 2.8 OBJETIVOS ESPECÍFICOS**

**CAPÍTULO II**  
**MARCO CONTEXTUAL**  
**“TRIBUNAL SUPERIOR DE JUSTICIA DE QUINTANA ROO”**

La información de este capítulo fue obtenida mediante el uso del sitio web del Tribunal Superior de Justicia de Quintana Roo (2021), y proporcionada por el asesor empresarial, Lic. Israel Damián Uc Avilés.

### **2.1. Antecedentes**

El Tribunal Superior de Justicia de Quintana Roo se creó en el año 1974, esta institución gubernamental se encarga de ejercer la jurisdicción del estado de Quintana Roo para hacer una representación al poder judicial en el estado, ante los demás poderes que rigen al país. Se concluye que:

El proceso histórico del Tribunal Superior de Justicia del Estado de Quintana Roo se divide en dos etapas que responden a las condiciones y circunstancias históricas de la entidad. La primera es la etapa de territorio federal y la segunda cuando adquiere la figura jurídica del estado libre y soberano. En el primer periodo que abarca de 1902 al 1974, las leyes que rigieron al Territorio de Quintana Roo fueron las mismas que regulan al Distrito Federal, por lo que no existía propiamente un Tribunal Superior de Justicia, las instancias superiores eran las de la capital de la República. Los primeros jueces de paz en la entidad datan de 1903 (Ávila, 2010, párr. 1).

El Tribunal Superior de Justicia de Quintana Roo ha tenido en su historia a 11 presidentes dentro del poder judicial del estado de Quintana Roo.

Desde que se inició el tribunal, ha habido personas que han ocupado el cargo de magistrado. De acuerdo con Ávila (2021) estos son:

- 1974 - 1975 Héctor Olayo Delgado Suárez

- 1975 - 1976 Ligia Minerva Mendoza Argucia
- 1976 - 1977 Felipe Amaro Santana
- 1977 - 1981 Miguel Ángel Angulo Castillo
- 1981 - 1987 Miguel de J. Peyrefitte Cupido
- 1987 - 1993 Celia Pérez Gordillo
- 1993 - 1996 Miguel Mario Angulo Flota
- 1996 - 2000 Joaquín González Castro
- 2000 - 2011 Lizbeth Loy Song Encalada
- 2011 - 2017 Fidel Gabriel Villanueva Rivero (párr. 2).

Actualmente, la persona que ocupa el cargo de presidente del poder judicial de Quintana Roo es el Magistrado José Antonio León Ruiz, quien ocupó el cargo en el año 2017.

## **2.2. Descripción de la empresa**

El Tribunal Superior de Justicia de Quintana Roo es una institución que se basa en el poder judicial, uno de los poderes de la unión de la política de México dentro de un gobierno federal y esta cuenta con una misión importante sobre la resolución de problemas.

Cuenta con una estructura de trabajo que va desde la presidencia, secretaría, ya después de estos departamentos van lo que son las salas que se especializan en diferentes giros del poder judicial la cual lo caracteriza. A continuación, se describirán todas las áreas que conforman a la institución.

Primero que nada, están las oficinas principales como lo son la presidencia y la secretaría general de la presidencia del consejo de la judicatura.

De ahí, el Tribunal Superior de Justicia de Quintana Roo se divide en 9 salas las cuales tienen diferentes especialidades, que serán catalogadas de acuerdo con su especialización en las diversas materias que manejan:

- Primera, Quinta y Sexta sala: especializadas en materia civil y mercantil.

- Segunda y Séptima sala: especializadas en materia familiar y materia familiar oral.
- Tercera y octava sala: especializadas en materia penal tradicional.
- Cuarta y novena sala: especializadas en materia penal oral.

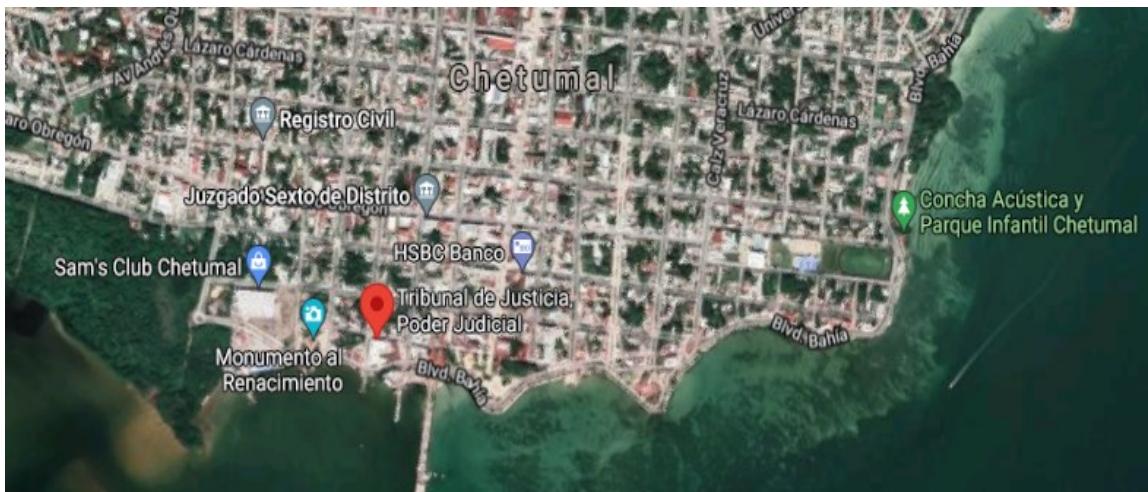
### 2.2.1. Organigrama

Debido a su tamaño, el organigrama puede ser consultado en el apartado de anexos de la memoria de estadía profesional (Anexo 1).

### 2.3. Ubicación

Actualmente, esta institución se encuentra ubicada en Av. Independencia Número 2. Esq. Boulevard Bahía, Col. Centro, C.P. 77000 en Chetumal, Quintana Roo, México. Su horario es de lunes a viernes desde las 08:30 A.M hasta las 15:00 P.M. El teléfono de atención es (983) 832 1000.

**Figura 1.** Ubicación del Tribunal Superior de Justicia de Quintana Roo



Fuente: Google Maps (2021).

## **2.4. Misión**

Resolver de manera justa, imparcial y expedita las controversias jurídicas de los particulares entre sí, de éstos con el Estado y de los órganos del Estado entre sí, mediante la integración de las más nuevas formas de impartir justicia, para garantizar el ejercicio y la tutela de los derechos de las personas conforme a los ordenamientos jurídicos, y la preservación de la convivencia pacífica en el marco del Estado de Derecho (Ávila, 2004, párr. 1).

## **2.5. Visión**

Ser un Poder Judicial con reconocimiento y prestigio público que se distinga por la imparcialidad, honradez, profesionalismo y excelencia en la impartición de justicia, orientado a mejorar la eficacia y calidad de su gestión, con procesos más ágiles y accesibles a toda la población (Ávila, 2004, párr. 1).

## **2.6. Valores**

El Tribunal Superior de Justicia de Quintana Roo no cuenta con valores, puesto que no es una empresa, sino que es una institución de gobierno. Por lo tanto, se proponen los siguientes valores:

- Justicia: Tener la necesidad de mantener equilibrada la armonía entre los integrantes de una sociedad.
- Igualdad: Reconocer a todos por igual y tratar a todos de la misma manera por la ley y seguir las leyes con las que se rige el estado, evitando la discriminación y agresión entre los miembros de la sociedad.
- Certeza: Juzgar por sí mismo mediante evidencias que acredite el éxito de

su libertad, mediante la impartición del poder judicial.

- Legalidad: Garantizar las primicias de la ley con respecto a cualquier clase de normas, aludiendo a los principios jurídicos, sin castigar acciones y/u omisiones si no se previene como delito por las leyes.
- Imparcialidad: Ser capaces de emitir un juicio y tomar una decisión, sin tratar de aspirar a la justicia y ser alguien parcial, dejando de lado los intereses personales al momento de decidir los asuntos judiciales.
- Independencia: Evitar someterse ante una voluntad que no se encuentre dentro de las leyes, manteniendo la imparcialidad, siguiendo las reglas y mantenerse en abstinencia de seguir actitudes que se encuentren fuera de las leyes establecidas.

## **2.7. Objetivo general**

Ejercer justicia judicial a la ciudadanía, mediante el uso de los principios morales por los que se rige el estado libre y soberano para asegurar y mantener la seguridad jurídica del estado al momento de impartir justicia, mediante el seguimiento de los valores que caracterizan al Tribunal Superior de Justicia de Quintana Roo (propuesta).

## **2.8. Objetivos específicos**

- Aplicar justicia de manera transparente, justa y libre, siguiendo el régimen judicial (propuesta).
- Mejorar los procesos jurídicos para ser catalogados como una institución gubernamental confiable y creíble (propuesta).
- Dar una buena imagen sostenida de las decisiones impartidas por el equipo de trabajo profesional de manera justa y seria (propuesta).
- Resolver conflictos de manera imparcial y completa (propuesta).

**CAPÍTULO III**  
**MARCO TEÓRICO**  
**“FUNDAMENTOS PARA EL DESARROLLO MÓVIL”**

**3.1 CONCEPTO DE DESARROLLO DE APLICACIONES MÓVILES**

**3.2 LA EVOLUCIÓN DE LAS APLICACIONES MÓVILES EN LAS EMPRESAS**

**3.3 DIFERENCIA ENTRE APLICACIONES WEB Y MÓVILES**

**3.4 TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES MÓVILES**

**3.4.1 ANDROID STUDIO**

**3.4.2 APACHE**

**3.4.3 API**

**3.4.4 DART**

**3.4.5 FLUTTER**

**3.4.6 JSON**

**3.4.7 MYSQL**

**3.4.8 POSTMAN**

**3.4.9 PROGRAMACIÓN SQL**

**3.4.10 VISUAL STUDIO CODE**

**3.4.11 WEB SERVICES**

**3.4.12 XAMPP**

**3.5 METODOLOGÍAS**

**3.5.1 METODOLOGÍA SCRUM**

**3.5.2 METODOLOGÍAS PARA EL DESARROLLO MÓVIL**

**3.6 CASOS DE ÉXITO**

**3.6.1 TRIBUNAL VIRTUAL MÓVIL**

**3.6.2 TRIBUNAL MÓVIL BCS**

**CAPÍTULO III**  
**MARCO CONTEXTUAL**  
**“FUNDAMENTOS PARA EL DESARROLLO MÓVIL”**

### **3.1. Concepto de desarrollo de aplicaciones móviles**

Como parte importante de las TIC, la programación o desarrollo de aplicaciones móviles es una de las ramas más importantes para una persona especializada en dicha carrera. En el futuro, el desarrollo móvil será un pilar importante en el mundo del desarrollo de *software*. Citando a Walterson (2019):

Las aplicaciones móviles son una manera más de comunicación, que cobra más importancia con el paso del tiempo. Probablemente en no mucho tiempo se convertirán en la única manera utilizada por los usuarios para interactuar con algunas Marcas, sobre todas aquellas que pertenecen al sector retail, es decir, la que su cliente final son particulares (párr. 4).

Esta rama surgió desde hace años, cuando se empezaron a adaptar los programas informáticos y desde la creación de los primeros dispositivos móviles. Muchos autores, e incluso especialistas de esta carrera han definido que el desarrollo de aplicaciones móviles sigue las reglas de la programación de sistemas informáticos.

En el desarrollo y diseño de aplicaciones la meta es que puedan ser ejecutadas en *smartphones* o *tablets*, a través de funciones aisladas y limitadas que son del agrado de los usuarios y desarrolladores, pese a que actualmente los dispositivos móviles ya tienen la potencia para la multifuncionalidad (Rokket Labs, 2020).

En pocas palabras, el desarrollo móvil es la rama de la informática de la cual se usa el diseño y la construcción de aplicaciones que se descargan en los dispositivos móviles, todo esto mediante el uso de los fundamentos de programación orientada al desarrollo de estas aplicaciones.

Para desarrollar una app móvil se deben usar herramientas como un ambiente

de desarrollo que se denomina “Entorno de Desarrollo Integrado” y un *framework* para estructurar las líneas de código de una app.

### **3.2. La evolución de las aplicaciones móviles en las empresas**

Hoy en día, hay que encontrar soluciones para innovar en la vida cotidiana, lo mismo ocurre en el desarrollo de aplicaciones, tanto web como móviles. Todo tiene un origen, así como una historia y un proceso de actualización de sistemas informáticos y “desde que las aplicaciones móviles llegaron al mercado, se hicieron cargo de casi todas las categorías (utilidades, productividad, juegos, entretenimiento, estilo de vida, redes sociales y noticias)” (Veleta, 2020, párr. 1). Los equipos de desarrollo tienen la tarea de diseñar y desarrollar aplicaciones móviles de acuerdo con el contexto en el que se vayan a utilizar, ya sea para empresas, instituciones de gobierno, proyectos escolares, etc.

Por tal motivo, se manejan versiones para identificar aspectos importantes como el diseño, colores, estructura del proyecto, entre otros elementos que conforman una aplicación móvil. Por lo tanto, una aplicación móvil va evolucionando y llega a ser un sistema que se adapte a las necesidades de las personas, actualizándose para llegar a más usuarios. Las aplicaciones móviles han causado un gran impacto en la sociedad conforme evolucionan.

Este fenómeno evolutivo se fue extendiendo poco a poco por las empresas, las cuales ven una excelente oportunidad para atraer a sus clientes para que consuman y utilicen los servicios que ofrecen, haciendo que el uso de aplicaciones móviles sea un arma potencial e innovadora para el mundo empresarial (Solera, 2020).

En definitiva, las empresas y/o instituciones de gobierno necesitan innovar con la implementación de aplicaciones móviles para hacer más ágiles sus procesos y que sean aplicaciones corporativas, haciendo más sencilla la ejecución de procesos en cuanto a su uso en las áreas funcionales de dicha empresa.

### **3.3. Diferencia entre aplicaciones web y móviles**

“El desarrollo de aplicaciones no consiste únicamente en planificar un listado de funcionalidades, también implica ciertas decisiones a nivel tecnológico, como el lenguaje de programación o el tipo de servidor” (Montejo, 2020, párr. 1). Este tema no es nada nuevo en la informática.

El problema surge cuando las personas piensan que la funcionalidad de las aplicaciones web y móviles son las mismas, pero no es así. Por tal motivo, se apreciará una explicación sobre algunas diferencias entre estos dos tipos de sistemas informáticos.

Existen ciertas diferencias entre aplicaciones web y móviles, por ejemplo, una aplicación móvil es un programa que se puede descargar e instalar en dispositivos móviles mediante el uso de tiendas como *Google Play Store* y *App Store*, mientras que una aplicación web son páginas que tienen un tamaño de pantallas para visualizarse en dispositivos móviles, *tablets* y *smartphones* (MurciaEconomía, 2019).

El acceso a estas aplicaciones es diferente, para usar una aplicación móvil se debe realizar la descarga de esta en el dispositivo, una vez terminado se creará un acceso directo para poder usarse, obviamente con la necesidad de usar internet para descargar dicha aplicación.

En cambio, el acceso a una aplicación web siempre será por conexión a internet mediante el uso de un navegador, que permita visualizar todo el contenido de esta, ingresando su dirección o URL en el buscador del navegador.

### **3.4. Tecnologías para el desarrollo móvil**

Antes de iniciar con todo proyecto de TI, se deben de tener presentes las tecnologías que se usarán para llevar a cabo productos de *software*. En este caso, el desarrollo de aplicaciones móviles se basa en el uso de tecnologías como lo son: Entorno de Desarrollo Integrado (IDE), Sistema Gestor de Base de

Datos (SGBD), Interfaz de Programación de Aplicaciones (API), paquetes de desarrollo, lenguajes de programación, entre muchas otras tecnologías para desarrollar aplicaciones móviles.

Por lo tanto, para llevar a cabo el desarrollo del aplicativo móvil que llevará por nombre "Tribunal Familiar Digital", se necesitarán de las tecnologías que se explicarán en este marco teórico.

### **3.4.1. Android Studio**

Cuando un programador inicia con el desarrollo móvil, *Android Studio* será lo primero que venga a la mente, dado que es uno de los ambientes del desarrollo móvil más utilizado en la actualidad y uno de los más populares dentro del desarrollo de aplicaciones móviles para dispositivos con Sistema Operativo *Android*.

Según Jiménez (2013), es una interfaz de desarrollo, es decir, un escritorio de trabajo para un desarrollador de aplicaciones móviles, donde se pueden encontrar todos los elementos y recursos que se vayan a utilizar en el proyecto móvil, siendo un entorno de desarrollo moderno que ofrece herramientas útiles para desarrollar aplicaciones móviles para el Sistema Operativo *Android*.

En definitiva, *Android Studio* es un IDE móvil desarrollado por Google, que ofrece una variedad de herramientas para crear aplicaciones móviles para dispositivos que tengan el Sistema Operativo *Android*, por lo que para este proyecto se usará *Android Studio* como el IDE para desarrollar la aplicación móvil.

### **3.4.2. Apache**

“El servidor Apache nació a mediados de los años noventa y es sus mejores años alcanzó una cuota de mercado del 70% del web, siendo el primer servidor que alojó más de 100 millones de sitios web” (De León, 2019, párr. 1).

Es un servidor que permite servir el contenido de las peticiones realizadas por los clientes, es decir, desde un navegador web. Afortunadamente, Apache es un

servidor gratuito y multiplataforma. Cuando se realiza la búsqueda de una página web en el navegador, Apache realiza su trabajo utilizando el protocolo HTTP para poder visualizar los elementos de dicha página de manera segura, para este proyecto se usará como herramienta donde se usará el protocolo HTTP para la comunicación y el uso de la base de datos como servidor local.

### **3.4.3. API**

Siguiendo a Fernández (2019), *Application Programming Interfaces* (API), o en español interfaz de programación de aplicaciones, es un protocolo para realizar la integración y desarrollo en el *software* de aplicaciones, teniendo como objetivo permitir la comunicación entre dos aplicaciones informáticas mediante un conjunto de reglas.

Por lo tanto, su uso en este proyecto será para consumir los servicios que ofrece el Tribunal Superior de Justicia de Quintana Roo y realizar la comunicación entre aplicaciones.

### **3.4.4. Dart**

El lenguaje de programación Dart se creó en el 2011 por Google que les permite a los desarrolladores usar un lenguaje orientado a objetos, siendo una alternativa de uso del popular lenguaje JavaScript, siendo más fácil y rápido de utilizar para el desarrollo de aplicaciones móviles (Pascual, 2021).

En definitiva, Dart es un lenguaje de programación que trabaja en conjunto con el *framework Flutter* para el desarrollo de aplicaciones para *Android*, *iOS*, web, entre otros, que compila el código realizado de forma nativa de cada plataforma. Por lo tanto, el uso de Dart en este proyecto como lenguaje de programación móvil es de suma importancia ya que se usa en conjunto con *Flutter*.

### 3.4.5. Flutter

“*Flutter* es un SDK de Google creado para desarrollar aplicaciones nativas multiplataformas, que permite a los desarrolladores crear aplicaciones iOS y Android utilizando exactamente el mismo código” (Merayo, 2019, párr. 1).

Ya para terminar, por *framework* o traducido como marco de trabajo, es una herramienta utilizada en la programación para la organización y desarrollo de *software*. Es una herramienta de *software* que ayuda al programador a desarrollar *software* de manera más rápida mediante las funciones que ofrece.

Por lo tanto, *Flutter* es un *framework* para el desarrollo de aplicaciones en diferentes plataformas, utilizado más para el desarrollo de aplicaciones móviles para plataformas como *Android* y *iOS*, por lo que se usará en este proyecto como el *framework* móvil para el desarrollo móvil.

### 3.4.6. JSON

JSON (*JavaScript Object Notation* o traducido como Notación de Objetos JavaScript) es un estándar para el almacenamiento e intercambio de información de texto sencillo para su fácil comprensión entre los desarrolladores de *software* (HostGator México, 2021).

Es un formato de programación sencillo que permite realizar el intercambio de información de datos, que va de la mano con el API REST o API, puesto que estos están escritos en formato JSON para hacer más fácil la comprensión y lectura de los datos a los programadores, siendo clave para el proyecto para que se pueda realizar la comunicación de datos de una API.

### 3.4.7. MySQL

MySQL es uno de los Sistemas Gestores de Base de Datos (SGBD) más usados para la programación, gestión y administración de bases de datos relacionales

mediante la ejecución de sentencias de programación mediante la ocupación del lenguaje SQL. Como señala Robledano (2019):

MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basado en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por *Sun MicroSystems* en 2008 y está a su vez comprada por *Oracle Corporation* en 2010, la cual ya era dueña de un motor propio InnoDB para MySQL (párr. 2).

Lo que puede ser una característica esencial en la administración de bases de datos con MySQL es que su manera de trabajar se basa en la creación de bases de datos relacionales, es decir, se utilizan muchas tablas que se relacionan entre sí para almacenar la información y mantener una organización de los datos de manera íntegra, siendo el SGBD usado para el proyecto móvil.

#### **3.4.8. Postman**

*Postman* es una herramienta utilizada para realizar pruebas, consumo y depuración de una API, con el objetivo de monitorear, escribir pruebas automatizadas para ellas, entre muchas otras operaciones que se puedan realizar. *Postman* se puede usar tanto en navegadores como aplicación de escritorio. Citando a Cuervo (2019) se concluye que:

*Postman* nace siendo una herramienta de *software* para la realización de peticiones entre APIs de manera sencilla, siendo el caso de que un usuario que utiliza *Postman* sea un desarrollador de aplicaciones, pueda operar una tarea mediante el uso de una API para desarrollar u operar sobre ella.

Para terminar, *Postman* es una herramienta de desarrollo web que permite realizar peticiones para el protocolo HTPP a cualquier API, que permite realizar pruebas a la hora de programar, así como permite comprobar la funcionalidad del desarrollo de las aplicaciones web, por lo que se usará para comprobar el contenido de las APIs que se usarán para la comunicación entre aplicaciones.

### 3.4.9. Programación SQL

La programación SQL es aquella que le permite al programador o Administrador de Bases de Datos ( *DataBase Administrator* en inglés) interactuar con una o muchas bases de datos relacionales que se encuentren sujetas a las sentencias de este lenguaje. SQL (de las siglas de Lenguaje de Consulta Estructurado traducido al español) es un lenguaje de base de datos valioso para una persona que se relacione con el campo de la informática para la recopilación y organización de la información.

La programación SQL es utilizada para administrar los datos organizados en tablas que se encuentran en las bases de datos relacionales, siendo así que el uso del lenguaje SQL pueda ser para consultar, actualizar y reorganizar los datos, así como para realizar la creación y actualización de la estructura de una base de datos y llevar un control de los datos (Universidad Internacional de Valencia, 2019).

En conclusión, el lenguaje SQL es utilizado por los Administradores de Bases de Datos para la recopilación de información en bases de datos relacionales, y guardar dicha información en tablas, es decir, toda información que llegue de las aplicaciones tanto web como móvil serán las que alimenten a la base de datos para almacenar la información obtenida en sus tablas relacionales, siendo este el lenguaje y sentencias SQL que se usarán para la base de datos.

### 3.4.10. Visual Studio Code

Se define a *Visual Studio Code* como un editor de código fuente o como un IDE de código abierto que le permite al programador trabajar con diversos lenguajes de programación, así como realizar funciones como descargar y administrar extensiones para personalizar a este editor para trabajar de una manera más agilizada en los proyectos informáticos.

Citando a EcuRed contributors (2020), se concluye que:

*Visual Studio Code* es un editor de código fuente desarrollado por

Microsoft para Windows, Linux y macOS. Incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código. También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto (párr. 1).

En definitiva, *Visual Studio Code* es una herramienta de *software* que se utiliza para trabajar con muchos lenguajes de programación, principalmente utilizado para el desarrollo de aplicaciones web, por lo que se usará en el proyecto como IDE para la creación de las APIs.

### **3.4.11. Web Services**

Los servicios web o *web services* son tecnologías que se implementan en el desarrollo de aplicaciones web, donde se emplean protocolos y estándares para el intercambio de datos entre aplicaciones informáticas.

Un *web service* llega a ser una vía de comunicación y operatividad entre máquinas conectadas a la red, ya sean públicos o privados, ya que esto se basa en el envío de solicitudes y el recibimiento de una respuesta entre un cliente (el que envía la solicitud de información al servidor) y un servidor (aquel que devuelve una respuesta al cliente), siendo así una manera de traficar los mensajes entre dos máquinas (Baquero, 2015).

Para finalizar, el *web service* está diseñado para soportar la interacción máquina-máquina, donde se pueden apreciar interfaces descritas en un formato procesable en un equipo de cómputo, haciendo posible las interacciones mediante el intercambio de mensajes con el uso de estándares informáticos, por lo que tendrán un papel importante en el funcionamiento de la aplicación móvil, ya que se consumirán en forma de API para visualizar los datos y servicios que se ofrecen.

### 3.4.12. XAMPP

XAMPP es un paquete de *software* libre, donde su principal control cae en el SGBD MySQL, servidor Apache y los intérpretes para los lenguajes de programación PHP y Perl. Su nombre proviene del acrónimo X de cualquiera de los Sistemas Operativos, A de Apache, M de MariaDB/MySQL, P de PHP y PP de Perl.

XAMPP es una herramienta basada en PHP utilizada para probar el desarrollo web sin la necesidad de acceder a internet, así como no se tienen que realizar configuraciones entre los servidores que ofrece ya que esta herramienta provee una configuración funcional al momento de ser instalada en una *laptop* o *PC* donde se vaya a desarrollar una aplicación web (García, 2020).

En definitiva, XAMPP es un producto de *software* empleado para el desarrollo web, conteniendo herramientas de *software* como servidor, manejador de base de datos e intérpretes para los códigos de los lenguajes de programación, por lo que el uso de XAMPP será importante ya que activando este paquete de *software* se podrá realizar la base de datos y usar el servidor Apache para visualizar el SGBD MySQL.

## 3.5. Metodologías

En retrospectiva, una metodología hace referencia a los procedimientos basados en principios lógicos, que hacen posible alcanzar ciertos objetivos que rigen en una investigación y son “parte del proceso de investigación o método científico, que sigue a la propedéutica, y permite sistematizar los métodos y las técnicas necesarias para llevarla a cabo” (EcuRed contributors, 2014, párr. 1).

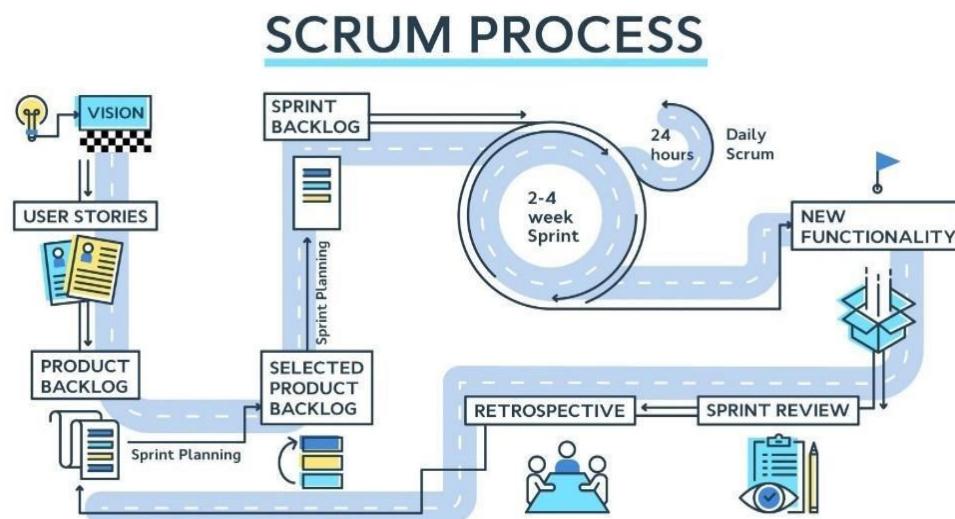
Los métodos son elegidos por el investigador para facilitar el descubrimiento de conocimientos ya conocidos y confiables para solucionar una situación problemática que se había planteado.

### 3.5.1. Metodología Scrum

La metodología Scrum se utiliza para la gestión de procesos y equipos, que es difícil para cualquier empresa, no hablando de la parte de recursos, sino de la parte del tiempo, protocolos y asignación de tareas de mayor peso.

La metodología Scrum es aquella que permite planear proyectos complejos que se estén desarrollando en entornos ya sean cambiantes o dinámicos de manera flexible, esta metodología se basa en la entrega de avances parciales del producto final con base al valor que se le ofrece a un cliente, mejorando el trabajo entre equipos para ayudarlos a aprender y organizar sobre éxitos o fracasos para ganar experiencia mediante el uso de herramientas para llevar una organización más ágil y productiva (Galiana, 2021). La metodología Scrum sirve para abordar tareas complejas de un proyecto, mejorando el trabajo para llevar una buena organización.

**Figura 2.** Procesos de la metodología Scrum



**Fuente:** Fernández (2019).

### 3.5.2. Metodologías para el desarrollo móvil

El desarrollo móvil no se salva de sufrir problemas a la hora de realizar el proyecto de *software*. Todo proyecto informático se lleva a cabo por equipos de desarrollo que requieren de metodologías de desarrollo para organizar las actividades de manera rápida. Teniendo en cuenta a Rodríguez (2011), las metodologías para el desarrollo de aplicaciones móviles son las siguientes:

- Modelo *Waterfall* (en cascada): Clásico. Sólo aplicable cuando están totalmente cerrados los requisitos y no van a cambiar. No hay retroalimentación entre las fases en que se divide el proyecto. Por lo que cada fase se va cerrando de forma secuencial. Todo el proceso está fijado por fechas límites y presupuestos. Este modelo sólo es aconsejable para proyectos móviles muy controlados y previsibles, no existe incertidumbre por lo que se quiere hacer ni influyen los cambios en la industria.
- Desarrollo ágil: Las metodologías ágiles son muy adecuadas para el desarrollo de aplicaciones móviles. En primer lugar, la alta volatilidad del entorno hace que constantemente el equipo de desarrollo se deba adaptar a nuevos terminales, cambios en la plataforma o en el entorno de desarrollo.
- *Mobile-D*: El objetivo de este método es conseguir ciclos de desarrollo muy rápidos en equipos muy pequeños. Fue creado en un proyecto finlandés en 2005, pero sigue estando vigente. Basado en metodologías conocidas pero aplicadas de forma estricta como: *extreme programming*, *Crystal Methodologies* y *Rational Unified Process* (párr. 3).

En conclusión, el desarrollo móvil no es sencillo, se tienen que seguir ciertas reglas de programación, así como métodos, modelos, incluso metodologías para que todo salga bien y se pueda entregar un producto de *software* móvil a tiempo y sin errores.

### **3.6. Casos de éxito**

#### **3.6.1. Tribunal Virtual Móvil**

Tribunal Virtual Móvil es una aplicación móvil creada por el Poder Judicial del Estado de Nuevo León en el 2019, con la cual los usuarios podrán acceder a la aplicación con el objetivo de consultar sus expedientes autorizados de manera rápida y organizada (Poder Judicial del Estado de Nuevo León, 2019).

Esta aplicación funciona con base a los servicios que ofrece la institución gubernamental del Estado de Nuevo León, donde se pueden consultar los expedientes sin ningún problema, solo se debe iniciar sesión con el nombre de usuario y con la contraseña, solo así el usuario podrá acceder a sus expedientes. En dado caso de que no exista una cuenta de usuario, solo se podrá obtener la información de los distritos del Estado de Nuevo León mediante un directorio de distritos en la interfaz de bienvenida, así como también se tendrá información de contactos donde aparecerá información para contactar con la persona que realizó la aplicación móvil dentro de la institución.

Por lo que se ha visto en *Google Play Store* sobre sus pantallas, el área de oportunidad en la que debería mejorar sería en el diseño de las interfaces ya que se ven sencillas y no se ven tan atractivas, ya que solo se encuentra con un diseño básico. Por otro lado, las funcionalidades de la aplicación móvil llamada “Tribunal Virtual Móvil” funcionan de manera correcta, cumpliendo con sus objetivos.

**Figura 3.** Interfaz de la aplicación móvil Tribunal Virtual Móvil.



**Fuente:** Poder Judicial del Estado de Nuevo León (2019).

### 3.6.2. Tribunal Móvil BCS

Tribunal Móvil de BCS es una aplicación móvil creada por el Tribunal Superior de Justicia del Estado de Baja California Sur, esta aplicación tiene como objetivo brindar acceso a los usuarios mediante el ingreso a esta al momento de iniciar sesión, para poder obtener sus expedientes de manera organizada en carpetas, autorización y actualización (Tribunal Superior de Justicia de Baja California Sur, 2015).

Las funcionalidades de la aplicación son las de brindar un mapa con la localización de todos los juzgados o distritos existentes en el Estado de Baja California Super, así como el uso y consulta de los servicios que ofrece el Tribunal Superior de Justicia del Estado de Baja California Sur.

Con esta aplicación, se podrá tener todos los expedientes enlistados por

carpetas, de manera actualizada o por autorización de la institución.

En cuanto a las áreas de oportunidad, se comentará el diseño, puesto que es un diseño sencillo, algo llamativo y teniendo seriedad para una dependencia, pero en la parte de funcionalidad, esta aplicación funciona bien con los objetivos que se plantean.

**Figura 4.** Interfaz de la aplicación móvil Tribunal Móvil de BCS.



**Fuente:** Poder Judicial del Baja California Sur (2015).

**CAPÍTULO IV**  
**DESARROLLO DEL PROYECTO**  
**“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL**  
**SUPERIOR DE JUSTICIA DE QUINTANA ROO”**

- 4.1 ACTIVIDADES REALIZADAS**
- 4.2 CRONOGRAMA DE ACTIVIDADES**
- 4.3 METODOLOGÍA APLICADA**
- 4.4 MATERIAL, EQUIPO Y HERRAMIENTAS UTILIZADAS**
- 4.5 PROPUESTA**
  - 4.5.1 REQUERIMIENTOS**
  - 4.5.2 DISEÑO DEL SISTEMA**
  - 4.5.3 SPRINTS**
  - 4.5.4 DIAGRAMA PAD DE LA APLICACIÓN MÓVIL**
  - 4.5.5 DIAGRAMAS DE CASO DE USO DE LA APLICACIÓN MÓVIL**
  - 4.5.6 DISEÑO DE LA BASE DE DATOS**
  - 4.5.7 MAPA DE SITIO DE LA APLICACIÓN MÓVIL**
  - 4.5.8 BOSQUEJOS DEL SISTEMA**
  - 4.5.9 DIAGRAMA TECNOLÓGICO DE LA APLICACIÓN MÓVIL**
  - 4.5.10 CAPTURAS DE PANTALLA DE LA APLICACIÓN MÓVIL**

**CAPÍTULO IV**  
**DESARROLLO DEL PROYECTO**  
**“APP MÓVIL TRIBUNAL FAMILIAR DIGITAL DEL TRIBUNAL**  
**SUPERIOR DE JUSTICIA DE QUINTANA ROO”**

#### **4.1. Actividades realizadas**

Para la creación de la aplicación móvil denominada “Tribunal Familiar Digital”, se realizaron las siguientes actividades:

1. Se recolectó la información necesaria mediante una pequeña entrevista al asesor empresarial para saber sobre los requerimientos necesarios para que la aplicación móvil funcione de manera correcta.
2. Se empezó con la planificación del proyecto móvil y se seleccionaron todas las herramientas de *software* y equipo informático que se utilizó para el desarrollo del aplicativo móvil.
3. Se realizó un prototipo de lo que sería la aplicación móvil, para luego sufrir un rediseño totalmente nuevo.
4. Se diseñaron nuevas pantallas las cuales fueron definitivas para el contenido de la aplicación móvil.
5. Se construyeron y estructuraron las interfaces de la aplicación móvil para tener una buena vista para los usuarios.
6. Se realizó el consumo de las APIs (Interfaces de Programación de Aplicaciones) en la aplicación móvil para que funcionen de manera correcta y se puedan visualizar en esta.
7. Se realizó la ejecución de las pruebas de manera local mediante la compilación del proyecto para que se visualice en un dispositivo móvil.
8. Se corrigieron los errores que se encontraron al momento de compilar el proyecto móvil.
9. Se verificó el funcionamiento de la aplicación móvil correctamente y se entregó el código del proyecto.

10. Se subió la aplicación móvil a la tienda móvil *Google Play Store*.

#### **4.2. Cronograma de actividades**

En la tabla 1 se presenta de manera cronológica la forma en la que se desarrollaron las actividades para el desarrollo del proyecto en el periodo de estadía profesional mayo-agosto 2021.

**Tabla 1.** Cronograma de actividades para el desarrollo del proyecto

**Nombre del proyecto:** "App móvil Tribunal Familiar Digital del Tribunal Superior de Justicia de Quintana Roo"

**Período de realización:** 06 de mayo al 08 de agosto de 2021

	<b>ACTIVIDAD/PRODUCTO</b>	<b>SEMANAS</b>				<b>OBSER-VACIO-NES</b>
		<b>MAYO</b>	<b>JUNIO</b>	<b>JULIO</b>	<b>AGOSTO</b>	
1	Recolectar información sobre los requerimientos necesarios para el funcionamiento de la aplicación móvil.					
2	Empezar con la planificación del proyecto y seleccionar las herramientas y equipo que se utilizarán para el desarrollo del aplicativo móvil.					
3	Realizar el prototipo de lo que será la aplicación móvil.					
4	Diseñar las pantallas definitivas que contendrá la aplicación móvil.					
5	Construir y estructurar las interfaces de la aplicación móvil.					
6	Realizar el consumo de las APIs en la aplicación móvil para que funcionen de manera correcta.					
7	Realizar la ejecución de pruebas de manera local, compilando el proyecto para que se visualice en un dispositivo móvil.					
8	Corregir los errores que se encuentren					

	al momento de compilar el proyecto.													
9	Verificar que la aplicación móvil funcione correctamente y entregar el código del proyecto.													
1 0	Subir la aplicación móvil a <i>Google Play Store</i> .													

Fuente: Elaboración propia (2021).

#### 4.3. Metodología aplicada

De acuerdo con la manera de trabajar en el Tribunal Superior de Justicia de Quintana Roo para el desarrollo del proyecto móvil, la metodología aplicada fue la Scrum, esto porque es una metodología ágil el momento de realizar modificaciones del funcionamiento lógico y se pueden identificar más a fondo las soluciones para el desarrollo del aplicativo móvil, por lo que permitió llevar un buen análisis y se pudieron asignar las tareas mediante la entrega de la distribución de carga, agilizando así la realización del proyecto móvil.

No obstante, se tomaron conceptos de la metodología Scrum para hacer nuevas definiciones de procesos para ayudar en el desarrollo de la aplicación móvil, organizando todo, estructurando y rediseñando las pantallas de la aplicación móvil para que se vea más sencilla, atractiva y agradable para los usuarios.

El uso de la metodología Scrum ofreció ciertas ventajas, entre ellas se destacan:

- Intervención del cliente en el proceso de desarrollo de la aplicación móvil.
- Entrega del producto de manera parcial.
- Eliminación de tareas innecesarias.
- Simplicidad al momento de desarrollar la aplicación móvil.

Como opinión, se puede decir que con el uso de la metodología Scrum fue satisfactoria, ya que cumplió con su propósito el cual era agilizar los procesos de elaboración del proyecto y se logró concluir en tiempo y forma, a pesar de estar un poco ajustado de tiempo el periodo de estadías profesionales, pero gracias al uso de la metodología Scrum, se pudieron obtener buenos resultados.

En la tabla 2 se presenta la manera en que se desarrolla un *Sprint* de la

metodología Scrum el cual se dividió en las 5 etapas de dicha metodología para abordarse de manera más rápida y eficiente en el proyecto móvil.

**Tabla 2.** Etapas de la metodología Scrum en el proyecto

Metodología SCRUM		
Nombre	Descripción	¿Cómo se utilizó?
Reunión de planificación de <i>Sprint</i>	En esta reunión se define el incremento de producto "entregable" que se llevará a cabo, la forma en que el equipo desarrollará el incremento y el trabajo asignado a cada uno.	Se realizaron pláticas semanales con los asesores de estadía como lo son el empresarial y el académico. Con el asesor empresarial se realizaban reuniones para definir la entrega del proyecto y de la manera en la que se desarrollaría el aplicativo móvil. Mientras que con el asesor académico se realizaron pláticas para la entrega parcial de las pantallas del aplicativo móvil y definir la manera en la que se avanzará con el proyecto.
El Scrum Diario	Su objetivo es que el equipo de desarrollo se sincronice. Primero, cada miembro expone el trabajo realizado en las 24 horas anteriores, si ha habido algún problema o imprevisto que se haya interpuesto en la consecución del objetivo diario y el trabajo que se prevé desarrollar en el día.	Durante el proceso de desarrollo de la aplicación móvil, al momento de llegar al tribunal, exponía el trabajo que estaba realizando con la aplicación. Se llevaba haciendo el Scrum todos los días para que se vaya inspeccionando el desarrollo del proyecto y se dé una respuesta al trabajo realizado.
Trabajo de desarrollo durante el <i>Sprint</i>	Cuando un <i>Sprint</i> es demasiado largo, la definición de lo que se está construyendo puede	Cada vez que había una revisión del proyecto, se llegaba a cambiar la forma de trabajar con el aplicativo

	cambiar, puede aumentar la complejidad y puede aumentar el riesgo. Los <i>Sprints</i> permiten previsibilidad al garantizar la inspección y la adaptación de los avances hacia una meta de por lo menos cada mes de calendario.	móvil, puesto que se encontraban ciertos puntos que no coincidían con el desarrollo del proyecto, haciendo más ágil el proceso de la programación del aplicativo móvil, sin hacer más complejo el trabajo de desarrollo.
Revisión del <i>Sprint</i>	Se lleva a cabo al final del <i>Sprint</i> , para inspeccionar el incremento y adaptar, si es necesario, el Product Backlog.	Durante el proceso de desarrollo, se estuvo inspeccionando el incremento de producción de la aplicación móvil con el asesor académico, dando una retroalimentación sobre los avances vistos a lo largo del periodo de estadía profesional.
Retrospectiva del <i>Sprint</i>	Es una oportunidad para inspeccionarse a sí mismo y crear planes de mejora para ser ejecutados durante el siguiente <i>Sprint</i> , teniendo el propósito de revisar el último <i>Sprint</i> e identificar los temas que salieron bien y las áreas de mejora.	Se hicieron mejoras con respecto a los procesos y herramientas utilizadas para el desarrollo del aplicativo móvil, encontrando áreas de mejora como lo es el diseño de las pantallas del aplicativo móvil.

Fuente: Elaboración propia (2021).

#### 4.4. Material, equipo y herramientas utilizadas

Durante el desarrollo de la aplicación móvil, se han utilizado herramientas tanto de *hardware* como de *software*, así como el uso de equipo para el desarrollo y gestión de este como lo es la computadora de trabajo en donde se desarrolló la

aplicación móvil y un teléfono celular donde se ejecutaron las pruebas de dicha aplicación.

A continuación, se visualizarán las características del equipo de *hardware* empleado en el desarrollo de la aplicación móvil:

**Tabla 3.** Características de la *laptop HP Notebook*

<i>Laptop HP Notebook</i>	
Características	Descripción
Sistema Operativo	<i>Microsoft Windows 10 Home Single Language</i>
Versión del sistema	10.0.1.19043 Compilación 19042
Modelo del sistema	<i>HP Laptop 15-db0xxx</i>
Procesador	AMD a4-9125 RADEON R3, 4 COMPUTE CORES 2C+2G, 2300 Mhz, 2 procesadores principales, 2 procesadores lógicos
Fabricante del Sistema Operativo	<i>Microsoft Corporation</i>
Memoria RAM	4.00 GB
Tipo de sistema	Sistema operativo de 64 bits, procesador x64

Fuente: Elaboración propia (2021).

**Tabla 4.** Características del dispositivo móvil Oppo A12

<i>Teléfono celular Oppo A12</i>	
Características	Descripción
Sistema Operativo	<i>Android versión 9</i>
Modelo	CPH2083

Procesador	Octanúcleo
Memoria RAM	3.00 GB
Almacenamiento en el dispositivo	32.0 GB
Versión de <i>hardware</i>	CPH2083EX_11
Versión de <i>software</i>	CPH2083EX_11_A.57
Versión de <i>Kernel</i>	4.9.117

Fuente: Elaboración propia (2021).

Así mismo, en el diseño y desarrollo del aplicativo móvil se utilizaron ciertas herramientas para que esta pueda funcionar de la manera correcta. A continuación, se presentarán todas las herramientas de *software* empleadas en el desarrollo del proyecto:

**Tabla 5.** Características de las herramientas de *software* utilizadas

<b>Herramientas de <i>software</i> empleadas para el desarrollo del proyecto</b>	
<b>Características</b>	<b>Descripción</b>
Entornos de Desarrollo Integrados (IDEs)	<i>Android Studio</i> para el desarrollo móvil y <i>Visual Studio Code</i> como alternativa de este y en donde se realizaron las APIs.
Framework de desarrollo móvil	<i>Flutter</i> fue el framework informático utilizado para el desarrollo del aplicativo móvil.
Lenguaje(s) de programación	Dart para la programación móvil, SQL para la base de datos y JSON para la creación de las APIs.
Servidores	Apache para usar como servidor en la visualización del gestor de base de datos en el navegador.
Protocolos	APIs (Interfaces de Programación de Aplicaciones) para la comunicación de los servicios a la aplicación móvil.
Sistemas Gestores de Bases de Datos	MySQL para la base de datos.

Ejecutor de pruebas	<i>Postman</i> para la ejecución de pruebas de las APIs.
Paquetes de desarrollo	XAMPP para activar los servicios de Apache y MySQL.

Fuente: Elaboración propia (2021).

## 4.5. Propuesta

La propuesta para esta estadía profesional tiene por nombre “App móvil Tribunal Familiar Digital del Tribunal Superior de Justicia de Quintana Roo”.

### 4.5.1. Requerimientos

Al momento de iniciar un proyecto, en especial un proyecto de *software*, se deben de tomar ciertas acciones como lo son identificar y explicar cuáles serán los requerimientos que tendrá el sistema a desarrollar. En el desarrollo de *software*, los requerimientos principales son los funcionales y no funcionales. Por lo tanto, en la tabla 6 se podrá observar más sobre aquellos requerimientos para el desarrollo de la aplicación móvil Tribunal Familiar Digital.

**Tabla 6.** Requerimientos de la app móvil Tribunal Familiar Digital

<b>Requerimientos de la aplicación móvil Tribunal Familiar Digital</b>	
<b>Requerimientos funcionales</b>	<b>Requerimientos no funcionales</b>
Módulo de inicio de sesión.	El diseño de las pantallas debe ser sencillo y atractivo.
Módulo de registro de usuario.	La aplicación móvil debe de funcionar correctamente.
Módulo de identificación para tomar foto.	El uso de la aplicación móvil debe ser fácil de manejar.
Módulo de sentencias públicas.	La aplicación móvil debe de ser

	adaptable en ambientes web y móvil.
Módulo de portal de citas.	El tiempo de respuesta de la aplicación móvil no debe ser tardado.
Módulo de expedientes electrónicos.	La estructura de las pantallas debe ser cómodas y sencillas para los usuarios.

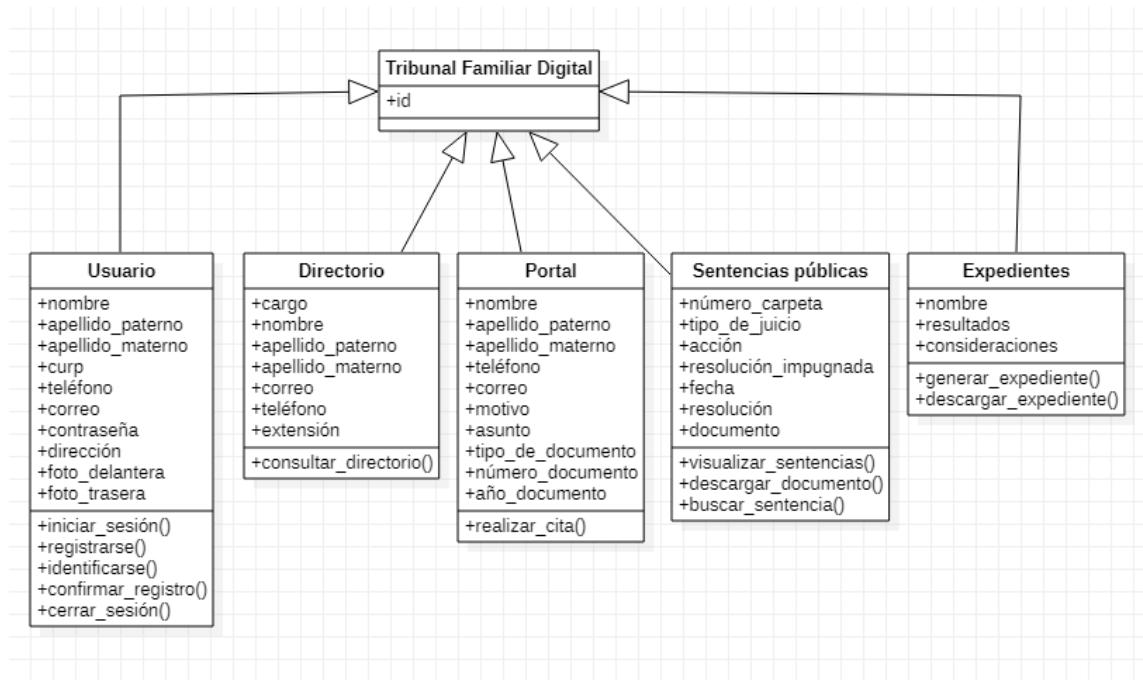
Fuente: Elaboración propia (2021).

#### 4.5.2. Diseño del sistema

El diseño del sistema es un proceso que se detalla paso a paso para concluir con el producto de *software* planteado para que funcione correctamente, definiendo la arquitectura, los módulos funcionales, las interfaces del sistema y los datos que se visualizarán dentro de la esta. Una manera de saber sobre el diseño del sistema es con un diagrama de clases, el cual es un diagrama usado en la ingeniería de *software* para describir la estructura de un sistema donde pueden visualizar las clases del sistema, atributos, operaciones y la relación entre los objetos que interactúan en esta.

A continuación, en la figura 5 se puede apreciar el diagrama de clases de la aplicación móvil Tribunal Familiar digital, donde se puede ver el diseño del sistema de una manera lógica previa a la programación de la aplicación móvil.

**Figura 5.** Diagrama de clases de la app móvil Tribunal Familiar Digital



**Fuente:** Elaboración propia (2021).

#### 4.5.3. Sprints

Como se mencionó con anterioridad en las etapas de la metodología Scrum (tabla 2), para aplicar dicha metodología se llevaron 5 etapas en la elaboración del proyecto. Por lo tanto, en estos *Sprints* del proyecto se dividirán en 3 partes: inicio, diseño y desarrollo y programación.

En la tabla 7, se mostrará el primer *Sprint*, el en cual se podrán visualizar las tareas esenciales las cuales dan inicio al proyecto bajo el uso de la metodología Scrum, partiendo desde la recolección de datos hasta la configuración de todo el proyecto mediante el uso del *framework Flutter*.

**Tabla 7.** Sprint 1 del proyecto

ID Backlog	Tarea	Tipo	Fecha Inicio	Fecha Conclusión	Estado
01	Recolectar información importante sobre el proyecto a realizar.	Investigación	04/05/2021	06/06/2021	Concluido
02	Crear el proyecto en <i>Android Studio</i> utilizando el <i>framework Flutter</i> .	Creación y configuración	05/05/2021	06/06/2021	Concluido
03	Realizar la configuración del proyecto mediante el uso de las herramientas necesarias para la creación de este.	Configuración	05/05/2021	07/06/2021	Concluido

Fuente: Elaboración propia (2021).

En la tabla 8, se mostrará el segundo *Sprint*, el cual fue uno de los más interesantes y extensos de realizar, puesto que se realizaron todos los diseños de la aplicación móvil, así como se llevó la estructuración de las pantallas con base a las necesidades del Tribunal Superior de Justicia de Quintana Roo.

**Tabla 8.** *Sprint* 2 del proyecto

ID Backlog	Tarea	Tipo	Fecha Inicio	Fecha Fin	Estado
04	Creación de la pantalla de bienvenida de la aplicación móvil.	Diseño	04/05/2021	05/05/2021	Concluido
05	Creación de la pantalla de acciones de ingreso del usuario.	Diseño	04/05/2021	05/05/2021	Concluido
06	Creación de la pantalla de inicio de sesión.	Diseño	06/05/2021	07/05/2021	Concluido
07	Creación de la pantalla de registro de usuarios.	Diseño	07/05/2021	08/05/2021	Concluido
08	Creación de la pantalla de identificación de credencial de usuario.	Diseño	08/05/2021	09/05/2021	Concluido
09	Creación de la pantalla de confirmación de registro de usuarios.	Diseño	09/05/2021	10/05/2021	Concluido

10	Creación de la pantalla del tablero de la aplicación móvil.	Diseño	11/05/2021	14/05/2021	Concluido
11	Creación del menú desplegable de la aplicación móvil.	Diseño	15/05/2021	16/05/2021	Concluido
12	Creación del módulo de directorio.	Diseño	17/05/2021	19/05/2021	Concluido
13	Creación del módulo de sentencias públicas.	Diseño	20/05/2021	22/05/2021	Concluido
14	Creación del módulo del portal de citas.	Diseño	23/05/2021	25/05/2021	Concluido
15	Creación del módulo de expedientes electrónicos.	Diseño	26/05/2021	28/05/2021	Concluido
16	Creación de las alertas de inicio y cierre de sesión.	Diseño	29/05/2021	01/06/2021	Concluido
17	Modificar las pantallas de la aplicación móvil.	Diseño	05/06/2021	29/06/2021	Concluido

Fuente: Elaboración propia (2021).

Para finalizar con los *Sprint* del proyecto, en la tabla 9 se mostrará el tercer *Sprint* del proyecto el cual ya consta de cuestiones de desarrollo y programación de los módulos para que funcionen correctamente en la aplicación móvil, con la creación de las rutas, gestión y programación de la base de datos en MySQL y la creación

de las APIs para que sean consumidas en la aplicación móvil mediante el uso del *framework Flutter* que emplea el lenguaje de programación Dart en el IDE *Android Studio*.

**Tabla 9.** Sprint 3 del proyecto

ID Backlog	Tarea	Tipo	Fecha Inicio	Fecha Conclusión	Estado
18	Creación de la base de datos en MySQL.	Desarrollo y programación	30/06/2021	02/07/2021	Concluido
19	Realizar la API para el inicio de sesión y registro de usuarios.	Programación	03/07/2021	05/07/2021	Concluido
20	Realizar la API para el directorio.	Programación	06/07/2021	08/07/2021	Concluido
21	Realizar la API para el portal de citas.	Programación	09/07/2021	11/07/2021	Concluido
22	Realizar la API para el expediente electrónico.	Programación	12/07/2021	14/07/2021	Concluido
23	Consumir la API de login para realizar la validación	Desarrollo y programación	15/07/2021	17/07/2021	Concluido

	del inicio de sesión.				
24	Consumir la API de registro para los nuevos usuarios.	Desarrollo y programación	18/07/2021	20/07/2021	Concluido
25	Consumir la API de directorio para mostrar los datos de los miembros de la segunda sala familiar.	Desarrollo y programación	22/07/2021	24/07/2021	Concluido
26	Consumir la API de portal de citas para registrar una cita.	Desarrollo y programación	26/07/2021	28/07/2021	Concluido
27	Consumir la API de expediente electrónico para los expedientes de los usuarios.	Desarrollo y programación	29/07/2021	01/08/2021	Concluido

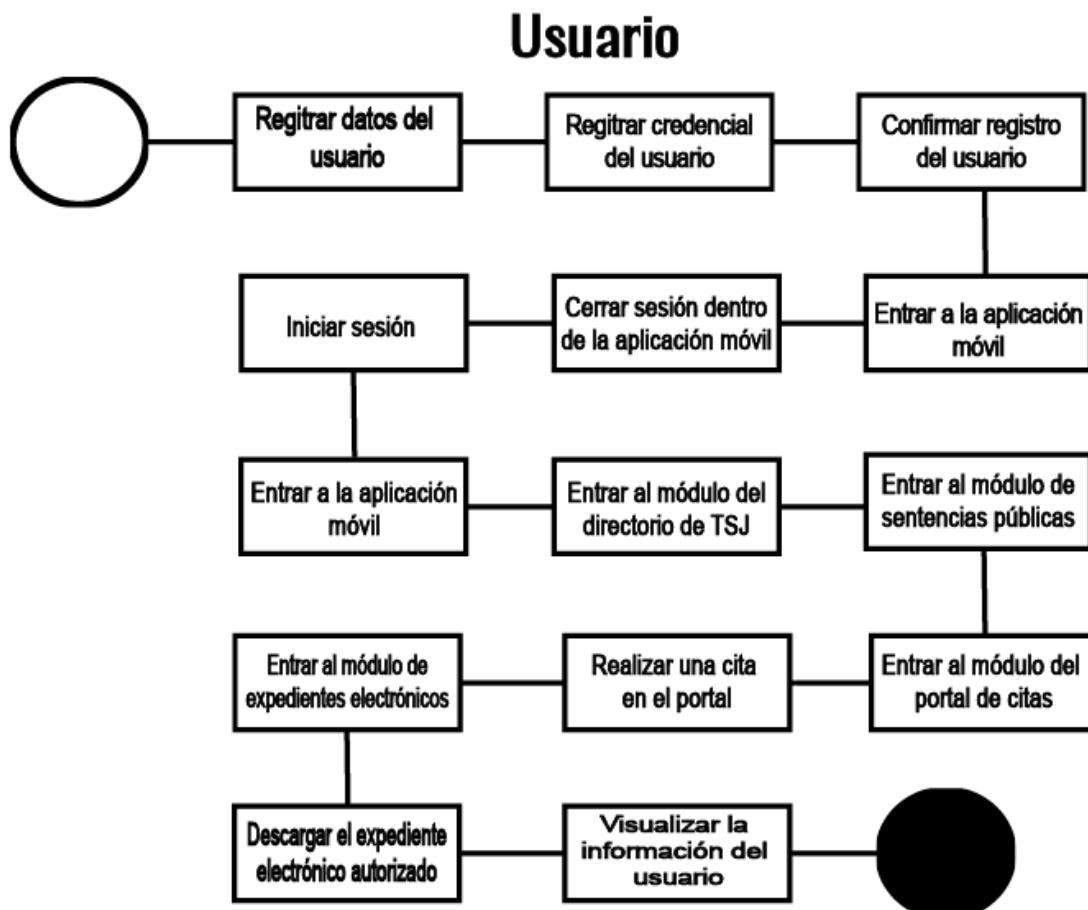
Fuente: Elaboración propia (2021).

#### 4.5.4. Diagrama PAD de la aplicación móvil

El diagrama PAD es un diagrama de *software* utilizado en empresas de manera estructural, mostrando procesos que definen una estructura funcional y que se anticipa al proceso de este, representando un conjunto de procesos esenciales de la empresa. En este caso, el diagrama PAD es el que se utiliza para definir una estructura funcional que presenta los procesos principales de una aplicación informática.

En la figura 6, se puede apreciar de manera simple un diagrama PAD del funcionamiento de la aplicación móvil de una manera estructurada.

**Figura 6.** Diagrama PAD de la app móvil Tribunal Familiar Digital



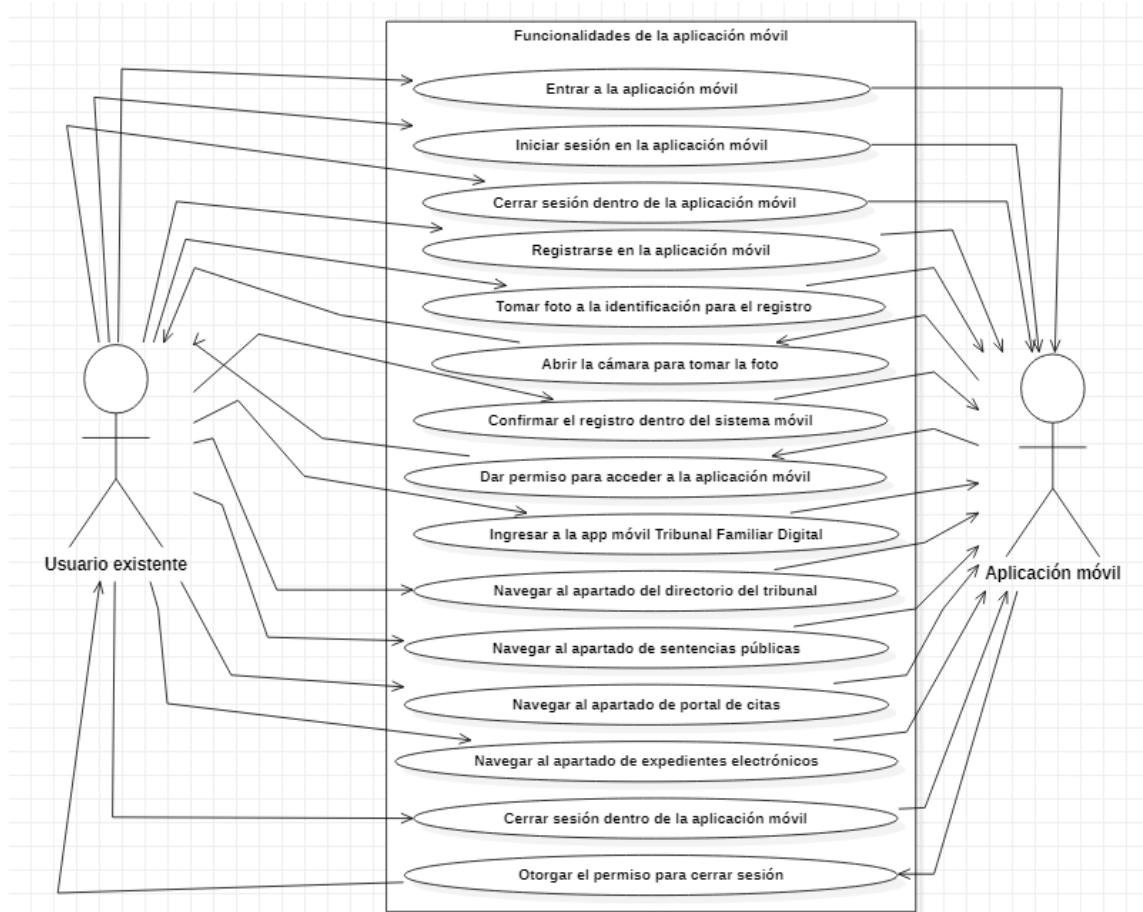
Fuente: Elaboración propia (2021).

#### 4.5.5. Diagramas de caso de uso de la aplicación móvil

El diagrama de caso de uso es otro de los diagramas de *software* empleados para la descripción de los sistemas informáticos, utilizado para describir las actividades que se realizan entre los actores del sistema.

En la figura 7 se explica más sobre el diagrama de caso de uso sobre el funcionamiento del sistema, en donde habrá dos actores, quienes serán el usuario y la aplicación móvil, los cuales interactuarán mediante las actividades que serán los casos de uso para verificar los procesos que realizan en la utilización del sistema móvil.

**Figura 7.** Diagrama de caso de uso de la app móvil Tribunal Familiar Digital



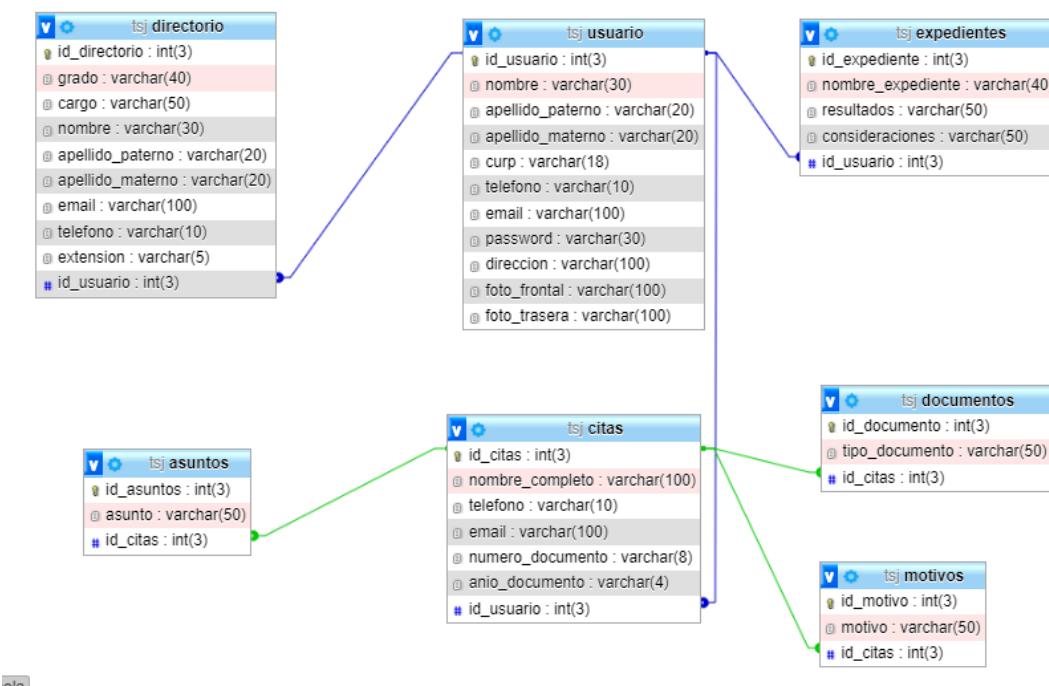
**Fuente:** Elaboración propia (2021).

#### 4.5.6. Diseño de la base de datos

Como se mencionó en el capítulo 3, el diseño de la base de datos sería mediante el manejo del Sistema Gestor de Base de Datos MySQL mediante programación SQL.

Para saber sobre el diseño de la base de datos de manera lógica, en la figura 8 se puede observar un modelado de base de datos que se hace para representar de manera lógica la base de datos, a este modelo se le denomina modelo relacional, el cual representa los datos que se almacenan en las tablas de la base de datos relacional en MySQL.

**Figura 8.** Modelo relacional de la base de datos en MySQL



Fuente: Elaboración propia (2021).

#### 4.5.7. Mapa de sitio de la aplicación móvil

En términos de aplicaciones web, un mapa de sitio es un diagrama o lista de todas las páginas de un sitio web que son accesibles por los usuarios que

acceder a este sitio. Ahora bien, el mapa de sitio en una aplicación móvil es una estructura donde se puede visualizar de manera ordenada y estructurada los módulos o partes de las pantallas de una aplicación móvil.

En la figura 9, se puede observar este mapa del sitio realizado para describir las pantallas y módulos con los que cuenta la aplicación móvil Tribunal Familiar Digital.

**Figura 9.** Mapa del sitio de la app móvil Tribunal Familiar Digital



**Fuente:** Elaboración propia (2021).

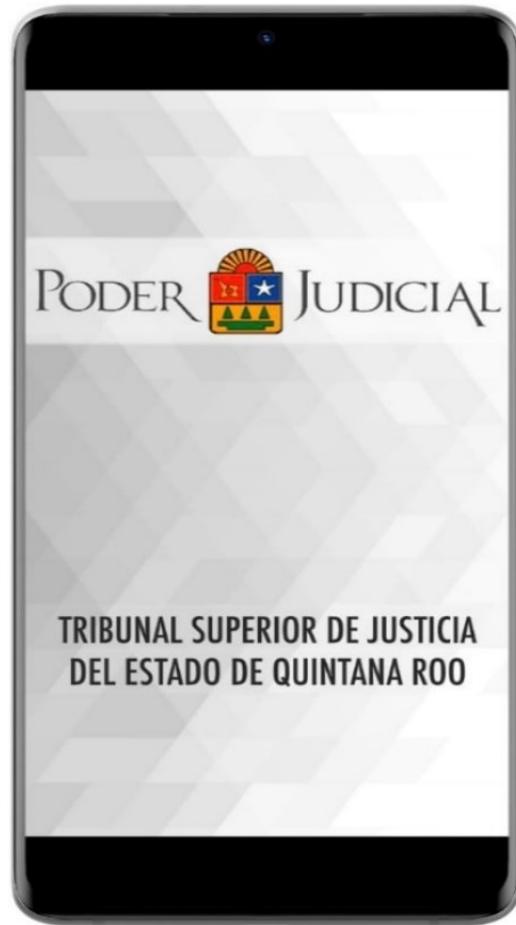
#### 4.5.8. Bosquejos del sistema

Los *mockups* o bosquejos son los diseños digitales de una aplicación ya sea web o móvil. Son los diseños que se utilizan al inicio del proyecto para plasmar las ideas y conceptos que se tienen de lo que será la aplicación. En este caso, como es una aplicación móvil, se incluyó la estructura de las interfaces o pantallas para que se puedan apreciar de manera atractiva y sencilla.

A continuación, se mostrarán algunos bosquejos de lo que se tenía pensado para el desarrollo de la aplicación móvil Tribunal Familiar Digital.

Como primer bosquejo, en la figura 10 se puede apreciar el *mockup* de la pantalla animada, la cual cargará presentando el logo y el nombre de la institución, junto a una barra circular de progreso que indicará que la aplicación está cargando.

**Figura 10.** *Mockup* de la pantalla animada



**Fuente:** Elaboración propia (2021).

En la figura 11, se puede observar el *mockup* de la pantalla de las acciones del usuario, la cual su función será la de indicar la acción que hará el usuario, si se encuentra registrado en la aplicación, iniciará sesión. De lo contrario, si el usuario no se encuentra registrado, tendrá que realizar el formulario de registro.

**Figura 11.** Mockup de la pantalla de acciones de ingreso



**Fuente:** Elaboración propia (2021).

En la figura 12, se presenta el *mockup* de la pantalla del formulario de registro, en el cual el usuario deberá ingresar sus datos personales para poder ingresar a la aplicación móvil y ser un usuario existente en la aplicación móvil Tribunal Familiar Digital.

**Figura 12.** Mockup de la pantalla de registro de usuarios



**Fuente:** Elaboración propia (2021).

En la figura 13, se plasma el *mockup* de la pantalla de identificación de usuario, en donde una vez registrados los datos en la pantalla anterior, el usuario tendrá que tomar foto a su credencial de elector de lado frontal y trasero.

**Figura 13.** Mockup de la pantalla de identificación del usuario



**Fuente:** Elaboración propia (2021).

En la figura 14, se presenta el *mockup* de la pantalla de confirmación de registro, en donde será la última parte para el registro de usuarios, donde el usuario escribirá su correo y contraseña para que sean sus credenciales para el acceso a la aplicación móvil, siendo ahora usados para el inicio de sesión.

**Figura 14.** Mockup de la pantalla de confirmación de registro de usuarios



**Fuente:** Elaboración propia (2021).

En la figura 15, se puede visualizar el *mockup* de la pantalla de inicio de sesión o *login* de usuario, en donde el usuario ingresará a la aplicación móvil escribiendo sus credenciales, las cuales son el correo electrónico y la contraseña.

**Figura 15.** Mockup de la pantalla de inicio de sesión



**Fuente:** Elaboración propia (2021).

En la figura 16, se puede apreciar el *mockup* de una alerta del inicio de sesión, la cual abrirá un diálogo de alerta que le indicará al usuario que sus credenciales no son válidas. Por lo tanto, significa que la validación de usuarios funcionará correctamente, ya que, si no sale una alerta al iniciar sesión, el usuario existe en el sistema. De lo contrario, si por alguna razón el usuario no existe o ingresa sus credenciales mal, saldrá esta alerta.

**Figura 16.** Mockup de la alerta de inicio de sesión



**Fuente:** Elaboración propia (2021).

En la figura 17, se puede apreciar el *mockup* de la pantalla del tablero o *dashboard* de la aplicación móvil, el cual tendrá los módulos principales de la app móvil Tribunal Familiar Digital para que el usuario pueda usar de manera correcta la aplicación móvil. Los módulos son: directorio, sentencias públicas, portal y expediente electrónico.

**Figura 17.** Mockup de la pantalla del dashboard de la aplicación móvil



**Fuente:** Elaboración propia (2021).

Ahora bien, en la figura 18, se puede observar una vista previa de lo que sería el *mockup* de la pantalla del módulo directorio, el cual tendrá un diseño de cartas presentando al equipo de trabajo de la segunda sala familiar del Tribunal Superior de Justicia de Quintana Roo.

**Figura 18.** Mockup de la pantalla del módulo directorio



**Fuente:** Elaboración propia (2021).

En la figura 19, se puede ver el *mockup* de la pantalla del módulo sentencias públicas, donde se indicó que será una vista web o *webview* de una página del micrositio de la segunda sala del Tribunal Superior de Justicia de Quintana Roo.

**Figura 19.** Mockup de la pantalla del módulo sentencias públicas



**Fuente:** Elaboración propia (2021).

En la figura 20, se puede visualizar lo que sería el *mockup* de pantalla del módulo portal, el cual es básicamente un portal de citas, donde se agenda una cita para solicitar los servicios del Tribunal Superior de Justicia de Quintana Roo.

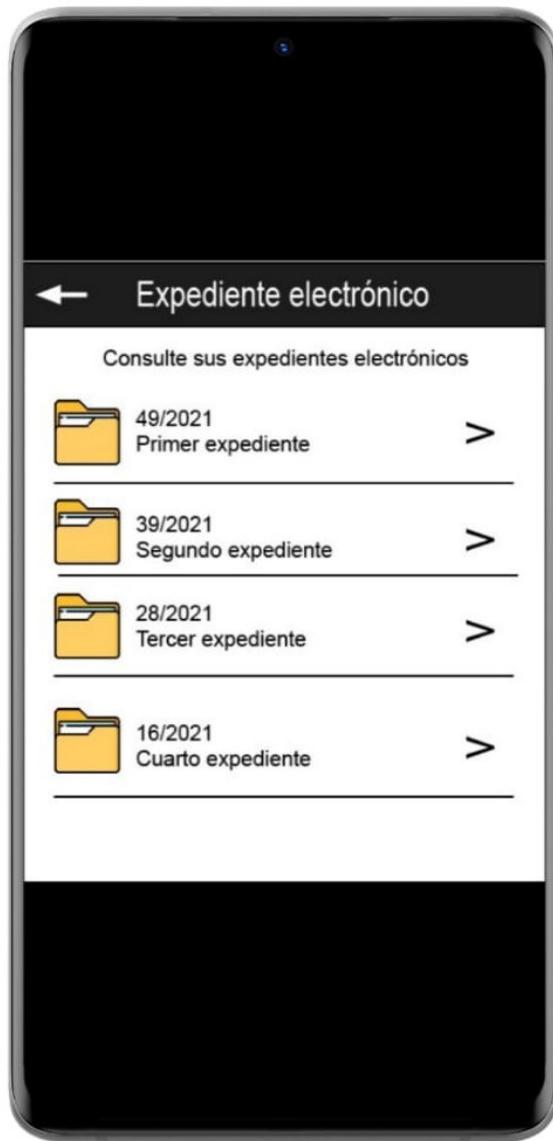
**Figura 20.** Mockup de la pantalla del módulo portal



**Fuente:** Elaboración propia (2021).

En la figura 21, se puede apreciar lo que sería el *mockup* de la pantalla del módulo expedientes electrónicos, en el cual, los usuarios pueden consultar todos los expedientes con los que cuenta dentro de la aplicación móvil.

**Figura 21.** Mockup de la pantalla del módulo expedientes electrónicos



**Fuente:** Elaboración propia (2021).

En la figura 22, se puede visualizar lo que sería el *mockup* del menú desplegable, el cual se encontrará en la parte superior izquierda de la pantalla de la aplicación móvil. En este menú, se pueden encontrar otras pantallas como la página principal, el apartado de vinculación de expedientes, la información personal del usuario y la parte de cerrar sesión dentro de la aplicación móvil.

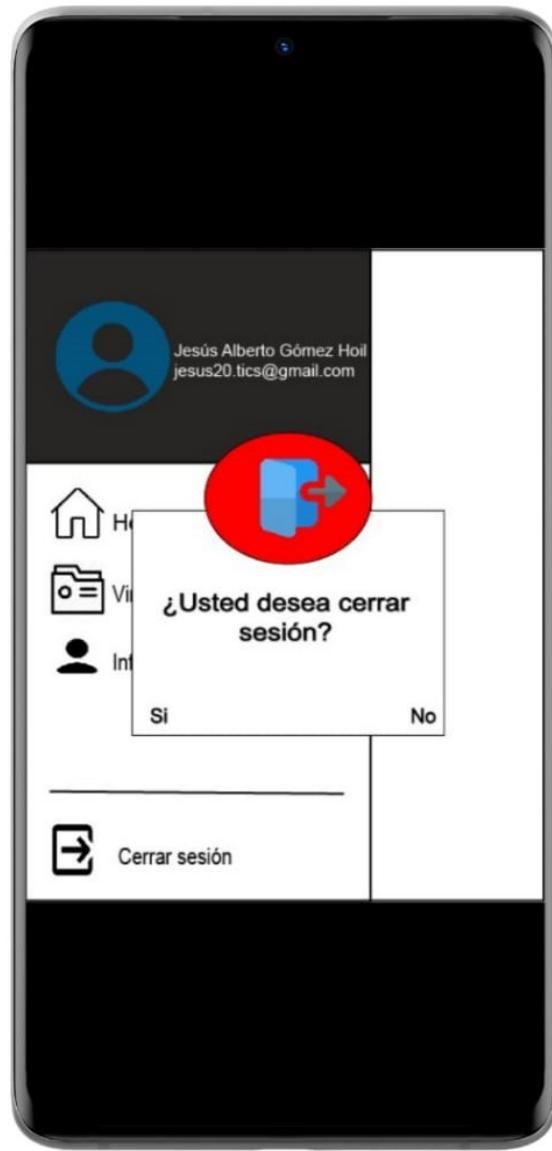
**Figura 22.** Mockup del menú desplegable



**Fuente:** Elaboración propia (2021).

En la figura 23, se puede apreciar otro diálogo de alerta para la parte del cierre de sesión dentro de la aplicación móvil. Esta alerta le indicará al usuario una vez apretado el botón “Cerrar sesión” si desea salir de la aplicación móvil, dándole 2 opciones de sí y no. En caso de aceptar el cierre de sesión, se cierra todo el progreso que tenía el usuario y lo mandará a la pantalla de acciones de ingreso. Ahora bien, si el usuario no acepta el cierre de sesión, se mantendrá dentro de su sesión en la aplicación móvil y podrá seguir navegando dentro de esta.

**Figura 23.** Mockup de la alerta de cierre de sesión



**Fuente:** Elaboración propia (2021).

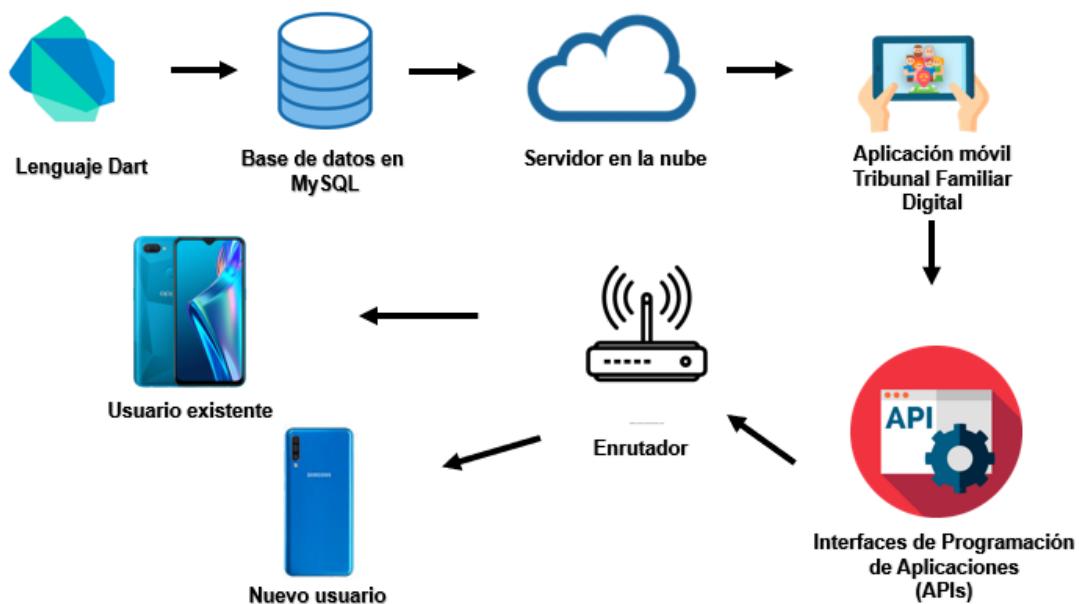
Estos serían todos los bosquejos o *mockups* para darse una idea de lo que será la aplicación móvil Tribunal Familiar Digital. Estos *mockups* fueron de gran ayuda para tener una idea de la estructura de la aplicación móvil.

#### 4.5.9. Diagrama tecnológico de la aplicación móvil

Un diagrama tecnológico o de infraestructura es aquel que presenta un conjunto de elementos tecnológicos por lo que se integran y sustentan ciertos elementos y/o recursos necesarios para la funcionalidad de una aplicación ya sea web o móvil.

A continuación, se mostrará en la figura 24 el diagrama de infraestructura de la aplicación móvil Tribunal Familiar Digital para explicar sobre los elementos tecnológicos integrados dentro de esta.

**Figura 24.** Diagrama tecnológico de la aplicación móvil



Fuente: Elaboración propia (2021).

#### 4.5.10. Capturas de pantalla de la aplicación móvil

Para finalizar con el capítulo 4, se presentará el producto de *software* terminado, es decir, se mostrarán los *screenshots* o capturas de pantalla de todas las pantallas de la aplicación móvil, con el diseño final y funcionando correctamente con el consumo de las APIs, las alertas, vistas web, entre otras funcionalidades

que ofrece la aplicación móvil.

En la figura 25, se puede apreciar la pantalla de bienvenida o *splashscreen* de la aplicación móvil. Esta pantalla muestra el logo y el nombre de la institución gubernamental.

**Figura 25.** Pantalla de presentación animada



**Fuente:** Elaboración propia (2021).

En la figura 26, se puede observar la pantalla de acciones de inicio de la aplicación móvil, la cual le brinda al usuario 2 opciones para ingresar a la aplicación móvil. Si un usuario se encuentra registrado, solo dará clic al botón de iniciar sesión. Sin embargo, si es un usuario nuevo, le dará clic a registrar para estar dentro del sistema móvil.

**Figura 26.** Pantalla de acciones de ingreso



**Fuente:** Elaboración propia (2021).

En la figura 27, se puede observar la pantalla de inicio de sesión, que, como se explicó en la figura anterior (Figura 26), si el usuario se encuentra registrado, podrá iniciar sesión con tan solo ingresar sus credenciales, es decir, deberá ingresar su correo electrónico y contraseña para autorizar la entrada a la aplicación móvil.

**Figura 27.** Pantalla de inicio de sesión



**Fuente:** Elaboración propia (2021).

En la figura 28, se puede visualizar un diálogo de alerta, la cual aparecerá en 2 circunstancias: la primera es cuando el usuario no existe dentro del sistema y la segunda cuando el usuario escribe incorrectamente sus credenciales. Si no llega a pasar ninguna de estas circunstancias, no aparecerá la alerta y el usuario podrá acceder a la aplicación móvil.

**Figura 28.** Pantalla de alerta de inicio de sesión



**Fuente:** Elaboración propia (2021).

En la figura 29, se puede visualizar la pantalla de registro de usuarios, que, como se explicó anteriormente en la pantalla de acciones de registro (Figura 26), si el usuario es nuevo y no cuenta con credenciales, tendrá que registrarse dentro de la aplicación e ingresará sus datos para almacenarse en la base de datos en MySQL.

**Figura 29.** Pantalla de registro de usuarios



**Fuente:** Elaboración propia (2021).

En la figura 30, se puede observar la pantalla de identificación. Una vez que el usuario haya ingresado correctamente sus datos, se redireccionará a esta pantalla, en la cual el usuario tendrá que tomar fotos a las partes delantera y trasera de su credencial de elector.

**Figura 30.** Pantalla para tomar foto a la identificación del usuario



**Fuente:** Elaboración propia (2021).

En la figura 31, se puede apreciar la pantalla de confirmación de registro. Esta será la última pantalla para el registro de usuario, donde el usuario ingresará su correo electrónico y su contraseña, los cuales serán tomados como las credenciales del usuario para iniciar sesión, en vez de tener que realizar nuevamente el registro de usuarios.

**Figura 31.** Pantalla de confirmación de registro



**Fuente:** Elaboración propia (2021).

En la figura 32, se puede observar la pantalla del tablero o *dashboard* de la aplicación móvil. Este tablero contendrá el menú desplegable y los 4 módulos importantes para el funcionamiento de la aplicación móvil, los cuales son: directorio, sentencias públicas, portal y expediente electrónico.

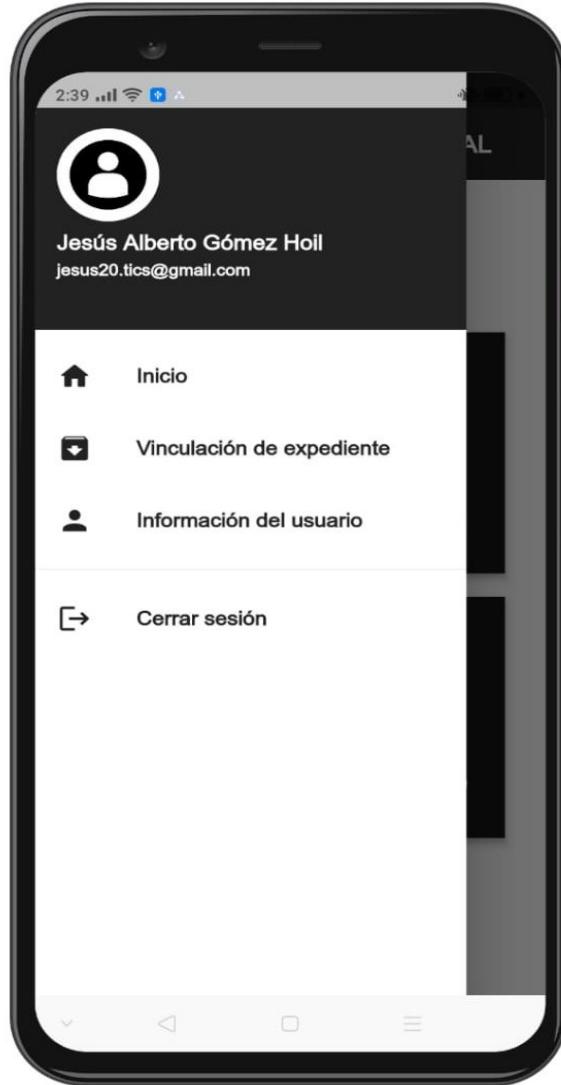
**Figura 32.** Pantalla del *dashboard* de la aplicación móvil



**Fuente:** Elaboración propia (2021).

En la figura 33, se puede apreciar la pantalla del menú desplegable de la aplicación móvil. Este menú tendrá en la parte superior un ícono de usuario y datos como el nombre completo y la dirección de correo electrónico del usuario. La lista de los módulos del menú es: página principal, vinculación de expedientes, información del usuario y el cierre de sesión.

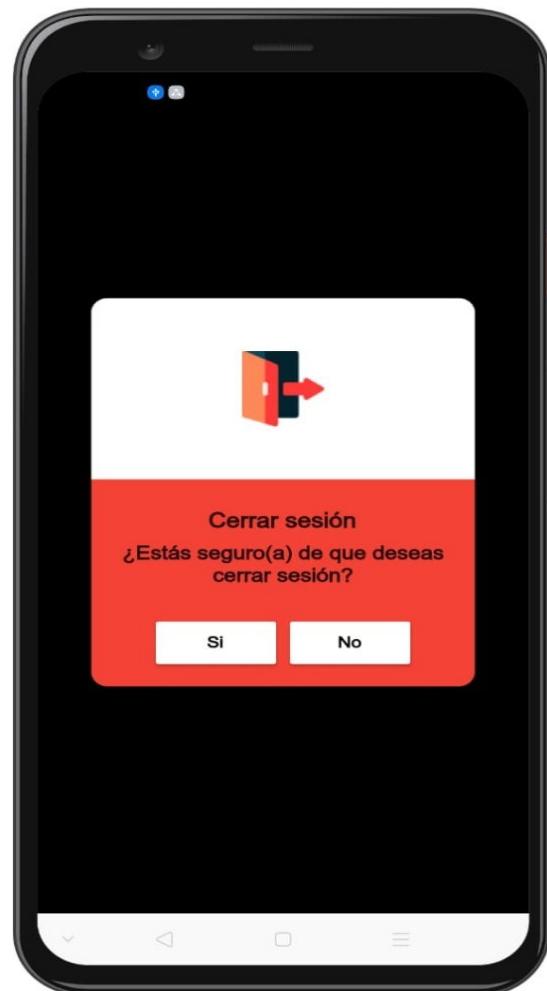
**Figura 33.** Pantalla del menú desplegable



**Fuente:** Elaboración propia (2021).

En la figura 34, se puede apreciar una alerta más, la cual corresponde con el cierre de sesión dentro de la aplicación móvil. Aparecerá una alerta de diálogo después de haber apretado el botón de “Cerrar sesión”, en donde la alerta le dirá al usuario si desea salir de la aplicación móvil. Esta alerta le dará 2 opciones, una de sí y otra de no al usuario. Si el usuario acepta salir de la aplicación móvil, se cerrará la sesión y lo mandará a la pantalla de acciones, pero si el usuario no acepta cerrar sesión, se mantendrá dentro de la aplicación móvil.

**Figura 34.** Pantalla de alarma de cierre de sesión



**Fuente:** Elaboración propia (2021).

En la figura 35, se puede observar la pantalla del módulo directorio, el cual será una pantalla de visualización y contacto, es decir, se verá un directorio con las fotos y datos de los miembros de la segunda sala familiar, y debajo de todos los datos, estarán 3 botones con funciones diferentes. El botón azul será para descargar el Curriculum Vitae del empleado, el botón rojo será para mandar un mensaje por Gmail y el botón verde será para realizar una llamada al empleado.

**Figura 35.** Pantalla del módulo directorio



**Fuente:** Elaboración propia (2021).

En la figura 36, se puede apreciar la pantalla del módulo de sentencias públicas, la cual será una vista web o *webview*, donde se tomó una pantalla de la página web existente de la segunda sala familiar y se realizó la programación para adaptarla a una pantalla de la aplicación móvil. La función de este módulo es visualizar todas las sentencias públicas realizadas en la segunda sala familiar.

**Figura 36.** Pantalla del módulo sentencias públicas



**Fuente:** Elaboración propia (2021).

En la figura 37, se puede visualizar la pantalla del portal de citas, la cual tiene como función agendar una cita con los empleados encargados de la segunda sala familiar. El módulo funciona como un formulario de registro de datos para solicitar una cita con un empleado para consultar los servicios ofrecidos en la segunda sala familiar.

**Figura 37.** Pantalla del módulo portal de citas



**Fuente:** Elaboración propia (2021).

Y, por último, en la figura 38 se puede observar la pantalla del módulo de expedientes electrónicos, en donde el usuario podrá consultar y descargar todos los expedientes disponibles que le pertenezcan.

**Figura 38.** Pantalla del módulo expedientes electrónicos



**Fuente:** Elaboración propia (2021).

## CONCLUSIÓN

Este proyecto se concluyó de manera satisfactoria, ya que las APIs realizadas para el funcionamiento de la aplicación móvil trabajan de manera correcta y la base de datos se logró conectar con estas y con la aplicación móvil. El objetivo general que se mencionó en el capítulo 1, fue desarrollar una aplicación móvil que llevaría por nombre “Tribunal Familiar Digital” la cual tendría un diseño sencillo y funcional con el fin de realizar los trámites y facilitar a los usuarios el acceso a los servicios que brinda la segunda sala familiar del “Tribunal Superior de Justicia de Quintana Roo”. Este objetivo ayudó lo suficiente a tener nuevos conocimientos adquiridos para seguir adelante con el desarrollo de la aplicación móvil y a la culminación de esta dieran buenos resultados.

Así mismo, se pudo llevar poco a poco el cumplimiento de este proyecto con la ayuda de los objetivos específicos como son el funcionamiento de la aplicación mediante el consumo de servicios, usar elementos de diseño móvil mediante programación, realizar la gestión de usuarios mediante el inicio de sesión y el registro de usuarios, etc.

Este proyecto hizo que fuera más fácil conocer sobre la empresa donde se realizaron las estadías, haciendo referencia al capítulo 2 en donde también se propusieron nuevos valores y objetivos bajo el marco de referencia del “Tribunal Superior de Justicia de Quintana Roo”.

De igual forma, en el capítulo 3 se fundamentó teóricamente la programación de la aplicación móvil, permitiendo los resultados positivos, gracias a los conceptos técnicos vistos en este capítulo que permitieron concluir el proyecto.

La experiencia obtenida en este proyecto fue la mejor, ya que se aprendieron muchas partes importantes sobre la vida laboral y sobre llevar un proyecto individualmente y demostrar las habilidades aprendidas durante la estancia en la Universidad Tecnológica de Chetumal.

Todo esto fue posible gracias al uso de los conocimientos adquiridos en la UT Chetumal, ya que sin estos tampoco se pudo hacer posible la conclusión del

proceso de estadías profesionales. Logrando así un proyecto completo y con buenas vistas para llegar a realizar actualizaciones en algún futuro. Todo se llevó paso a paso con la realización de bosquejos o *mockups* para darse una idea de lo que sería la aplicación móvil, la aplicación de la metodología Scrum durante la estadía profesional, identificación de requerimientos para la aplicación móvil, la realización de un diagrama de arquitectura de procesos, implementar diagramas de *software* para describir de manera lógica el sistema móvil, realizar un mapa de sitio de las funcionalidades de la aplicación móvil y de cómo se llevan a cabo las pantallas de esta y la esquematización de un diagrama tecnológico para saber cómo trabajan en conjunto las tecnologías utilizadas en el proyecto, las cuales se describen en el capítulo 3.

Este periodo de estadías fue una experiencia gratificante, llena de conocimientos nuevos y experiencias laborales importantes para el desarrollo profesional del estudiante. A pesar de la contingencia de COVID-19, se moderaron las visitas al tribunal para evitar contagios y mantener una sana distancia. Sin embargo, se pudo trabajar bien y concluir de manera satisfactoria este proyecto gracias a los asesores asignados, ya que por ellos se aprendieron nuevas formas de enseñanza y se llevaron diferentes formas de dirección de un proyecto de TI.

La aplicación móvil “Tribunal Familiar Digital” es un buen proyecto que pinta bien para tener un futuro como una de las aplicaciones gubernamentales más conocidas dentro del estado de Quintana Roo, ya que si se le da continuidad se pueden implementar nuevas funcionalidades y sacar nuevas versiones para que los usuarios puedan tener presente en sus dispositivos móviles esta aplicación y se vaya innovando más y más hasta llegar a ser un proyecto grande.

## REFERENCIAS BIBLIOGRÁFICAS

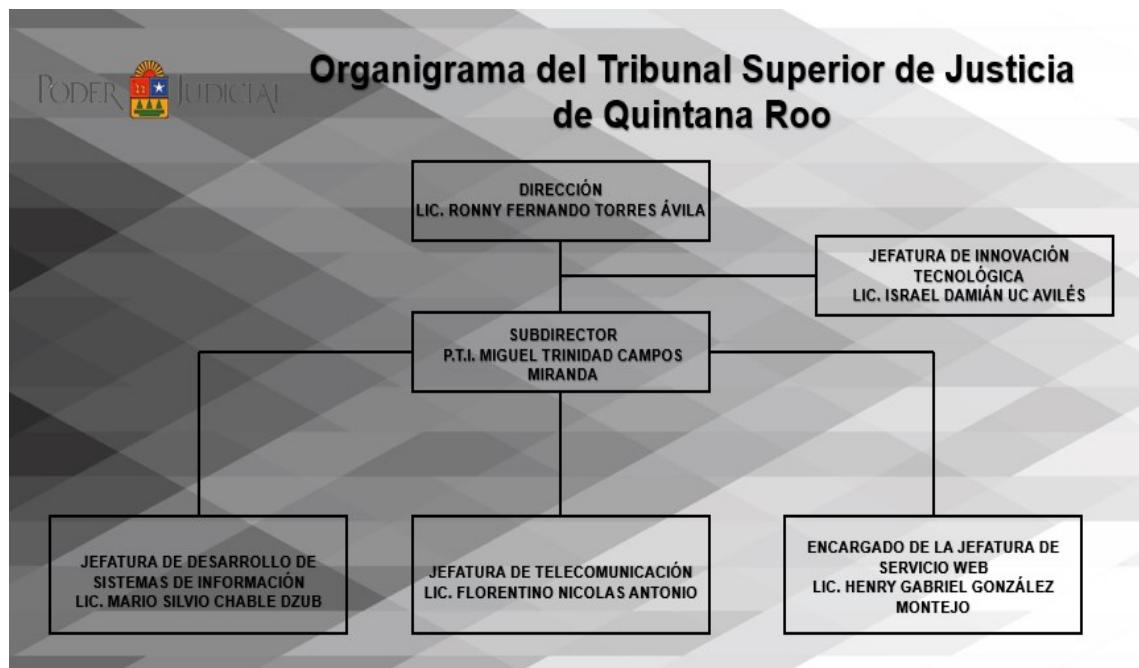
- Alvino, C. (2021). *Estadísticas de la situación digital de México en el 2020–2021*. Branch. Recuperado de <https://branch.com.co/marketing-digital/estadisticas-de-la-situacion-digital-de-mexico-en-el-2020-2021/>
- Ávila, L. S. (2004). *Misión y Visión del Tribunal Superior de Justicia de Quintana Roo*. Recuperado de <https://www.tsjqroo.gob.mx/index.php/nuevapagina/mision-y-vision>
- Ávila, L. S. (2010). *Historia del Tribunal Superior de Justicia de Quintana Roo*. Recuperado de <https://www.tsjqroo.gob.mx/index.php/nuevapagina/historia>
- Baquero, J. M. B. (3 de agosto de 2015). *¿Qué son los webs services y qué tecnología usar en su desarrollo?* [Blog]. Recuperado de <https://www.arsys.es/blog/programacion/web-services-desarrollo/>
- Cuervo, P. V. (2019). *¿Qué es Postman?* Recuperado de <https://www.arquitectoit.com/postman/que-es-postman/>
- De León, Á. (7 de agosto de 2019). *Servidor Apache*. [Blog]. Recuperado de <https://blog.infranetworking.com/que-es-apache-servidor/>
- EcuRed contributors. (2014). *Metodología*. Recuperado de <https://www.ecured.cu/Metodolog%C3%ADA>
- EcuRed contributors. (2020). *Visual Studio Code*. Recuperado de [https://www.ecured.cu/Visual\\_Studio\\_Code](https://www.ecured.cu/Visual_Studio_Code)
- Fernández, E. (27 de junio de 2019). *¿Qué es la metodología Scrum y por qué aplicarla?* [Blog]. Recuperado de <https://www.anfix.com/blog/que-es-la-metodologia-scrum>
- Fernández, H. (2019). *La importancia de las apps móviles en las empresas*. Economía TIC. Recuperado de <https://economiatic.com/importancia-apps-moviles-empresas/>
- Fernández, Y. (2019). *API: qué es y para qué sirve*. Recuperado de <https://www.xataka.com/basics/api-que-sirve>

- Galiana, P. (2021). *Metodología Scrum: qué es y cómo utilizarla para acometer proyectos*. Recuperado de <https://www.iebschool.com/blog/metodologia-scrum-agile-scrum/>
- García, M. (8 de marzo 2020). *¿Qué es XAMPP y como puedo usarlo?* [Blog]. Recuperado de <https://www.nettix.com.pe/blog/web-blog/que-es-xampp-y-como-puedo-usarlo>
- Google Maps (2021). *Tribunal de Justicia, Poder Judicial*. Recuperado de <https://goo.gl/maps/XAo3J8xUFWTvV3389>
- HostGator México. (26 de enero de 2021). *¿Por qué JSON es uno de los estándares más utilizados para el intercambio de información?* [Blog]. Recuperado de <https://www.hostgator.mx/blog/que-es-json/>
- Jiménez, E. (2013). *Desarrollando para Android (I): Android Studio*. Recuperado de <https://androidayuda.com/2013/07/26/desarrollando-para-android-i-android-studio/>
- Merayo, V. M. (2019). *Primeros Pasos con Flutter*. Recuperado de <https://www.adictosaltrabajo.com/2019/04/30/primeros-pasos-con-flutter/>
- Montejo, D. (10 de junio de 2020). *Diferencias entre APP y aplicación web*. [Blog]. Recuperado de <https://rubrika.es/marketing-digital/desarrollo-de-aplicaciones/diferencias-app-aplicacion-web/>
- MurciaEconomía. (2019). *La diferencia entre la app móvil y la app web*. Recuperado de <https://murciaeconomia.com/art/64550/la-diferencia-entre-la-app-movil-y-la-app-web>
- Pascual, J. A. (2021). *Las claves del éxito de Dart, el lenguaje de programación de Google que bate récords de uso*. Recuperado de <https://computerhoy.com/noticias/tecnologia/que-es-dart-para-que-sirve-793487>
- Poder Judicial del Estado de Nuevo León. (2019). *Tribunal Virtual - Apps en Google Play*. Recuperado de [https://play.google.com/store/apps/details?id=com.test.tribunalvirtual&hl=es\\_MX](https://play.google.com/store/apps/details?id=com.test.tribunalvirtual&hl=es_MX)

- Robledano, A. (2019). *Qué es MySQL: Características y ventajas*. Recuperado de <https://openwebinars.net/blog/que-es-mysql/>
- Rodríguez, T. (2011). *Métodos aplicables para el desarrollo de aplicaciones móviles*. Recuperado de <https://www.genbeta.com/desarrollo/metodos-aplicables-para-el-desarrollo-de-aplicaciones-moviles>
- Rokket Labs. (18 de noviembre de 2020). *¿Qué es el desarrollo de aplicaciones móviles?* [Blog]. Recuperado de <https://www.rokkelabs.com/es/blog/que-es-el-desarrollo-de-aplicaciones-moviles>
- Solera, S. (14 de enero de 2020). *La evolución de las aplicaciones móviles para empresas*. [Blog]. Recuperado de <https://www.occamagenciadigital.com/blog/la-evolucion-de-las-aplicaciones-moviles-para-empresas>
- Tribunal Superior de Justicia de Baja California Sur. (2015). *Tribunal Móvil BCS - Apps en Google Play.* Recuperado de [https://play.google.com/store/apps/details?id=mx.itlp.miexpediente&hl=es\\_MX](https://play.google.com/store/apps/details?id=mx.itlp.miexpediente&hl=es_MX)
- Tribunal Superior de Justicia de Quintana Roo. (2021). *Inicio*. Segunda sala familiar. Recuperado de <https://tsjqroo.gob.mx/Micrositios/Salas/Segunda/>
- Universidad Internacional de Valencia. (2019). *Programación SQL: para qué sirve y quién la necesita.* Recuperado de <https://www.universidadviu.com/es/actualidad/nuestros-expertos/programacion-sql-para-que-sirve-y-quien-la-necesita>
- Veleta, M. (2020). *La importancia de las aplicaciones móviles para empresas*. Recuperado de <https://phidevinc.com/es/la-importancia-de-las-aplicaciones-moviles-para-empresas>
- Walterman, R. (2019). *Por qué son tan necesarias las aplicaciones móviles*. Recuperado de <https://walterman.es/por-que-son-tan-necesarias-las-aplicaciones-moviles/>

## ANEXOS

**Anexo 1.** Organigrama del Tribunal Superior de Justicia de Quintana Roo



**Fuente:** Elaboración propia (2021).

## Anexo 2. Código del *main* de la aplicación móvil

```

❶ main.dart ✘
lib > Pantallas > main.dart
1 import 'package:flutter/material.dart';// Importar todos los paquetes o pantallas de la aplicación móvil.
2 import 'package:tribunal_familiar_digital_v2/Pantallas/Acciones_del_Usuario.dart';// Importar el Widget de las acciones de ingreso.
3 import 'package:tribunal_familiar_digital_v2/Pantallas/Confirmar_Registro.dart';// Importar el Widget de confirmación de registro.
4 import 'package:tribunal_familiar_digital_v2/Pantallas/Dashboard_App.dart';// Importar el Widget del dashboard.
5 import 'package:tribunal_familiar_digital_v2/Pantallas/Directorio_Tribunal.dart';// Importar el Widget del directorio.
6 import 'package:tribunal_familiar_digital_v2/Pantallas/Expedientes_Electronicos.dart';// Importar el Widget de expediente electrónico.
7 import 'package:tribunal_familiar_digital_v2/Pantallas/Informacion_del_Usuario.dart';// Importar el Widget de la información del usuario.
8 import 'package:tribunal_familiar_digital_v2/Pantallas/Inicio_de_Sesion.dart';// Importar el Widget del inicio de sesión.
9 import 'package:tribunal_familiar_digital_v2/Pantallas/Menu_Desplegable.dart';// Importar el Widget del menú desplegable.
10 import 'package:tribunal_familiar_digital_v2/Pantallas/Portal_Tribunal.dart';// Importar el Widget del portal.
11 import 'package:tribunal_familiar_digital_v2/Pantallas/Presentacion_Animada.dart';// Importar el Widget de la presentación animada.
12 import 'package:tribunal_familiar_digital_v2/Pantallas/Registro_de_Credencial.dart';// Importar el Widget de registro de credencial.
13 import 'package:tribunal_familiar_digital_v2/Pantallas/Registro_de_Usuario.dart';// Importar el Widget del registro de usuarios.
14 import 'package:tribunal_familiar_digital_v2/Pantallas/Sentencias_Publicas.dart';// Importar el Widget de sentencias públicas.
15
16 void main() {
17   runApp(MyApp());// Correr la aplicación móvil.
18 }
19
20 class MyApp extends StatelessWidget {// Clase MyApp.
21   @override
22   Widget build(BuildContext context) {// Widget constructor
23     return MaterialApp// Función MaterialApp.
24       routes: {// Rutas de las pantallas de la aplicación móvil.
25         "/animacion": (context) => pantalla_animacion(),// Ruta de la pantalla animada.
26         "/acciones": (context) => acciones(),// Ruta de la pantalla de las acciones de ingreso.
27         "/login": (context) => login(),// Ruta de la pantalla del login.
28         "/registro": (context) => singup(),// Ruta de la pantalla de registro.
29         "/home": (context) => homepage(),// Ruta de la pantalla de home.
30         "/credencial": (context) => credencial(),// Ruta de la pantalla de la credencial.
31         "/confirmacion": (context) => confirmar(),// Ruta de la pantalla de confirmación de registro.
32         "/menu": (context) => menu(),// Ruta de la pantalla menú.
33         "/portal": (context) => portal(),// Ruta de la pantalla del portal.
34         "/sentencias": (context) => sentencias(),// Ruta de la pantalla de sentencias públicas.
35         "/directorio": (context) => directorio(),// Ruta de la pantalla del directorio.
36         "/expedientes": (context) => expedientes(),// Ruta de la pantalla de expedientes electrónicos.
37         "/informacion": (context) => informacion(),// Ruta de la pantalla de información de los usuarios.
38       },
39       initialRoute: "animacion",// Declaración de la ruta inicial.
40       title: 'Tribunal Familiar Digital',// Asignar el título de la aplicación móvil.

```

Fuente: Elaboración propia (2021).

### Anexo 3. Código de la presentación animada

```

lib > Pantallas > Presentacion_Animada.dart
1 import 'dart:async';// Importar la librería async de Dart.
2 import 'package:animate_do/animate_do.dart';// Importar la librería animate_do de Dart.
3 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
4 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
5 import 'package:tribunal_familiar_digital_v2/Pantallas/Acciones_del_Usuario.dart';// Importar el Widget de las acciones de ingreso.
6
7
8 class pantalla_animacion extends StatefulWidget {// StatelessWidget de la pantalla animada.
9   @override
10   _pantalla_animacionState createState() => _pantalla_animacionState(); // Crear el estado del Widget de la pantalla animada.
11 }
12
13 class _pantalla_animacionState extends State { // Crear la clase estado que se hereda del StatelessWidget.
14   @override
15   void initState() { // Método void initState,
16     super.initState(); // Realizar uso del initState en un Super.
17     Future.delayed( // Esperar la animación con el Future.delayed.
18       Duration(seconds: 8), // Marcar la duración en 8 segundos de carga.
19       () {
20         Navigator.pushReplacementNamed(context, "/acciones"); // Navegar a la pantalla de acciones de registro.
21       }
22     );
23     super.initState(); // Realizar uso del initState en un Super.
24     Timer(Duration(seconds: 10), ()=> print("Bienvenida realidaza!")); // Esperar 10 segundos para visualizar la pantalla.
25   }
26   @override
27   Widget build(BuildContext context) { // Crear el Widget constructor.
28     return Scaffold( // Función Scaffold para implementar elementos básicos del diseño.
29       body: Center( // Función Center para centrar las funciones de los elementos del diseño.
30         child: fondo_gradiente(), // Widget para el fondo de la aplicación.
31       ),
32     );
33   }
34 }
35
36 Widget fondo_gradiente() { // Widget fondo_gradiente.
37   return Container( // Usar la función Container.
38     height: double.infinity, // Asignar el tamaño de la altura como infinita.
39     width: double.infinity, // Asignar el tamaño del ancho como infinita.
40     decoration: BoxDecoration( // Usar la función BoxDecoration para dar forma al fondo de la pantalla.

```

Fuente: Elaboración propia (2021).

## Anexo 4. Código de las acciones de registro

```

Acciones_del_Usuario.dart X
lib > Pantallas > Acciones_del_Usuario.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3
4 class accion_a_realizar extends StatefulWidget {// StafulWidget de las acciones de registro de usuarios.
5   @override
6   _accion_a_realizarState createState() => _accion_a_realizarState();// Crear el estado del StafulWidget.
7 }
8
9 class _accion_a_realizarState extends State<accion_a_realizar> {// Crear la clase que se hereda al estado de las acciones.
10  @override
11  Widget build(BuildContext context) {// Crear el Widget constructor.
12    return ListView{// Función widget Lista para que se vea todo en forma de lista.
13      children: <Widget>[// Crear los hijos (children) para usar las demás funciones.
14        Container{// Crear el hijo con la función Container.
15          width: double.infinity,// Asignar el ancho como infinito.
16          //height: MediaQuery.of(context).size.height,
17          child: Column{// Usar la función Column.
18            mainAxisAlignment: MainAxisAlignment.spaceBetween,// Asignar la alineación principal con espacios entre líneas.
19            crossAxisAlignment: CrossAxisAlignment.center,// Asignar la alineación transversal como central.
20            children: <Widget>[// Crear los hijos (children) para usar las demás funciones.
21              Column{// Función Column para estructurar los widgets en columna.
22                children: <Widget>[// Crear los hijos (children) para usar las demás funciones.
23                  Image.asset("assets/TSJ.png", height: 330),// Imagen con el logo del tribunal.
24                  Text("Estimado usuario (a), favor de iniciar sesión o \n registrarse si aún no tiene una cuenta.",// Texto.
25                      textAlign: TextAlign.center, style: TextStyle{// Alinear texto en medio.
26                        fontSize: 18,// Tamaño de la fuente.
27                        fontFamily: "Arial",// Tipo de fuente.
28                        fontWeight: FontWeight.bold,// Estilo de la fuente en negritas.
29                        color: Colors.black,// Color de la fuente.
30                      },
31                    ),
32                  ],
33                ),
34              ],
35            ),
36            SizedBox(height: 125.0),// Dar un espacio en la altura.
37            Column{// Función Column para estructurar los widgets en columna.
38              children: <Widget>[// Crear los hijos (children) para usar las demás funciones.
39                MaterialButton{// Uso de un botón de tipo MaterialButton.
40                  minWidth: 200,// Asignar un ancho mínimo.
41                  height: 60.0,// Asignar la altura.
42                  onPressed: (){// Función onPressed para indicar cuál será la acción a realizar al apretar el botón.
43                },
44              ],
45            ),
46          ],
47        ),
48      ],
49    );
50  }
51}

```

**Fuente:** Elaboración propia (2021).

## Anexo 5. Código del inicio de sesión del usuario

```

Inicio_de_Sesion.dart X
lib > Pantallas > Inicio_de_Sesion.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3
4 class iniciar_sesion extends StatefulWidget {// StatefulWidget del inicio de sesión.
5   @override
6   _iniciar_sesionState createState() => _iniciar_sesionState(); // Crear el estado del Widget del inicio de sesión.
7 }
8
9 class _iniciar_sesionState extends State<iniciar_sesion> {// Crear la clase estado que se hereda del StatefulWidget.
10   @override
11   Widget build(BuildContext context) {// Crear el Widget constructor.
12     return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
13       appBar: AppBar(// Función AppBar para la barra de la pantalla.
14         elevation: 0,// Asignar una elevación del contenido en 0.
15         brightness: Brightness.light,// Brillo claro.
16         backgroundColor: Colors.black87,// Color de fondo de la barra.
17         leading: IconButton(// Indicar un ícono para usar como botón de regreso.
18           onPressed: (){// Cuando se presione el botón realizará una navegación.
19             Navigator.pop(context); // Usar el método pop.
20           },
21           icon: Icon(// Indicar un ícono para usar como botón de regreso.
22             Icons.arrow_back_ios_outlined, // Implementar el ícono.
23             size: 20,// Tamaño del ícono.
24             color: Colors.white, // Color del ícono
25           ),
26         ),
27         title: Text("TRIBUNAL FAMILIAR DIGITAL",// Título de la barra.
28           style: TextStyle(// Estilo de la letra.
29             fontWeight: FontWeight.bold, // Letra tipo negritas.
30             fontSize: 20, // Tamaño de la letra.
31             fontFamily: "Arial"// Fuente de la letra.
32           ),
33         ),
34         centerTitle: true, // Centrar el título de la barra.
35       ),
36       body: ListView(// Poner el cuerpo de la pantalla en tipo de lista.
37         children: <Widget>[// Asignar los children.
38           Container(// Usar la función Container.
39             child: Column(// Usar la función Column asignado de un child.
40               children: <Widget>[// Asignar los children.

```

Fuente: Elaboración propia (2021).

## Anexo 6. Código del registro de usuarios

```

Registro_de_Usuario.dart X
lib > Pantallas > Registro_de_Usuario.dart
1 import 'dart:convert';// Importar la librería async de Dart.
2 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
3 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
4 import 'package:http/http.dart' as http;// Importar la librería http.
5 import 'package:tribunal_familiar_digital_v2/Modelos/Modelo_Registro.dart';// Importar el Widget del modelo de registro.
6 import 'package:tribunal_familiar_digital_v2/Servicios/Servicio_Registro.dart';// Importar el Widget del servicio de registro.
7 import 'package:url_launcher/url_launcher.dart';// Importar el Widget url_launcher de Dart.
8
9 class registrar_usuario extends StatefulWidget {// StatelessWidget del registro de usuario.
10   @override
11   _registrar_usuarioState createState() => _registrar_usuarioState(); // Crear el estado del Widget del registro de usuario.
12 }
13
14 class _registrar_usuarioState extends State<registrar_usuario> {// Crear la clase estado que se hereda del StatelessWidget.
15   final GlobalKey<FormState> formKey = GlobalKey<FormState>(); // Instanciar un registro del formulario.
16   final sesionSingup = new RegistroServicios(); // Instanciar un sesionSingup del servicio de registro.
17   RegistroModel sesion = new RegistroModel(); // Instanciar una sesión del modelo de registro.
18
19   bool checkBoxValue = false; // Marcar como falso el valor del checkbox.
20
21   @override
22   Widget build(BuildContext context) { // Crear el Widget constructor.
23     return Scaffold( // Función Scaffold para implementar elementos básicos del diseño.
24       appBar: AppBar( // Barra de la pantalla de la aplicación.
25         elevation: 0, // Elevación del contenido en 0.
26         brightness: Brightness.light, // Brillo claro.
27         backgroundColor: Colors.black87, // Fondo de la barra de la aplicación.
28         leading: IconButton( // Indicar un ícono para usar como botón de regreso.
29           onPressed: () { // Cuando se presione el botón realizará una navegación.
30             Navigator.pushReplacementNamed(context, "/acciones"); // Usar el método Navigator para redireccionar a las acciones.
31           },
32           icon: Icon( // Indicar un ícono para usar como botón de regreso.
33             Icons.arrow_back_ios_outlined, // Implementar el ícono.
34             size: 20, // Tamaño del ícono.
35             color: Colors.white, // Color del ícono.
36           ),
37         ),
38         title: Text("TRIBUNAL FAMILIAR DIGITAL"), // Título de la barra.
39         style: TextStyle( // Estilo de la letra del título.
40           fontWeight: FontWeight.bold, // Letras en negritas.
41         ),
42       ),
43     );
44   }
45 }

```

Fuente: Elaboración propia (2021).

## Anexo 7. Código del registro de la credencial

```

Registro_de_Credencial.dart X
lib > Pantallas > Registro_de_Credencial.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3 import 'package:image_picker/image_picker.dart';// Importar la librería image_picker de Dart.
4 import 'dart:async';// Importar la librería async de Dart.
5 import 'dart:io';// Importar la librería io de Dart.
6
7 class registrar_credencial extends StatefulWidget {// StatefulWidget del registro de credencial.
8   @override
9     _registrar_credencialState createState() => _registrar_credencialState(); // Crear el estado del Widget del registro de credencial.
10 }
11
12 class _registrar_credencialState extends State<registrar_credencial> {// Crear la clase estado que se hereda del StatefulWidget.
13   @override
14   Widget build(BuildContext context) { // Crear el Widget constructor.
15     return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
16       appBar: AppBar(// Barra de la pantalla de la aplicación.
17         elevation: 0,// Elevación del contenido en 0.
18         brightness: Brightness.light,// Brillo claro.
19         backgroundColor: Colors.black87,// Color de fondo de la barra de la aplicación.
20         leading: IconButton(// Indicar un ícono para usar como botón de regreso.
21           onPressed: (){// Cuando se presione el botón realizará una navegación.
22             Navigator.pop(context); // Usar el método pop del Navigator.
23           },
24           icon: Icon(// Indicar un ícono para usar como botón de regreso.
25             Icons.arrow_back_ios_outlined, // Implementar el ícono.
26             size: 20,// Tamaño del ícono.
27             color: Colors.white, // Color del ícono.
28           ),
29         ),
30         title: Text("TRIBUNAL FAMILIAR DIGITAL",// Título de la barra.
31           style: TextStyle(// Estilo del título.
32             fontWeight: FontWeight.bold, // Letras en negritas.
33             fontSize: 20, // Tamaño de la letra.
34             fontFamily: "Arial"// Fuente de la letra.
35           ),
36         ),
37         centerTitle: true, // Centrar el título.
38       ),
39       body: ListView(// Poner el cuerpo de la pantalla en forma de lista.
40         children: <Widget>[] // Asignar los children.
41       )
42     );
43   }
44 }

```

**Fuente:** Elaboración propia (2021).

## Anexo 8. Código de la confirmación de registro

```

Confirmar_Registro.dart ×

lib > Pantallas > Confirmar_Registro.dart
1 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
2 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
3
4 class confirmar_usuario extends StatefulWidget {// StatefulWidget para confirmar al usuario.
5   @override
6   _confirmar_usuarioState createState() => _confirmar_usuarioState();// Crear el estado del Widget de confirmar usuario.
7 }
8
9 class _confirmar_usuarioState extends State<confirmar_usuario> {// Crear la clase estado que se hereda del StatefulWidget.
10   @override
11   Widget build(BuildContext context) {// Crear el Widget constructor.
12     return Scaffold{// Función Scaffold para implementar elementos básicos del diseño.
13       appBar: AppBar{// Función AppBar para la barra de la pantalla.
14         elevation: 0,// Asignar una elevación del contenido en 0.
15         brightness: Brightness.light,// Asignar un brillo claro.
16         backgroundColor: Colors.black87,// Asignar el color del fondo de la barra.
17         leading: IconButton{// Indicar un botón para regresar.
18           onPressed: (){// Cuando se presione el botón realizará una navegación.
19             Navigator.pop(context); // Usar el método pop.
20           },
21           icon: Icon{// Indicar un ícono para usar como botón de regreso.
22             Icons.arrow_back_ios_outlined,// Implementar el ícono.
23             size: 20,// Tamaño del ícono.
24             color: Colors.white,// Color de ícono.
25           },
26         ),
27         title: Text("TRIBUNAL FAMILIAR DIGITAL",// Título de la barra de la aplicación.
28           style: TextStyle{// Estilo de la letra.
29             fontWeight: FontWeight.bold,// Letra de tipo negritas.
30             fontSize: 20,// Tamaño de la letra.
31             fontFamily: "Arial");// Tipo de letra.
32           },
33         ),
34         centerTitle: true,// Indicar que se centrará el título.
35       },
36       body: ListView{// Asignar como lista el cuerpo de la pantalla.
37         children: <Widget>[]// Asignar los children.
38         Container{// Uso de la función Container.
39           child: Column{// Uso de la función Column.
40             children: <Widget>[]// Asignar los children.
}

```

Fuente: Elaboración propia (2021).

## Anexo 9. Código del *dashboard*

```

Dashboard_App.dart ×

lib > Pantallas > Dashboard_App.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3 import 'package:tribunal_familiar_digital_v2/Pantallas/Menu_Desplegable.dart';// Importar el Widget del menú desplegable.
4
5 class pantalla_home extends StatefulWidget { // StatelessWidget del dashboard.
6   @override
7   _pantalla_homeState createState() => _pantalla_homeState(); // Crear el estado del Widget del dashboard.
8 }
9
10 class _pantalla_homeState extends State<pantalla_home> { // Crear la clase estado que se hereda del StatelessWidget.
11   @override
12   Widget build(BuildContext context) { // Crear el Widget constructor.
13
14     var size = MediaQuery.of(context).size; // Asignar el tamaño con el método MediaQuery.
15
16     var estilo_letra1 = TextStyle( // Asignar el primer estilo de letra para los títulos de la barra.
17       fontSize: 20, // Tamaño de la letra.
18       color: Colors.white, // Color de la letra.
19       fontFamily: "Arial", // Fuente de las letras.
20       fontWeight: FontWeight.bold, // Letras en negritas.
21     );
22
23     var estilo_letra2 = TextStyle( // Asignar el segundo estilo de letra para los párrafos y oraciones.
24       fontSize: 35, // Tamaño de la fuente.
25       color: Colors.black87, // Color de la fuente.
26       fontFamily: "Arial", // Fuente de las letras.
27       fontWeight: FontWeight.bold, // Letras en negritas.
28     );
29
30     return Scaffold(
31       drawer: navigation_drawer(), // Asignar como menú el Widget navigation_drawer.
32       appBar: AppBar( // Barra de la pantalla de la aplicación.
33         backgroundColor: Colors.black87, // Color del fondo de la barra.
34         elevation: 0, // Elevación del contenido en 0.
35         brightness: Brightness.light, // Brillo claro.
36         title: Text("TRIBUNAL FAMILIAR DIGITAL", // Texto.
37           style: estilo_letra1, // Estilo del texto.
38         ),
39         centerTitle: true, // Centrar el título de la barra.
40       ),

```

**Fuente:** Elaboración propia (2021).

## Anexo 10. Código del menú desplegable

```

❶ Menu_Desplegable.dart X
lib > Pantallas > ❷ Menu_Desplegable.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3 import 'package:tribunal_familiar_digital_v2/Pantallas/Dashboard_App.dart';// Importar el Widget del dashboard.
4
5 class menu_desplegable extends StatefulWidget {// StatelessWidget del menú desplegable.
6   @override
7     _menu_desplegableState createState() => _menu_desplegableState(); // Crear el estado del Widget del menú desplegable.
8 }
9
10 class _menu_desplegableState extends State<menu_desplegable> {// Crear la clase estado que se hereda del StatelessWidget.
11   @override
12     Widget build(BuildContext context) {// Crear el Widget constructor.
13       return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
14         drawer: navigation_drawer(),// Asignar como menú el Widget navigation_drawer.
15         body: ListView(// Poner el cuerpo de la pantalla en tipo de lista.
16           children: <Widget>[]// Asignar los children.
17             | homepage(),// Insertar el Widget del homepage.
18             ],
19           ),
20         );
21     }
22   }
23
24 class menu extends StatelessWidget {// StatelessWidget del menú.
25   @override
26     Widget build(BuildContext context) {// Crear el Widget constructor.
27       return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
28         body: Center(// Función Center para centrar las funciones de los elementos del diseño.
29           child: fondo_gradiante(),// Insertar el Widget fondo_gradiante.
30           ),
31         );
32     }
33
34 Widget fondo_gradiante() {// Widget fondo_gradiante.
35   return Container(// Usar la función Container.
36     height: double.infinity,// Asignar el tamaño de la altura como infinita.
37     width: double.infinity,// Asignar el tamaño de lo ancho como infinita.
38     decoration: BoxDecoration(// Usar la función BoxDecoration para dar forma al fondo de la pantalla.
39       gradient: LinearGradient(// Usar la función LinearGradient para dar forma y diseño al fondo de la pantalla.
40         begin: Alignment.topRight,// Alinear al inicio a la derecha.

```

**Fuente:** Elaboración propia (2021).

## Anexo 11. Código del módulo de directorio del tribunal

```

lib > Pantallas > Directorio_Tribunal.dart
1 import 'dart:convert';// Importar la librería dart:convert para convertir JSON en Dart.
2 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
3 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
4 import 'package:tribunal_familiar_digital_v2/Modelos/Modelo_Directorio.dart';// Importar el Widget del modelo.
5 import 'package:tribunal_familiar_digital_v2/Pantallas/Menu_Desplegable.dart';// Importar el Widget del menú desplegable.
6 import 'package:http/http.dart' as http;// Importar la librería http.
7 import 'package:url_launcher/url_launcher.dart';// Importar la librería url_launcher.

8
9 class directorio_tsj extends StatefulWidget {// StatelessWidget del directorio.
10   @override
11   _directorio_tsjState createState() => _directorio_tsjState(); // Crear el estado del Widget del directorio.
12 }
13
14 class _directorio_tsjState extends State<directorio_tsj> { // Crear la clase estado que se hereda del StatelessWidget.
15
16   DirectorioModel registros = new DirectorioModel(); // Instanciar un registro del modelo del directorio.
17   List<DirectorioModel> data = List<DirectorioModel>(); // Crear una lista e instanciar del modelo directorio.
18
19   Future<List<DirectorioModel>> tomar_datos(DirectorioModel registros) async { // Realizar un Future de la lista del modelo del directorio.
20     //Map parametros = ModalRoute.of(context).settings.arguments;
21     final String _url = 'http://localhost/TribunalAPP/public/api'; // Declarar la url de la API.
22     final String url = '${_url}/directorio'; // Declarar la ruta directa de la API junto con la anterior.
23     // Declarar una respuesta que se hará por medio del GET con el protocolo HTTP.
24     var resp = await http.get(Uri.parse(url), headers: {"Content-type": "application/json", "Accept": "application/json"});
25
26     var informacion = List<DirectorioModel>(); // Declarar una variable información e instanciar con la lista del modelo del directorio.
27
28     print('Todo Bien');// Imprimir que todo salió bien.
29     print(url);// Imprimir la url completa.
30
31     if(resp.statusCode == 200){ // Condición if que indica que si el statusCode de la respuesta es 200, realizará las siguientes operaciones.
32       var decodedData = json.decode(resp.body); // Decodificar la respuesta en JSON.
33       print(resp.statusCode); // Imprimir el statusCode de la respuesta.
34       print(decodedData); // Imprimir la respuesta decodificada.
35       for(decodedData in decodedData){ // Entrar a un foreach para recorrer todos los datos.
36         informacion.add(DirectorioModel.fromJson(decodedData)); // Añadir información del modelo directorio con los datos decodificados.
37       }
38     }
39   }
40   return informacion; // Regresar la información.
}

```

Fuente: Elaboración propia (2021).

## Anexo 12. Código del módulo de sentencias públicas

```

❶ Sentencias_Publicas.dart X
lib > Pantallas > ❷ Sentencias_Publicas.dart
1 import 'dart:async';// Importar la librería async de Dart.
2 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
3 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
4 import 'package:webview_flutter/webview_flutter.dart';// Importar la librería webview_flutter de Dart.
5
6 class sentencias_tsj extends StatefulWidget {// StafulWidget de las sentencias públicas.
7   @override
8     _sentencias_tsjState createState() => _sentencias_tsjState(); // Crear el estado del Widget de las sentencias públicas.
9 }
10
11 class _sentencias_tsjState extends State<sentencias_tsj> { // Crear la clase estado que se hereda del StafulWidget.
12
13   final Completer<WebViewController> _controller = Completer<WebViewController>(); // Instancias el controlador para la vista web.
14
15   @override
16   Widget build(BuildContext context) { // Crear el Widget constructor.
17
18     var estilo_letral = TextStyle( // Asignar el primer estilo de letra para los títulos de la barra.
19       fontSize: 18, // Tamaño de la letra.
20       color: Colors.white, // Color de la letra.
21       fontFamily: "Arial", // Fuente de las letras.
22       fontWeight: FontWeight.bold, // Letras en negritas.
23     );
24
25     return Scaffold( // Función Scaffold para implementar elementos básicos del diseño.
26       appBar: AppBar( // Barra de la pantalla de la aplicación.
27         backgroundColor: Colors.black87, // Color del fondo de la barra de la aplicación.
28         elevation: 0, // Elevar el contenido a 0.
29         brightness: Brightness.light, // Brillo claro.
30         leading: IconButton( // Indicar un ícono para usar como botón de regreso.
31           onPressed: () { // Cuando se presione el botón realizará una navegación.
32             Navigator.pushNamed(context, "/home"); // Usar el método Navigator para redireccionar al dashboard.
33           },
34           icon: Icon( // Indicar un ícono para usar como botón de regreso.
35             Icons.arrow_back_ios_outlined, // Implementar el ícono.
36             size: 20, // Tamaño del ícono
37             color: Colors.white, // Color del ícono.
38           ),
39         ),
40         title: Text("SENTENCIAS PÚBLICAS"), // Título de la barra.

```

**Fuente:** Elaboración propia (2021).

### Anexo 13. Código del módulo de portal de citas

```

Portal_Tribunal.dart
lib > Pantallas > Portal_Tribunal.dart
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3
4 class portal_tsj extends StatefulWidget {// StatefulWidget del portal de citas.
5   @override
6   _portal_tsjState createState() => _portal_tsjState(); // Crear el estado del Widget del portal de citas.
7 }
8
9 class _portal_tsjState extends State<portal_tsj> {// Crear la clase estado que se hereda del StatefulWidget.
10
11   var lista_motivo = ["Cita con Magistrado", "Cita con Secretario de Acuerdos"];// Declarar una lista de motivos de citas.
12   String vista_motivo = "Seleccione un motivo";// Declarar lo que se reflejará por primera vez en la lista desplegable.
13
14   var lista_asunto = ["Protesta", "Ratificación", "Desahogo de pruebas", "Personal"];// Declarar una lista de asuntos.
15   String vista_asunto = "Seleccione un asunto";// Declarar lo que se reflejará por primera vez en la lista desplegable.
16
17   var lista_tipo_de_documento = ["Toca", "Carpeta", "Cuadernillo de amparo de toca", "Cuadernillo de amparo de carpeta",
18   "Expedientillo", "Expediente", "Cuadernillo de queja"];// Declarar una lista de tipos de documentos.
19   String vista_tipo_de_documento = "Seleccione el tipo de documento";// Declarar lo que se reflejará por primera vez en la lista desplegable.
20
21   @override
22   Widget build(BuildContext context) {// Crear el Widget constructor.
23
24     var estilo_letra1 = TextStyle();// Asignar el primer estilo de letra para los títulos de la barra.
25     fontSize: 18,// Tamaño de la letra.
26     color: Colors.white,// Color de la letra.
27     fontFamily: "Arial",// Fuente de las letras.
28     fontWeight: FontWeight.bold,// Letras en negritas,
29   );
30
31     var estilo_letra2 = TextStyle();// Asignar el segundo estilo de letra para los párrafos.
32     fontSize: 20,// Tamaño de la letra.
33     color: Colors.black87,// Color de la letra.
34     fontFamily: "Arial",// Fuente de las letras.
35     fontWeight: FontWeight.bold,// Letras en negritas,
36   );
37
38   return Scaffold// Función Scaffold para implementar elementos básicos del diseño.
39     appBar: AppBar// Barra de la pantalla de la aplicación.
40       backgroundColor: Colors.black87,// Color del fondo de la barra de la aplicación.

```

Fuente: Elaboración propia (2021).

## Anexo 14. Código del módulo de expedientes electrónicos

```

  ⚒ Expedientes_Electronicos.dart ×
lib > Pantallas > ⚒ Expedientes_Electronicos.dart
1   import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2   import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3   import 'package:tribunal_familiar_digital_v2/Pantallas/Menu_Desplegable.dart';// Importar el Widget del menú desplegable.
4
5   class expedientes_tsj extends StatefulWidget {// StatelessWidget del expediente.
6       @override
7       _expedientes_tsjState createState() => _expedientes_tsjState(); // Crear el estado del Widget del expediente.
8   }
9
10  class _expedientes_tsjState extends State<expedientes_tsj> { // Crear la clase estado que se hereda del StatelessWidget.
11
12      var estilo_letra1 = TextStyle(// Asignar el primer estilo de letra para los títulos de la barra.
13          fontSize: 18, // Tamaño de la letra.
14          color: Colors.white, // Color de la letra.
15          fontFamily: "Arial", // Fuente de las letras.
16          fontWeight: FontWeight.bold, // Letras en negritas.
17      );
18
19      @override
20      Widget build(BuildContext context) { // Crear el Widget constructor.
21          return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
22              appBar: AppBar(// Función AppBar para la barra de la pantalla.
23                  backgroundColor: Colors.black87, // Asignar el color del fondo de la barra.
24                  elevation: 0, // Asignar una elevación del contenido en 0.
25                  brightness: Brightness.light, // Asignar un brillo claro.
26                  leading: IconButton(// Indicar un ícono para usar como botón de regreso.
27                      onPressed: (){} // Cuando se presione el botón realizará una navegación.
28                      | Navigator.pushNamed(context, "/home"); // Usar el método Navigator para redireccionar al dashboard.
29                  ),
30                  icon: Icon(// Indicar un ícono para usar como botón de regreso.
31                      Icons.arrow_back_ios_outlined, // Implementar el ícono.
32                      size: 20, // Tamaño del ícono.
33                      color: Colors.white, // Color del ícono.
34                  ),
35                  ),
36                  title: Text("EXPEDIENTES ELECTRÓNICOS", // Título de la barra.
37                      | style: estilo_letra1, // Estilo para la letra de la barra.
38                  ),
39                  centerTitle: true, // Centrar el título de la barra.
40              ),

```

**Fuente:** Elaboración propia (2021).

## Anexo 15. Código de la información del usuario

```

lib > Pantallas > Informacion_del_Usuario.dart X
1 import 'package:flutter/cupertino.dart';// Importar la librería Cupertino de Dart.
2 import 'package:flutter/material.dart';// Importar la librería material.dart de Dart.
3
4 class informacion_usuario extends StatefulWidget {// StatelessWidget de la información del usuario.
5   @override
6   _informacion_usuarioState createState() => _informacion_usuarioState(); // Crear el estado del Widget de la información del usuario.
7 }
8
9 class _informacion_usuarioState extends State<informacion_usuario> {// Crear la clase estado que se hereda del StatelessWidget.
10   @override
11   Widget build(BuildContext context) { // Crear el Widget constructor.
12
13     var estilo_letral = TextStyle(// Asignar el primer estilo de letra para los títulos de la barra.
14       fontSize: 18,// Tamaño de la letra.
15       color: Colors.white,// Color de la letra.
16       fontFamily: "Arial",// Fuente de las letras.
17       fontWeight: FontWeight.bold,// Letras en negritas.
18     );
19
20     return Scaffold(// Función Scaffold para implementar elementos básicos del diseño.
21       appBar: AppBar(// Función AppBar para la barra de la pantalla.
22         backgroundColor: Colors.black87,// Color del fondo de la barra.
23         elevation: 0,// Elevación del contenido en 0.
24         brightness: Brightness.light,// Brillo claro.
25         leading: IconButton(// Indicar un ícono para usar como botón de regreso.
26           onPressed: (){// Cuando se presione el botón realizará una navegación.
27             Navigator.pushNamed(context, "/menu"); // Usar el método Navigator para redireccionar al dashboard.
28           },
29           icon: Icon(// Indicar un ícono para usar como botón de regreso.
30             Icons.arrow_back_ios_outlined,// Implementar el ícono.
31             size: 20,// Tamaño del ícono.
32             color: Colors.white,// Color del ícono.
33           ),
34         ),
35         title: Text("INFORMACIÓN DEL USUARIO",// Título de la barra.
36           style: estilo_letral, // Estilo para la letra de la barra.
37         ),
38         centerTitle: true, // Centrar el título de la barra.
39       ),
40       body: ListView(// Poner el cuerpo de la pantalla en forma de lista con la función ListView.

```

**Fuente:** Elaboración propia (2021).