



UNIVERSIDAD CATÓLICA DEL MAULE

Facultad de Ciencias de la Ingeniería

Profesor Guía

Escuela de Ingeniería Civil Informática

Dr. Marco Toranzo

SIAP: SISTEMA DE AGENDA E INFORMACIÓN PARA PSICÓLOGA

LUIS JOAQUÍN CAROCA ARAYA

DAVID ALFREDO GONZÁLEZ ELGUETA

Memoria para optar al

Título Profesional de Ingeniero Ejecución en Computación e Informática

Talca, Octubre 2022

UNIVERSIDAD CATÓLICA DEL MAULE
FACULTAD DE CIENCIAS DE LA INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL INFORMÁTICA

MEMORIA PARA OPTAR AL
TÍTULO PROFESIONAL DE INGENIERO EJECUCIÓN EN COMPUTACIÓN
INFORMÁTICA

SIAP: SISTEMA DE AGENDA E INFORMACIÓN PARA PSICÓLOGA

LUIS JOAQUÍN CAROCA ARAYA

DAVID ALFREDO GONZÁLEZ ELGUETA

COMISIÓN EXAMINADORA

FIRMA

PROFESOR GUÍA

DR. MARCO TORANZO CÉSPEDES

PROFESOR COMISIÓN

PROFESOR COMISIÓN

NOTA FINAL EXAMEN DE TÍTULO

Talca, Octubre 2022

Agradecimientos

Agradecemos a nuestras familias, las cuales nos han apoyado desde el inicio de manera incondicional, entregándonos su confianza y apoyándonos durante toda nuestra formación como profesionales. Agradecerles también por el sacrificio que han hecho por nosotros y por siempre seguir creyendo tanto en nosotros como en nuestras capacidades para poder cumplir nuestros sueños.

Agradecemos a nuestros amigos y compañeros que nos han acompañado durante todo este proceso, compartiendo buenas experiencias, aprendizajes y apoyándonos los unos a los otros en todo momento.

Finalmente, agradecemos al profesor Marco Toranzo, por brindarnos tanto su apoyo como conocimientos y distintas herramientas durante nuestra formación profesional. Además de confiar en nuestras habilidades, como también el guiarnos durante el desarrollo de SIAP.

Sumario

Este proyecto de memoria aborda la creación de una plataforma web para una psicóloga particular, con el cual sus pacientes podrán solicitar horas directamente desde la plataforma, sin la necesidad de contactarse con ella para agendar una atención. Además, el sistema permitirá a los pacientes añadirse dentro de una lista de espera en caso de que la agenda se encuentre actualmente sin cupo. Finalmente, existirá la posibilidad de una suscripción vía correo electrónico, que permitirá a la psicóloga enviar artículos de interés, noticias, datos importantes, etc. a los suscritos.

Se investigaron tecnologías y metodologías o marcos de trabajo más recientes relacionados con el desarrollo web, en busca de la mejor solución posible. De esta forma se generan diversos documentos, un modelo relacional, *mockups* y una arquitectura cliente-servidor con tecnologías actuales como Laravel y Vue.js.

Se espera que SIAP pueda ser una herramienta que facilite la metodología de trabajo de la profesional y que pueda ser una aplicación que vaya evolucionando con el tiempo a medida que se requieran nuevas funciones.

Índice General

Introducción	11
1.1. Introducción.....	12
1.2. Problemática.....	12
1.3. Objetivo General	13
1.4. Objetivos Específicos	13
1.5. Motivación	14
1.6. Contribuciones.....	14
1.7. Alcance de la memoria.....	15
1.8. Organización del documento.....	15
Marco Teórico	16
2.1. Tecnologías.....	17
2.1.1. Backend	17
2.1.2. Frontend	21
2.1.3. API REST.....	21
2.1.4. Control de Versiones	23
2.1.5. Marco de Trabajo Basado en Procesos Ágiles.....	23
2.1.6. Jira	23
Marco de Trabajo y Propuesta de Solución	25
3.1. Marco de Trabajo Scrum	26
3.1.1. Product Backlog	26
3.1.2. Sprint Planning.....	26
3.1.3. Sprint Backlog	27
3.1.4. Team	27
3.1.5. Daily Scrum	27
3.1.6. Sprint Review.....	27

3.1.7. Sprint Retrospective.....	28
3.1.8. Increment	28
3.2. Adaptación del Marco de Trabajo Scrum	28
3.2.1. Adaptación de Product Backlog y Sprint Backlog	28
3.2.2. Adaptacion Scrum Team	29
3.2.3. Adaptacion de Daily Scrum y Sprint Review	29
3.3. Propuesta de la Solución.....	30
3.3.1. Modelo de Datos	30
3.3.2. Arquitectura	31
3.3.3. Muestra del sistema.....	32
Implementación del Sistema	34
4.1. Product Backlog	35
4.1. Sprint N° 1.....	36
4.1.1. Sprint N° 1.....	36
4.1.2. Sprint Review.....	36
4.1.3. Sprint Retrospective.....	37
4.1.4. Increment	37
4.2.1. Sprint N° 2.....	38
4.2.2. Sprint Review.....	39
4.2.3. Sprint Retrospective.....	39
4.2.4. Increment	39
4.3.1. Sprint N° 3.....	40
4.3.2. Sprint Review.....	41
4.3.3. Sprint Retrospective.....	41
4.3.4. Increment	41
4.4.1. Sprint N° 4.....	42

4.4.2.	Sprint Review	42
4.4.3.	Sprint Retrospective	42
4.4.4.	Increment	42
4.5.1.	Sprint N° 5	43
4.5.2.	Sprint Review	43
4.5.3.	Sprint Retrospective	43
4.5.4.	Increment	44
4.6.1.	Sprint N° 2	45
4.6.2.	Sprint Review	45
4.6.3.	Sprint Retrospective	45
4.6.4.	Increment	46
4.7.1.	Sprint N° 7	46
4.7.2.	Sprint Review	47
4.7.3.	Sprint Retrospective	47
4.7.4.	Increment	47
4.8.1.	Sprint N° 8	48
4.8.2.	Sprint Review	50
4.8.3.	Sprint Retrospective	50
4.8.4.	Increment	50
4.9.1.	Sprint N° 9	54
4.9.2.	Sprint Review	55
4.9.3.	Sprint Retrospective	55
4.10.1.	Sprint N° 10	56
4.10.2.	Sprint Review	57
4.10.3.	Sprint Retrospective	57
4.11.1.	Sprint N° 11	59

4.11.2.	Sprint Review	60
4.11.3.	Sprint Retrospective	60
4.12.1.	Sprint N° 12.....	60
4.12.2.	Sprint Review	62
4.12.3.	Sprint Retrospective	62
4.13.1.	Sprint N° 13.....	62
4.13.2.	Sprint Review	64
4.13.3.	Sprint Retrospective	64
4.14.1.	Sprint N° 14.....	72
4.14.2.	Sprint Review	74
4.14.3.	Sprint Retrospective	74
Conclusiones y Trabajos Futuros		89
5.1.	Conclusiones.....	90
5.2.	Trabajos Futuros	91
5.3.	Referencias	92
5.4.	Anexos.....	93
5.5.	Anexos Digitales	93

Índice de Figuras

FIGURA 1.1: RESULTADOS OBTENIDOS EN TEST DE RESPUESTAS POR SEGUNDO [PSF2019].....	17
FIGURA 1.2: RESULTADOS OBTENIDOS EN TEST DE USO DE MEMORIA [PSF2019].....	18
FIGURA 1.3: RESULTADOS OBTENIDOS EN TEST DE TIEMPO DE RESPUESTA [PSF2019].....	18
FIGURA 1.4: RESULTADOS OBTENIDOS AL REVISAR CREACIÓN DE ARCHIVOS [PSF2019].....	18
FIGURA 1.5: EJEMPLOS DE CONSULTAS SQL [ELABORACIÓN PROPIA]	20
FIGURA 1.6: DIAGRAMA DE FUNCIONAMIENTO DE API-REST [DIS2019].....	22
FIGURA 2.1: FLUJO DE METODOLOGÍA SCRUM [ARREDONDO, E. (2016). SCRUM]	26
FIGURA 2.2: BACKLOG DE SPRINT 7 Y BACKLOG DE SIAP [ELABORACIÓN PROPIA]	29
FIGURA 2.3: MODELO RELACIONAL A UTILIZAR POR SIAP [ELABORACIÓN PROPIA]	30
FIGURA 2.4: ARQUITECTURA UTILIZADA POR SIAP [ELABORACIÓN PROPIA]	31
FIGURA 2.5: VISTA REGISTRO DE USUARIO [ELABORACIÓN PROPIA]	32
FIGURA 2.6: VISTA REPORTE DE USUARIOS [ELABORACIÓN PROPIA]	33
FIGURA 3.1: BACKLOG PARA EL DESARROLLO DE SIAP [ELABORACIÓN PROPIA]	35
FIGURA 3.2: BOARD DE UN SPRINT EN JIRA [ELABORACIÓN PROPIA]	35
FIGURA 3.3: PORTADA DE DOCUMENTO [ELABORACIÓN PROPIA]	38
FIGURA 3.4: MODELO RELACIONAL DE LA BASE DE DATOS DE SIAP [ELABORACIÓN PROPIA]	40
FIGURA 3.5: MOCKUP DE UN FORMULARIO EN SIAP [ELABORACIÓN PROPIA].....	44
FIGURA 3.6: DISEÑO DE HEADER [ELABORACIÓN PROPIA]	48
FIGURA 3.7: DISEÑO DE FOOTER [ELABORACIÓN PROPIA]	48
FIGURA 3.8: VISTA INICIO EN SIAP [ELABORACIÓN PROPIA]	51
FIGURA 3.9: PORTADA DEL DOCUMENTO DE MEMORIA [ELABORACIÓN PROPIA].....	52
FIGURA 3.10: PORTADA DE MANUAL DE DESARROLLADOR [ELABORACIÓN PROPIA].....	53
FIGURA 3.11: VISTA SERVICIOS [ELABORACIÓN PROPIA]	56
FIGURA 3.12: VISTA CONTACTO [ELABORACIÓN PROPIA]	58
FIGURA 3.13: VISTA AGENDAR HORA [ELABORACIÓN PROPIA]	65
FIGURA 3.14: VISTA LISTA DE ESPERA [ELABORACIÓN PROPIA]	66
FIGURA 3.15: CONEXIÓN DE BASE DE DATOS [ELABORACIÓN PROPIA]	66
FIGURA 3.16: ARCHIVOS DE MIGRACIONES DE BASE DE DATOS [ELABORACIÓN PROPIA]	67
FIGURA 3.17: PORTADA DE MANUAL DE USUARIO [ELABORACIÓN PROPIA]	68
FIGURA 3.18: VISTA MÉTODOS DE PAGO [ELABORACIÓN PROPIA]	69
FIGURA 3.19: VISTA MÉTODO DE ATENCIÓN [ELABORACIÓN PROPIA]	69
FIGURA 3.20: VISTA ¿CÓMO AGENDAR UNA HORA? [ELABORACIÓN PROPIA]	70
FIGURA 3.21: VISTA ¿CÓMO INGRESAR A SESIÓN? [ELABORACIÓN PROPIA]	71
FIGURA 3.22: VISTA PERFIL [ELABORACIÓN PROPIA]	75
FIGURA 3.23: VISTA LOGIN [ELABORACIÓN PROPIA].....	76

FIGURA 3.24: VISTA RECUPERAR CONTRASEÑA [ELABORACIÓN PROPIA]	76
FIGURA 3.25: VISTA OPCIONES DE ADMINISTRADOR [ELABORACIÓN PROPIA]	77
FIGURA 3.26: VISTA REPORTE DE USUARIOS [ELABORACIÓN PROPIA]	78
FIGURA 3.27: VISTA REPORTE DE SUSCRIPTORES [ELABORACIÓN PROPIA]	78
FIGURA 3.28: VISTA REPORTES DE AGENDA [ELABORACIÓN PROPIA]	79
FIGURA 3.29: VISTA REPORTE DE ADMINISTRADORES [ELABORACIÓN PROPIA]	79
FIGURA 3.30: VISTA REPORTE DE LISTA DE ESPERA [ELABORACIÓN PROPIA]	80
FIGURA 3.31: VISTA AGREGAR ADMINISTRADOR [ELABORACIÓN PROPIA]	80
FIGURA 3.32: PORTADA MANUAL DE USUARIO [ELABORACIÓN PROPIA].....	81
FIGURA 3.33: CONEXIÓN A BASE DE DATOS [ELABORACIÓN PROPIA]	82
FIGURA 3.34: MODELO DE TABLA USUARIO [ELABORACIÓN PROPIA]	82
FIGURA 3.35: RELACIÓN DE LA TABLA AGENDA CON LA TABLA USUARIO [ELABORACIÓN PROPIA]	83
FIGURA 3.36: CONTROLADOR DE TABLA SUSCRIPTOR [ELABORACIÓN PROPIA].....	84
FIGURA 3.37: EJEMPLO DE INSERCIÓN DE DATOS A TABLA USUARIO [ELABORACIÓN PROPIA].....	86
FIGURA 3.38: DATOS YA INGRESADOS EN GESTOR DE BASE DE DATOS [ELABORACIÓN PROPIA].....	86
FIGURA 3.39: PORTADA FINAL DE DOCUMENTO MEMORIA DE TÍTULO [ELABORACIÓN PROPIA]	87

Capítulo 1

Introducción

Este capítulo presenta la problemática, los objetivos generales, los objetivos específicos, la motivación y las contribuciones de la memoria.

1.1. Introducción

Hoy en día la tecnología está cada vez más presente dentro de la vida de las personas, tanto a nivel personal como a nivel profesional. Esto se vio impulsado aún más debido a la pandemia presente actualmente en el mundo. Es por esto por lo que muchas personas intentan llevar su vida cotidiana de manera más normal posible, incorporando distintas tecnologías dentro de sus actividades profesionales, para así poder hacer frente a la situación previamente mencionada. Bajo esta premisa, muchas entidades de la salud se vieron en la obligación de trabajar de manera online al 100% y vía remota, siendo la más solicitada por las pacientes la medicina de salud mental, debido a los cambios drásticos en sus estilos de vida.

Los profesionales de la salud mental muchas veces trabajan de manera privada y autónoma, es por esto por lo que muchos no poseen vías digitales para contacto con sus pacientes, ni agenda o respaldo de fichas. Este es el caso de la psicóloga María José Caroca Araya, quien actualmente trabaja con un servicio gratuito que presta el sitio web www.doctoralia.cl, que es una plataforma para entidades de la salud en donde pueden presentar sus perfiles, medios de atención y los pacientes pueden solicitar horas de atención con el profesional deseado.

Es por esto por lo que se plantea el desarrollo de una plataforma web personalizada para la psicóloga, en donde pueda presentar la información completa que ella desee, además de que los pacientes puedan solicitar sus horas de atención.

1.2. Problemática

La psicóloga María José Caroca Araya actualmente cuenta con un servicio gratuito para darse a conocer de manera online. Dicho servicio le permite presentar sus métodos de atención, métodos de trabajo, administrar una agenda de atención médica, entre otras cosas. Sin embargo, se desea crear un sitio web personalizado, esto debido a que, si bien la plataforma actual le presta servicios gratuitos y puede trabajar relativamente bien con este, este puede presentar algunas desventajas o falencias, las cuales son:

- El aumento de profesionales del área, provocando que cada vez sea más difícil que los pacientes encuentren los perfiles deseados.

- Si bien el servicio es gratuito, existe la posibilidad de pagar para que el perfil del profesional aparezca siempre en los primeros resultados, provocando que los demás perfiles se vean desplazados hacia los últimos resultados de la búsqueda.
- Poca personalización de un perfil profesional, causando que no se pueda presentar toda la información deseada o que esta se presente de manera desordenada.
- La no existencia de una lista de espera en caso de que la agenda se encuentre llena puede provocar descontento y provoca que existan veces en que los profesionales pierdan horas de atención dentro del día en caso de que un paciente cancele su atención en último momento o que simplemente no pueda presentarse.

En conversaciones con la psicóloga, esta mencionó que si bien, dentro de la ciudad de Linares, en el servicio entregado por www.doctoralia.cl aparece siempre dentro de los primeros 3 resultados, el caso fue distinto cuando cambió de ciudad por motivos personales, provocando una baja de pacientes en una ciudad donde existen más profesionales de la misma área y que además pagan el servicio, desplazando aún más su perfil. Además, mencionó que muchas veces ella deseaba hacer sobrecupo con algún paciente que ella consideraba que era necesario, sin embargo, no era posible guardar registro de esa atención de manera inmediata dentro del sistema. Por último, mencionó que principalmente cuando trabajaba con niños, los padres de estos cancelaban su atención a pocos minutos antes de iniciarla, debido a que habían salido o por algún problema personal.

1.3. Objetivo General

Desarrollar una plataforma web que permita a los pacientes solicitar horas de atención, además de otros servicios que prestará la psicóloga.

1.4. Objetivos Específicos

Con el fin de satisfacer el objetivo general, se plantean los siguientes objetivos específicos:

- Especificar los requerimientos necesarios para el correcto funcionamiento de la solución.
- Utilizar el Marco de Trabajo Scrum.

- Integrar tecnologías Frontend y Backend para el desarrollo de la plataforma.
- Validación del sistema

1.5. Motivación

Los pacientes buscan una atención relacionada a la salud mental con algún profesional de confianza, debido a comentarios que pudieran haber escuchado o leído en algún sitio o por recomendación de cercanos que estén realizando una terapia en esos momentos. Fuera de eso, encontrar profesionales que generen la confianza necesaria para revelar sus problemas o secretos relacionados con su vida privada puede ser complicado si no se presentan de manera confiable y atractiva las metodologías y técnicas de trabajo para abarcar las distintas problemáticas a lo largo de las sesiones que lleve una terapia aplicada por un psicólogo.

Si bien cada vez es más común el preocuparse por la salud mental de las personas, muchas veces se escucha a gente decir que están bien o que no necesitan ningún tipo de atención o ayuda profesional, debido a que ellos mismos niegan estar frente algún caso de estrés, ansiedad, depresión, entre otros. Es por esto por lo que se busca presentar de la manera más concisa y de forma atractiva para que las personas puedan ver que quizás es momento de buscar ayuda profesional y que decidan dar el paso para comenzar a mejorar su salud mental.

1.6. Contribuciones

La principal contribución es entregar a la psicóloga una plataforma web para administrar su agenda de manera online, en donde podrá consultar los pacientes con horas tomadas, horas disponibles, entre otras funciones más pequeñas, como entregar noticias de interés a personas que se suscriban vía correo electrónico.

Por otra parte, los pacientes podrán consultar diversa información sobre la psicóloga, como métodos de atención, diagnósticos con los que trabaja, experiencia, entre otros datos útiles, buscando dar la confianza necesaria para tomar la decisión de agendar una hora de atención. Por otro lado, los mismos pacientes podrán registrar sus horas de atención de manera personal, para agilizar el proceso y ellos puedan ver directamente los horarios disponibles y seleccionar el que más les convenga.

SIAP tiene como fin el apoyar la administración de pacientes y agenda para la psicóloga, mientras que para los pacientes busca facilitar el solicitar horas con la psicóloga.

1.7. Alcance de la memoria

En este proyecto de memoria se abordó el desarrollo de una aplicación web para una psicóloga particular, el cual es capaz de administrar pacientes registrados y la agenda de atenciones médicas que se posee. Además, se implementaron distintas tecnologías, con la finalidad de generar un sistema lo más seguro posible con un marco ágil de desarrollo.

1.8. Organización del documento.

Esta memoria está organizada en cinco capítulos, los cuales se indican a continuación:

- **Capítulo 1:** Presenta la problemática, los objetivos generales y específicos, la motivación, las contribuciones esperadas y el alcance del proyecto de memoria.
- **Capítulo 2:** Presenta el marco teórico, donde se exponen distintos conceptos necesarios para comprender el problema y el desarrollo para la solución propuesta.
- **Capítulo 3:** Presenta el marco de trabajo y su adaptación necesaria para dar una correcta solución al problema planteado en el Capítulo 1.
- **Capítulo 4:** Expone la implementación de la metodología presentada en el Capítulo 3 mediante la cual se logra dar una solución a la problemática.
- **Capítulo 5:** Presenta las conclusiones obtenidas, trabajos futuros y consideraciones finales del proyecto de memoria.

Capítulo 2

Marco Teórico

Este capítulo explica los conceptos principales y tecnologías empleadas en el desarrollo de la plataforma web.

2.1. Tecnologías

2.1.1. Backend

En el desarrollo de *Backend*, la mayoría de los lenguajes actuales permiten desarrollar una API Rest de manera fácil, mediante distintas librerías o *framework*. Por esto mismo es muy importante definir la arquitectura o entorno donde se desarrollará el sistema.

2.1.1.1. Entorno

La elección del entorno con el cual desarrollar una plataforma web no es trivial y hay que tener algunos aspectos en consideración como lo es, el rendimiento, la concurrencia, la capacidad de respuesta, entre otros.

En el artículo “*A comparative study of PHP frameworks performance*” [PSF2019] se realiza una comparación de distintas tecnologías para el desarrollo web, entre ellas Laravel, Symfony y CodeIgniter, los cuales son *frameworks* populares actualmente para el desarrollo web. En resumen, este estudio reveló que, dentro de estos 3 *frameworks*, Laravel es el más conveniente para el desarrollo de aplicaciones web, esto puede ser visto claramente en la Figura 1.1 donde se puede observar las respuestas por segundo entre los 3 *frameworks*, la Figura 1.2, donde se muestra la memoria utilizada por estos y en la Figura 1.3, donde se puede apreciar el tiempo de espera para las respuestas solicitadas. Sin embargo, no todo es perfecto para Laravel, en la Figura 1.4 se puede apreciar que Laravel es el que posee la mayor cantidad de archivos, lo que indica que puede afectar la velocidad de carga de la infraestructura.

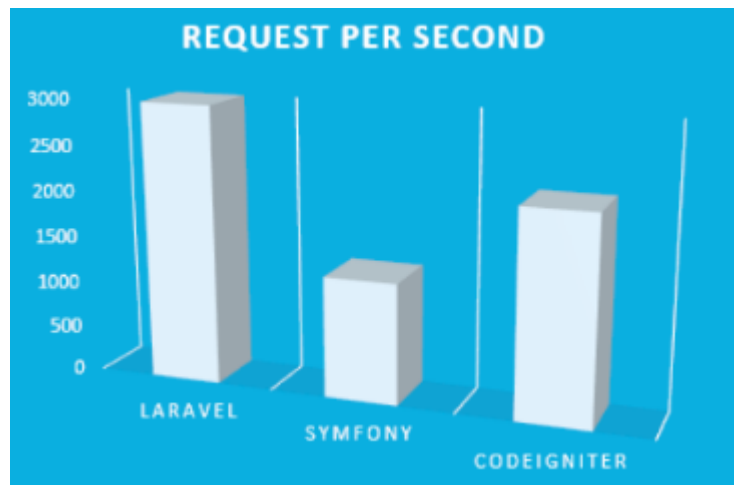


Figura 1.1: Resultados obtenidos en test de respuestas por segundo [PSF2019]

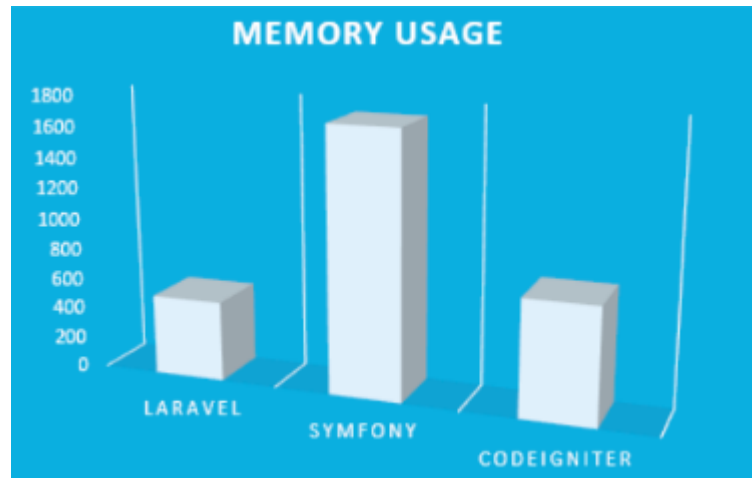


Figura 1.2: Resultados obtenidos en test de uso de memoria [PSF2019]

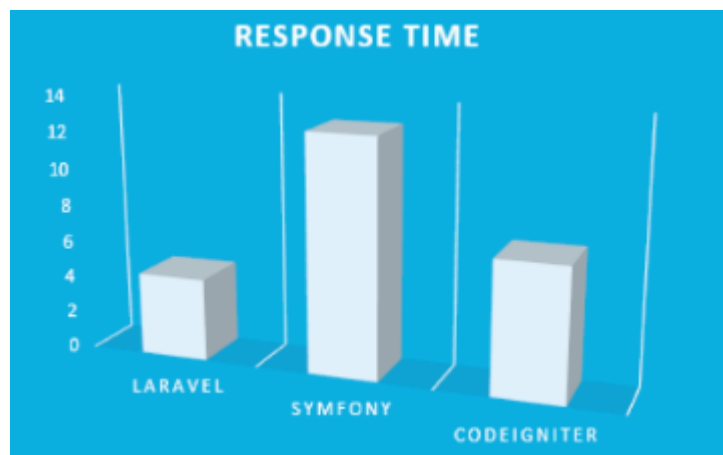


Figura 1.3: Resultados obtenidos en test de tiempo de respuesta [PSF2019]

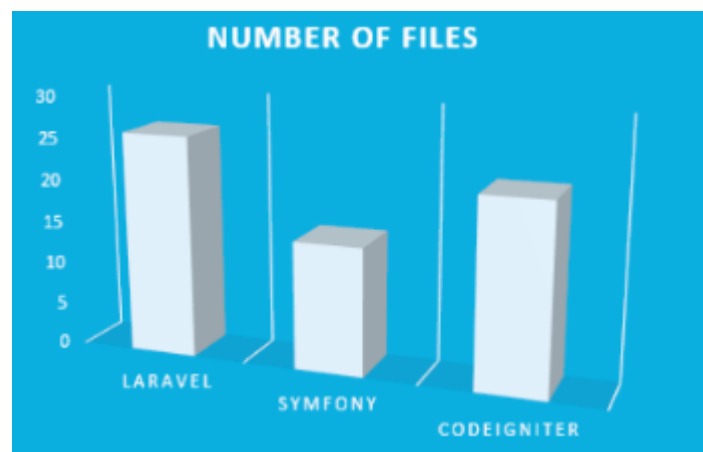


Figura 1.4: Resultados obtenidos al revisar creación de archivos [PSF2019]

2.1.1.2. Servidor

La selección de un servidor puede ser un poco complicada y subjetiva para su caso de uso. Esto se debe a que se elige basándose características particulares, como el peso del *framework*, velocidad, simplicidad, flexibilidad y configuración, o incluso popularidad. Gracias a la versatilidad de Laravel, su activa comunidad y a su nativa implementación de Vue.js (explicado más adelante), se ha definido que la mejor opción es trabajar con este poderoso *framework*.

“Laravel es un framework de aplicación web con una sintaxis expresiva y elegante. Creemos que el desarrollo debe ser una experiencia placentera y creativa para ser realmente satisfactorio. Laravel intenta eliminar el dolor del desarrollo al facilitar las tareas comunes utilizadas en la mayoría de los proyectos web.” [LAR2022]. Su API permite a los usuarios configurar rutas para enviar/recibir peticiones entre el *Frontend* y la base de datos (actuando como un marco de trabajo del servidor HTTP).

2.1.1.3. Modelo

El modelo es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tantas consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Esta información es almacenada comúnmente en una base de datos, la cual se define como: *“Un conjunto de datos persistentes a un mismo contexto y almacenados sistemáticamente para su posterior uso” [FIP2010].* Actualmente existen distintos tipos de bases de datos que permiten hacer consultas a la información almacenada:

- **Base de Datos no Relacionales.** Este tipo de base de datos son conocidas como No-SQL, son flexibles en cuanto a la consistencia de datos y se han convertido en una gran opción al momento de solucionar limitaciones que tiene el modelo relacional.
- **Base de Datos Relacional.** Estas bases de datos, como su nombre lo indica, utilizan un modelo relacional con la cual logran una gran consistencia al momento de almacenar la información. Es muy recomendable su uso cuando se requiere integridad y unicidad en los datos almacenados. Actualmente existe una gran variedad de base de datos relacionales, una de ellas es MariaDB.

MariaDB se autodefine como un “*MariaDB Server es uno de los servidores de bases de datos más populares del mundo. Está hecho por los desarrolladores originales de MySQL y se garantiza que seguirá siendo de código abierto. Los usuarios notables incluyen Wikipedia, WordPress.com y Google.*

MariaDB convierte los datos en información estructurada en una amplia gama de aplicaciones, desde banca hasta sitios web. Originalmente diseñado como un reemplazo directo mejorado para MySQL, MariaDB se usa porque es rápido, escalable y robusto, con un rico ecosistema de motores de almacenamiento, complementos y muchas otras herramientas que lo hacen muy versátil para una amplia variedad de casos de uso.” [MDB2022], Actualmente MariaDB es el gestor de bases de datos utilizados por XAMPP, el cual es un entorno de desarrollo para php gracias al servidor Apache.

2.1.1.4. ORM

El mapeo de objeto relacional es una técnica de programación que consiste en transformar las tablas de una base de datos en una serie de entidades que simplifiquen las tareas básicas de acceso a los datos almacenados en estas. Como es conocido, SQL se usa para acceder a muchas bases de datos, actualmente existen múltiples formas de realizar peticiones o consultas a los registros. Por ejemplo, la Figura 1.5 muestra las distintas formas de realizar una consulta en SqlServer, MariaDB y Oracle. Se debe mencionar

- **SqlServer:** *SELECT TOP 10 * FROM usuarios;*
- **MariaDB:** *SELECT * FROM usuarios LIMIT 10;*
- **Oracle:** *SELECT * FROM usuarios WHERE rownum<=20;*

Figura 1.5: Ejemplos de consultas SQL [Elaboración propia]

Debido a esto, un ORM genera una capa intermedia que abstrae el uso de la base de datos y centra el desarrollo en la aplicación. En SIAP, algunas ventajas del uso de un ORM son:

- Facilidad y velocidad de uso mediante operaciones básicas como el CRUD.
- Abstracción de la base de datos usada.
- Capacidad de migrar los datos a cualquier base de datos.

2.1.2. Frontend

“El Frontend son todas aquellas tecnologías que corren del lado del cliente, es decir, todas aquellas tecnologías que corren del lado del navegador web, generalizándose más que nada en tres lenguajes, HTML, CSS y JavaScript” [MPD2016]. Frontend se encarga de estilizar la página de tal manera que pueda quedar cómoda para la persona, es decir, utilizar técnicas de usabilidad para dar una mejor experiencia de usuario, así mismo un buen diseño de interacción, para colocar las cosas de tal manera que el usuario las pueda ubicar de manera rápida y cómoda. Existen tecnologías asociadas que integran el uso de estos tres lenguajes y que facilitan el desarrollo ágil de interfaces para el usuario, una de ellas es Vue.js.

2.1.2.1. Vue.js

Vue.js se define como *“un framework progresivo para construir interfaces de usuario.” [VJS2022]*. Sin embargo, lo cierto es que Vue.js es un excelente aliado para hacer todo tipo de aplicaciones web y SPA (*Single Page Application*).

Vue.js utiliza el gestor de paquetes NPM, el mismo que utiliza Node.js, otorgando acceso a muchas librerías y funciones que posee este poderoso entorno gestor.

Los componentes permiten separar la interfaz de usuario en piezas independientes, reutilizables y pensar en cada pieza de forma aislada. Los componentes pueden referirse a otros componentes en su salida. Esto permite utilizar la misma abstracción de componente para cualquier nivel de detalle. Un botón, un cuadro de diálogo, un formulario, una pantalla, etc.

2.1.3. API REST

La API Rest cambió por completo la ingeniería de software a partir del año 2000. En la actualidad no existe proyecto o aplicación que no disponga de una API Rest para la creación de servicios, como, por ejemplo, Facebook, Twitter y YouTube. Según Fielding *“un sistema Rest se define como cualquier interfaz entre sistemas que usen HTTP para obtener datos o generar operaciones sobre estos datos sin importar el formato” [APR2016]*. El desarrollo basado en API Rest otorga la facilidad de dividir el sistema en dos partes esenciales, como es el *Frontend* y *Backend* o comúnmente llamado cliente y servidor. Así podemos asegurar que el servicio de *Backend* pueda ser consumido por cualquier cliente o sistema, ya sea *framework*, librerías o

aplicaciones móviles desarrolladas en cualquier lenguaje, ya que la API otorga datos que están desacoplados y en un formato específico.

La Figura 1.6 muestra el funcionamiento de una API Rest, sus métodos asociados y los entes involucrados. A continuación, se describen los métodos para manipular los recursos HTTP.

- **GET:** Método para consultar y leer recursos.
- **POST:** Método para crear recursos.
- **PUT:** Método para editar recursos.
- **DELETE:** Método para eliminar recursos.

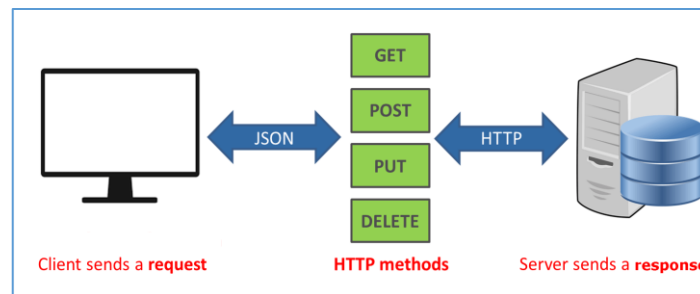


Figura 1.6: Diagrama de funcionamiento de API-REST [DIS2019]

Un punto importante dentro de la creación de una API es la documentación de esta, esto es fundamental para mantener y distribuir la información a otros desarrolladores. Tener una documentación bien estructurada facilita el proceso de aprendizaje sobre el nuevo entorno que se desarrollara. Sin embargo, documentar y mantener una API conlleva tiempo y recursos. Actualmente existen herramientas que facilitan esta tarea, algunas de las más conocidas son Raml [RAM2019], Slate [SLA2019] y Swagger [SWG2019]. Esta última es una herramienta poderosa para el diseño, creación, documentación y consumo de API, permite la interacción en tiempo real y la visualización mientras se construye. En este proyecto Swagger fue utilizado especialmente para la documentación y visualización de la API. La documentación completa de este proyecto se encuentra disponible en el anexo digital A.

2.1.4. Control de Versiones

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, además permite revertir archivos o un proyecto completo a un estado anterior, comparar cambios a lo largo del tiempo, ver quién modificó por última vez algo que puede estar causando un problema, quién introdujo un error y lo más importante cuándo lo hizo.

En la actualidad existen plataformas que soportan Git para ayudar a ser más fácil la creación de repositorios y el manejo de roles de un conjunto de desarrolladores, como lo es GitLab [GLB2019] y GitHub [GHB2019], además existen clientes gráficos, tales como SourceTree [STR2019], SmarGit [SMG2019] y GitCola [GCO2019]. Durante el todo el desarrollo se implementó GitHub como pilar central para mantener un control total sobre las versiones del sistema.

2.1.5. Marco de Trabajo Basado en Procesos Ágiles

El desarrollo ágil de software envuelve un enfoque para la toma de decisiones en los proyectos de software, que se refiere a métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. En la actualidad cualquier empresa dedicada al desarrollo de software que no opte por esta forma de trabajo difícilmente podrá sobrevivir en el mercado. Debido a esto existen muchas opciones para seguir el camino del desarrollo ágil entre ellas tenemos a *Scrum*, el cual es utilizado en el Capítulo 4 como marco de trabajo para la implementación de “SIAP”.

2.1.6. Jira

Jira es una “*herramienta ágil de gestión de proyectos compatible con cualquier metodología ágil, ya sea scrum, kanban o la tuya propia.*” [JRA2022]. En simples palabras, Jira es una pizarra blanca llena de notas adhesivas, donde cada nota es una tarea a realizar, en este caso llamadas incidencias”. Es por esto que Jira complementa la utilización de un marco de trabajo ágil otorgando un plus al momento de crear los requerimientos o en el caso de Scrum la lista del producto. En el Capítulo 4 se muestra el uso de esta herramienta y los resultados obtenidos. El acceso a esta herramienta se encuentra disponible en el anexo B.

En SIAP, con la integración de estas distintas tecnologías y en ayuda de un marco de trabajo ágil se genera una propuesta de solución para satisfacer la problemática inicial. En los siguientes capítulos se muestra la unión de estas herramientas y el desarrollo de la plataforma web mediante el marco de trabajo seleccionado.

Capítulo 3

Marco de Trabajo y Propuesta de Solución

Este capítulo presenta una descripción del marco de trabajo *Scrum*, una pequeña adaptación y la propuesta de solución a la problemática.

3.1. Marco de Trabajo Scrum

Scrum es “un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto” [LGS2017]. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. A continuación, se presta las características más relevantes de esta metodología. En la Figura 2.1 se presentan los elementos importantes de Scrum: Product Backlog, Sprint Planning, Sprint Backlog, Scrum Team, Daily Scrum, Sprint Retrospective, Sprint Review e Increment. La presentación de Scrum empleó los conceptos en inglés para evitar pedida de semántica.

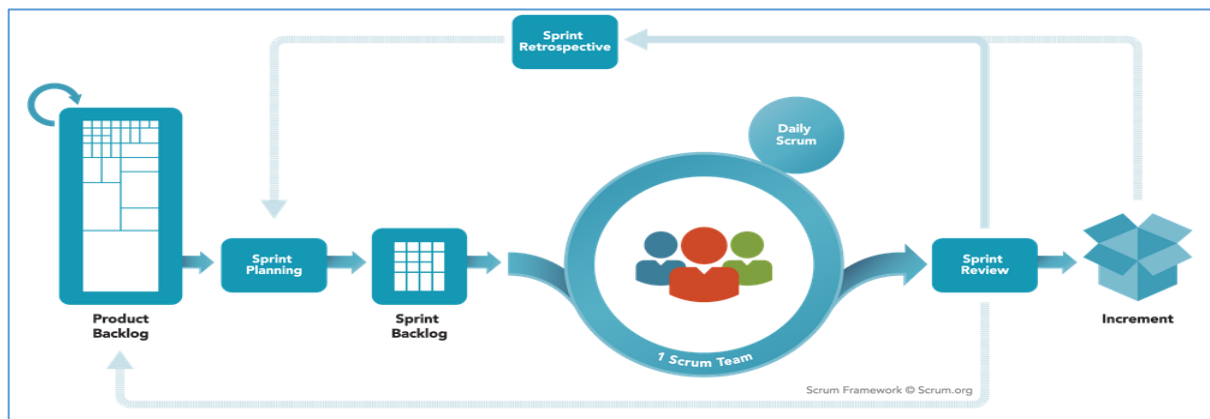


Figura 2.1: Flujo de Metodología Scrum [Arredondo, E. (2016). Scrum]

3.1.1. Product Backlog

El *Product Backlog* (Lista del Producto) es una lista ordenada de todo lo que se conoce que es necesario en el producto. Es la única fuente de historias de usuarios para cualquier cambio a realizarse en el producto. El *Product Owner* (Dueño de Producto) es el responsable del *Product Backlog*, incluyendo su contenido, disponibilidad y ordenación.

3.1.2. Sprint Planning

El *Sprint Planning* (Planificación de la iteración) es donde se estructura el plan de trabajo del *Sprint*, el cual es un período de tiempo determinado. Este plan se crea mediante el trabajo colaborativo del *Team Scrum* (Equipo de Scrum) completo.

3.1.3. Sprint Backlog

El *Sprint Backlog* (Lista de Pendientes) es el conjunto de elementos del *Product Backlog* seleccionados para el *Sprint*, más un plan para entregar el *Increment* (Incremento) de producto y conseguir los objetivos del *Sprint*.

3.1.4. Team

El *Product Owner* (Dueño del Producto) es el responsable de maximizar el valor del producto resultante del trabajo del *Team*. Además, es el único encargado de gestionar el *Product Backlog*.

Por su parte, el *Scrum Master* (Facilitador del Proyecto) es el encargado de guiar al *Team* a ser auto organizado y multifuncional, los ayuda a crear productos de alto valor y elimina impedimentos para el progreso.

Finalmente, el *Team* (Equipo de trabajo) consiste en profesionales que realizan el trabajo de entregar un incremento del producto.

3.1.5. Daily Scrum

Daily Scrum (Scrum Diario) es una reunión con un bloque de tiempo de 15 minutos para el *Team*. Este se lleva a cabo cada día del *sprint*. En él, el *Team* planea el trabajo para las siguientes 24 horas. Esto optimiza la colaboración y el desempeño del equipo inspeccionando el trabajo avanzado desde el último *Daily Scrum* y haciendo una proyección del trabajo del *Sprint* a realizar a continuación. El *Daily Scrum* se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad. En particular, en el proyecto las reuniones fueron semanales (*Weekly meeting*).

3.1.6. Sprint Review

El *Sprint Review* (Revisión de iteración) es una revisión para inspeccionar el *Increment* y adaptar el *Product Backlog* si fuese necesario. Durante el *Sprint Review*, el *Team* colabora acerca de lo que se hizo durante el *Sprint*. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del *Increment* tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración.

3.1.7. Sprint Retrospective

El *Sprint Retrospective* (Retrospectiva de iteración) es una oportunidad para el *Team* de inspeccionarse a sí mismo y de crear un plan de mejoras que sean abordadas durante el siguiente *Sprint*. El *Sprint Retrospective* tiene lugar después de la *Sprint Review* y antes del siguiente *Sprint Planning*.

El propósito del *Sprint Retrospective* según la guía de *Scrum* [LGS2017] es:

- Inspeccionar cómo fue el último *Sprint* en cuanto a personas, relaciones, procesos y herramientas
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras
- Crear un plan para implementar las mejoras a la forma en la que el *Team* desempeña su trabajo

3.1.8. Increment

El *Increment* (Incremento del producto) es la suma de todos los elementos del *Product Backlog* completados durante un *Sprint* y el valor de los incrementos de todos los *Sprints* anteriores. Al final de un *Sprint* el nuevo *Increment* debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la definición de “Terminado” del *Team*.

3.2. Adaptación del Marco de Trabajo Scrum

Fueron realizadas adaptaciones a Scrum para acomodarse al tiempo y espacio del Team compuesto por el Profesor Marco Toranzo y los Memorias Luis Caroca y David González.

3.2.1. Adaptación de Product Backlog y Sprint Backlog

Como se mencionó anteriormente, el *Product Backlog* y el *Sprint Backlog* son listas ordenadas relacionadas con el producto final. Es por esto que para agilizar la comunicación y la accesibilidad a esta lista se utilizara la herramienta Jira, quien permite generar el *Product Backlog* y *Sprint Backlog* en una pizarra virtual disponible para todo el *Team*. Por su parte, los *Sprint* se espera que sean máximo 9 en total y tienen una duración de 2 semanas, en los cuales

se pretende que al finalizar cada uno se entregue un incremento del producto. La Figura 2.2 muestra la adaptación del *Sprint Backlog* con la utilización de la herramienta *Jira*.

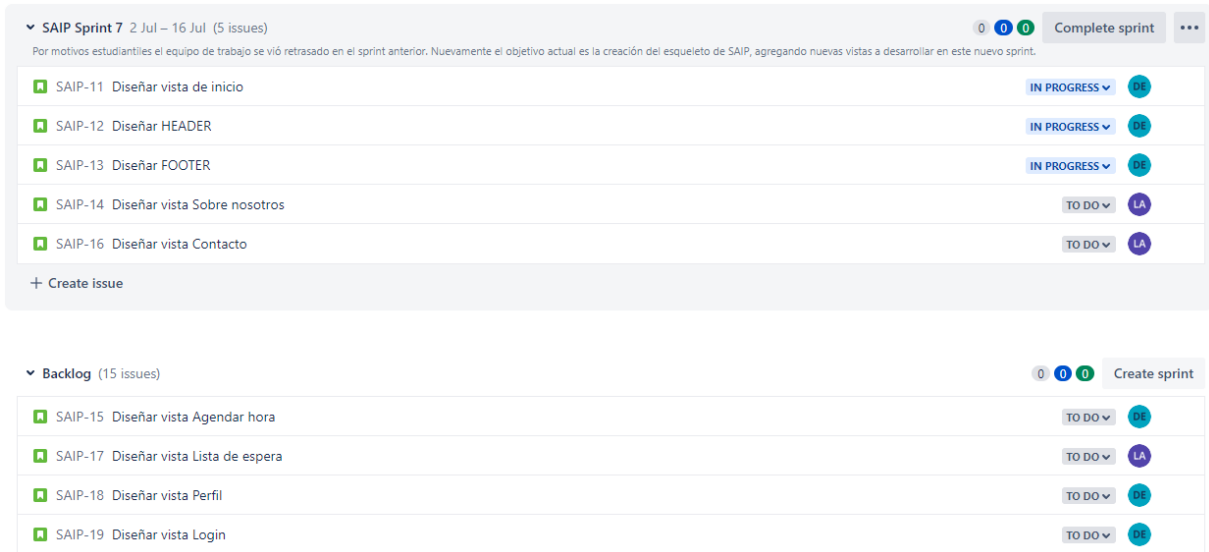


Figura 2.2: Backlog de Sprint 7 y backlog de SIAP [Elaboración propia]

3.2.2. Adaptacion Scrum Team

El *Team* está compuesto tres integrantes quienes cumplen los siguientes roles.

- **Scrum Master y Product Owner:** Profesor Marco Toranzo, quien es el responsable de maximizar el valor del producto, guiar el desarrollo y ayudar al *Team*.
- **Team:** Alumno Luis Caroca Araya y alumno David González Elgueta, quienes son encargados de investigar, desarrollar, probar y realizar las entregas de los incrementos por cada *sprint*.

3.2.3. Adaptacion de Daily Scrum y Sprint Review

Los *Daily Scrum* entre el *Team* y el *Scrum Master* no son posibles dentro de este desarrollo debido al escaso tiempo que se posee por parte de ambos miembros, es por esto que es modificado a reuniones semanales donde se analiza el avance del *Product Backlog* y el *Increment*. Este *Weekly Meeting* se realiza todos los sábados a la misma hora y tiene una duración de 1 a 2 horas máximo en la sala de clases, seguido de una reunión vía online donde solamente se reunía el *Team*, en donde además se genera un nuevo *Sprint Backlog*.

3.3. Propuesta de la Solución

Con los análisis realizados en las secciones anteriores se presenta a continuación un modelo y una arquitectura completa para dar solución a la problemática. Finalmente, se presentarán algunas vistas para ejemplificar algunas funcionalidades dentro de SIAP.

3.3.1. Modelo de Datos

Es imprescindible el diseño de un modelo de datos para esclarecer la idea al momento de realizar el desarrollo, entender el funcionamiento de las entidades, sus atributos, interacciones, los stakeholders del sistema y mantener siempre el foco en la propuesta. A continuación, la Figura 2.3 muestra el modelo relacional realizado en la plataforma Lucidchart.com [LCH2022].

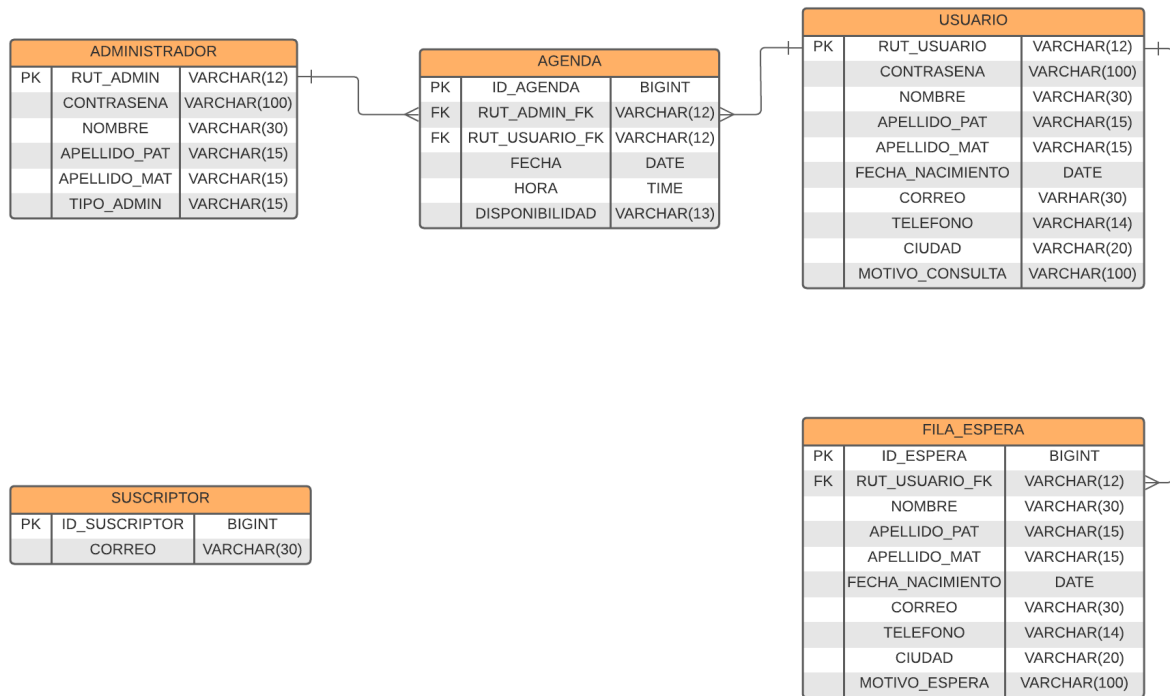


Figura 2.3: Modelo relacional a utilizar por SIAP [Elaboración propia]

3.3.2. Arquitectura

Por otra parte, en el desarrollo también es importante tener claro con que herramientas se debe trabajar, como se relacionan entre sí y que funcionalidad cumplirá cada una de ellas. A continuación, en la Figura 2.4 se presenta la arquitectura cliente-servidor de SIAP.

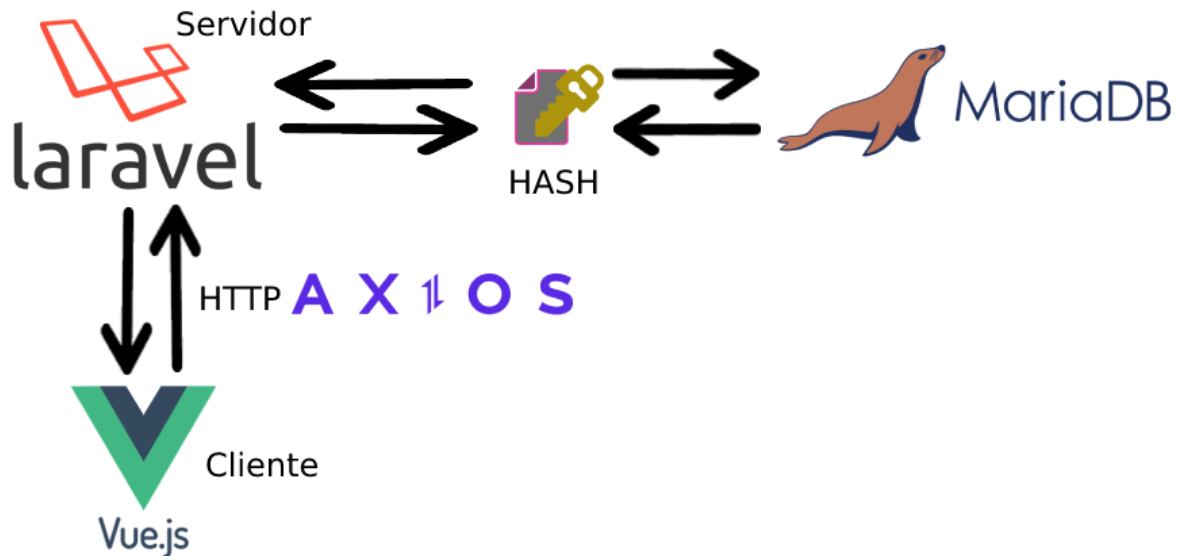


Figura 2.4: Arquitectura utilizada por SIAP [Elaboración propia]

A continuación, es presentada una explicación de los conceptos de la figura anterior:

- **Servidor:** Debido al rendimiento definido en el capítulo anterior, se define Laravel como entorno ideal para el desarrollo de esta aplicación en conjunto con Vue.Js, que es implementado de manera nativo, para el servidor.
- **Base de Datos:** Debido a que Laravel no requiere aplicaciones ni funciones externas, se decide utilizar netamente su funcionamiento para trabajar con el gestor de base de datos MariaDB, el cual nos permitirá recibir y realizar migraciones, además de almacenar los datos necesarios requeridos.
- **Seguridad HASH:** Por otra parte, la comunicación entre el servidor *Backend* y el *Frontend* o cliente será protegida mediante el sistema de seguridad que proporciona Laravel, el cual se aplica mediante la función Hash la cual almacena las contraseñas de manera segura y de manera compleja de descencriptar.
- **Frontend:** Vue.Js está a la vanguardia en el desarrollo de *Frontend*, permite realizar sistemas de una sola página, facilitando el desarrollo mediante componentes.

- **HTTP:** Axios es un cliente HTTP que es utilizado por Node.js, por esto mismo, es utilizado por Vue.js para trabajar la transferencia de información en los sistemas cliente-servidor.

En SIAP, modificar la metodología de trabajo es esencial para obtener resultados concretos, debido a los problemas y necesidades del equipo de trabajo, además con un modelo de datos y una arquitectura definida no existe ambigüedad al momento de desarrollar y se mantiene la vista en los objetivos establecidos anteriormente.

3.3.3. Muestra del sistema

Dentro de la solución propuesta en SIAP, se presenta el trabajo con distintas vistas para la muestra de información, formularios de registro de datos y las distintas funcionalidades dentro del sistema. Se debe destacar que existen 2 roles diferentes dentro de SIAP, el rol *Administrador*, encargado de la administración del sistema y los registros de la base de datos, y el rol *Usuario*, el cual podrá realizar distintas acciones dentro del sistema, como el agendar horas, agregarse a la lista de espera, etc.

Como se puede apreciar en la figura 2.5, en esta vista las personas pueden hacer registro de una cuenta de *Usuario* dentro de SIAP, la cual les permitirá acceso a las funciones de agenda. Para esto, el usuario deberá ingresar sus datos personales y de contacto e ingresar los datos del motivo de su consulta, para que así la profesional pueda acceder a dichos datos y saber de antemano el porque de la necesidad de la atención psicológica.

Figura 2.5: Vista Registro de usuario [Elaboración propia]

Como se puede ver en la figura 2.6, existe una vista de reporte de usuarios, la cual presenta una tabla con los usuarios registrados en SIAP en manera de lista para que la profesional tenga fácil acceso a los datos que ella requiera. Además de un control para la edición de datos, en caso de reconocer datos erróneos, la eliminación de datos, en caso de que un paciente desee dar de bajas sus atenciones por completo, y la opción de registrar nuevos usuarios a través del mismo formulario presentado en la figura 2.5. Esto con la intención de facilitar el registro de usuarios al administrador en caso de que un paciente solicite una hora a vía llamada telefónica, correo electrónico, servicios de mensajería como WhatsApp u otros.



Rut usuario	Nombre	Apellido paterno	Apellido materno	Fecha nacimiento	Correo electrónico	Teléfono	Ciudad	Acciones
18.779.181-2	Luis Joaquín	Caroca	Araya	02-Apr-1994	luis.caracathuenalec@gmail.com	+96997112049	Linares	 

Figura 2.6: Vista Reporte de usuarios [Elaboración propia]

Estos son solo unos ejemplos de las vistas que se pueden encontrar en SIAP, sin embargo, se puede acceder al detalle completo de las vistas de estas a través del documento [Manual de Usuario](#), el cual contiene una explicación detallada de cada vista y sus funciones. Además de presentar las funciones tanto de los usuarios, como de los administradores dentro de SIAP.

Capítulo 4

Implementación del Sistema

A continuación, se presenta un desglose por semana del desarrollo de la herramienta ‘SIAP’ mediante el uso del marco de trabajo *Scrum*, siguiendo la estructura y arquitectura definida en el capítulo anterior.

4.1. Product Backlog

Al inicio de este desarrollo, se identifican las historias de usuario del producto basado en lo pedido por el scrum master, los cuales son separados y escritos en dos listas de producto, la *Product Backlog API* y *Product Backlog Cliente*. La Figura 3.1 presenta una captura de la herramienta Jira en la que se aprecian algunas de las *issues* creadas (aún sin ser añadidas a un sprint ni sin ser asignadas a algún miembro del *Team*, que deben ser trabajadas dentro del periodo de desarrollo de SIAP:



Figura 3.1: Backlog para el desarrollo de SIAP [Elaboración propia]

A continuación, la Figura 3.2 se presenta el *board* de un sprint, en donde se pueden apreciar los 3 estados que puede tener un *issue*:

- TO DO: Son los *issues* del sprint que aún están pendientes a trabajar.
- IN PROGRESS: Son los *issues* del sprint que están actualmente en desarrollo.
- DONE: Son los *issues* del sprint que ya fueron finalizados y revisados.

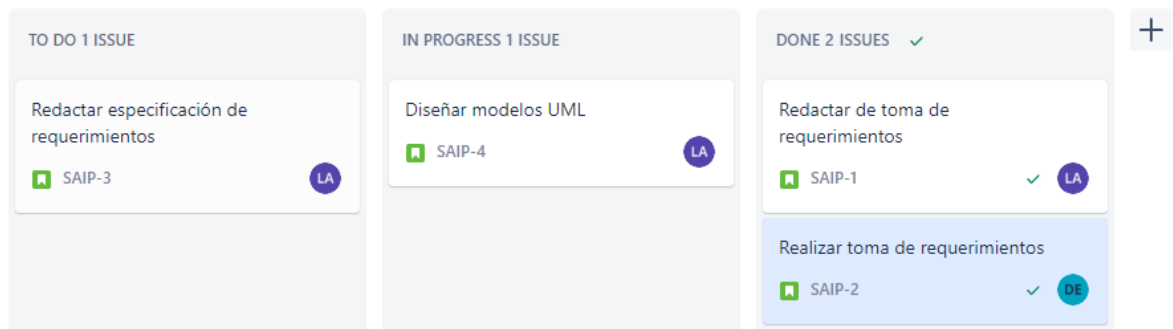


Figura 3.2: Board de un sprint en Jira [Elaboración propia]

4.1. Sprint N° 1

Este primer sprint se centra en las acciones previas al desarrollo de SIAP, centrándose netamente en lo que es la toma de requerimientos. Tiene una duración de 2 semanas en las cuales se espera que las diferentes tareas sean realizadas, revisadas y aprobadas en las *Weekly Meeting*.

4.1.1. Sprint N° 1

4.1.1.1. 1° Weekly Meeting

En la primera reunión, se define el *Sprint Backlog* para la actual semana, se presentan algunas recomendaciones a la arquitectura y tecnología de trabajo

4.1.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°1.

- **Redactar toma de requerimientos:** Se redacta la toma de requerimientos, debido a conversaciones previas con el cliente, se tenía conocimiento previo de lo necesario, entonces esta pudo ser escrita de manera que guiara claramente al cliente a las respuestas más concretas posibles.
- **Realizar toma de requerimientos:** Se realiza la toma de requerimientos, en este caso, fue realizada a través de encuestas de Google.
- **Redactar especificación de requerimientos:** Se redacta el documento de donde se especifican los requerimientos identificados, se deja claro el equipo de trabajo, el cómo funcionará el sistema, etc.
- **Diseñar modelos UML:** Se diseñan los diagramas UML necesarios para incluirlos en el documento de especificación de requerimientos.

4.1.1.3. 2° Weekly Meeting

En la segunda reunión semanal del *Sprint* N° 1, se revisa el estado del avance del *Sprint Backlog* y se dan por aprobadas las *issues* ya finalizadas.

4.1.2. Sprint Review

En esta reunión se revisa de manera completa el *Sprint* N°1 para generar el primer incremento de la aplicación. Se realizó una última reunión con el cliente, con la intención de que este tuviera conocimiento de los requerimientos extraídos en el proceso de toma de

requerimientos, por si presentaba disconformidad con alguno de estos o si deseaba agregar uno nuevo antes de dar por finalizado el proceso de requerimientos. Además, se realizaron consultas tanto al cliente como al *Scrum Master* sobre el avance del proyecto y ambas partes indicaron estar conformes, dando por cerrado y por fin finalizado el *Sprint* N°1.

4.1.3. Sprint Retrospective

Uno de los puntos más importantes, sin embargo, muchas veces olvidado, es la toma de requerimientos. Una mala toma de requerimientos puede llevar al desarrollo de una aplicación o sistema que difiera de las necesidades del cliente. Es por esto por lo que se debe realizar de la manera más clara y concisa posible, teniendo en cuenta el cliente con el que se está tratando y las necesidades que este presenta. Gracias a estos requerimientos, el equipo de desarrollo podrá trabajar en los siguientes *Sprints* sin problemas.

4.1.4. Increment

Como consecuencia del primer *Sprint* de este trabajo, el cual era centrado en la toma de requerimientos, se obtuvieron los siguientes resultados presentados en las subsecciones.

4.1.4.1. Documento de especificación de requerimientos.

Este documento almacena la información base para el desarrollo de SIAP, conteniendo los requerimientos especificados por el equipo de desarrollo, gracias a los datos que se pudieron extraer de la toma de requerimientos que tuvo lugar previamente con el cliente.

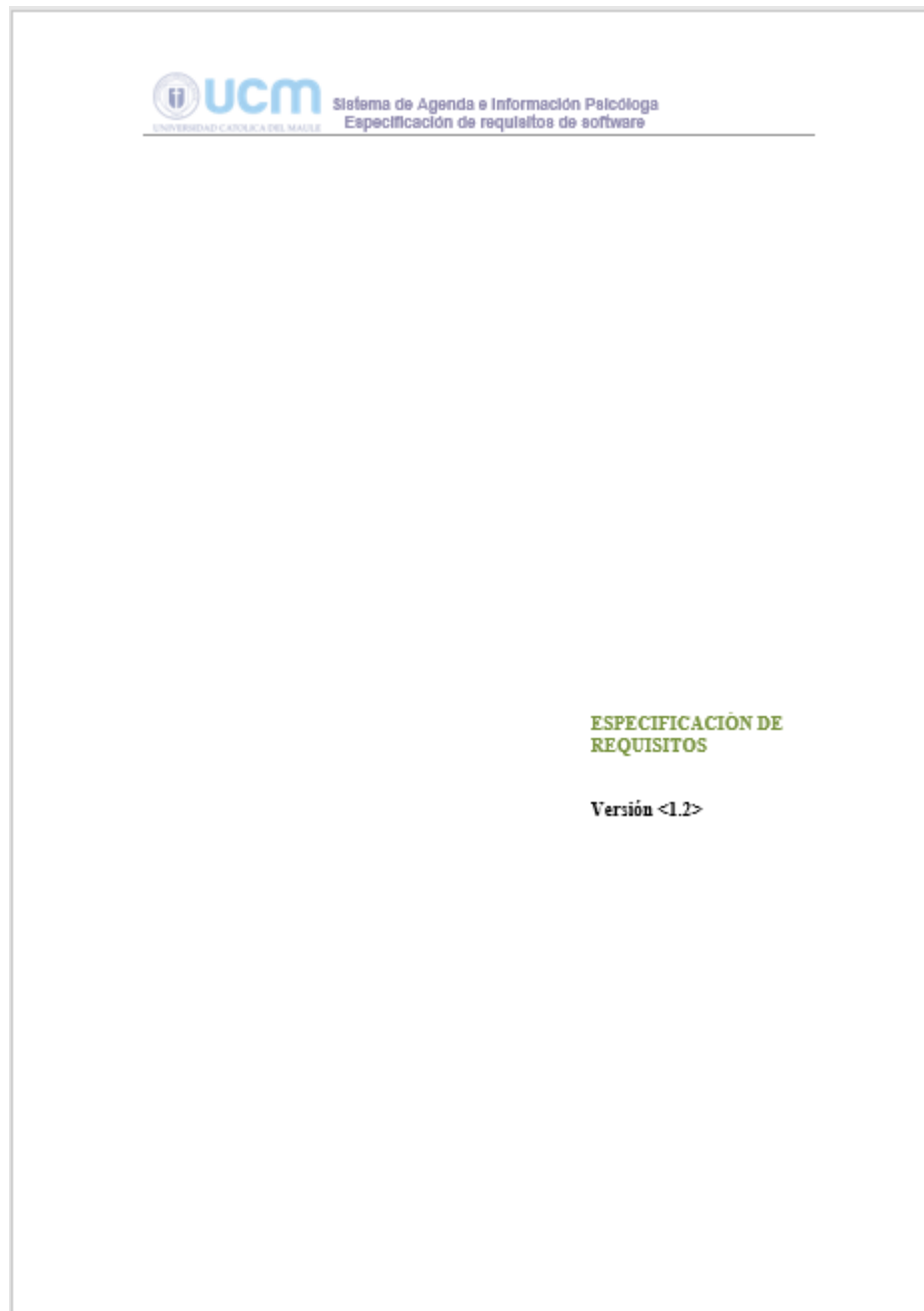


Figura 3.3: Portada de documento [Elaboración propia]

4.2. Sprint N°2

El segundo *Sprint* se centra en la creación del diseño de la base de datos a utilizar por SIAP. Tiene una duración de 2 semanas en las cuales se espera que las diferentes tareas sean realizadas, revisadas y aprobadas en las *Weekly Meeting*.

4.2.1. Sprint N° 2

4.2.1.1. 1° Weekly Meeting

En esta primera reunión se analizan los datos que deben ser almacenados, generando una lluvia de ideas por el *Team* de cuál sería la manera correcta de crear este modelo para que sea entendible y sea modelada en, al menos, tercera forma normal (3FN).

4.2.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°2.

- **Diseñar base de datos:** Se diseña el modelo la base de datos, procurando que este esté modelado como mínimo en tercera forma normal.

4.2.1.3. 2° Weekly Meeting

Se consulta al *Scrum Master* por consejos para el desarrollo del modelado de la base de datos, donde se busca esclarecer ciertas dudas relacionadas a la sección de noticias de la base de datos.

4.2.2. Sprint Review

En esta reunión se conversó respecto al modelo de la base de datos, en donde el *Team* llegó a la conclusión de que se había conseguido llegar al modelo ideal para SIAP.

4.2.3. Sprint Retrospective

A diferencia del *Sprint* anterior, en esta ocasión se debió solicitar mayor ayuda al *Scrum Master* para que ayudará al *Team* con el desarrollo del modelo de la base de datos. Sin embargo, se consiguió llegar al modelo que logra satisfacer todas las necesidades del cliente, ser ordenada en sus datos y estar en 3NF.

4.2.4. Increment

Como resultado de este segundo *Sprint* se obtienen el siguiente modelo, presentado en la subsección.

4.2.4.1. Modelo relacional de base de datos.

Este modelo relacional representa de manera visual los datos que se trabajarán en SIAP, al igual que sus relaciones entre las distintas tablas de la base de datos.

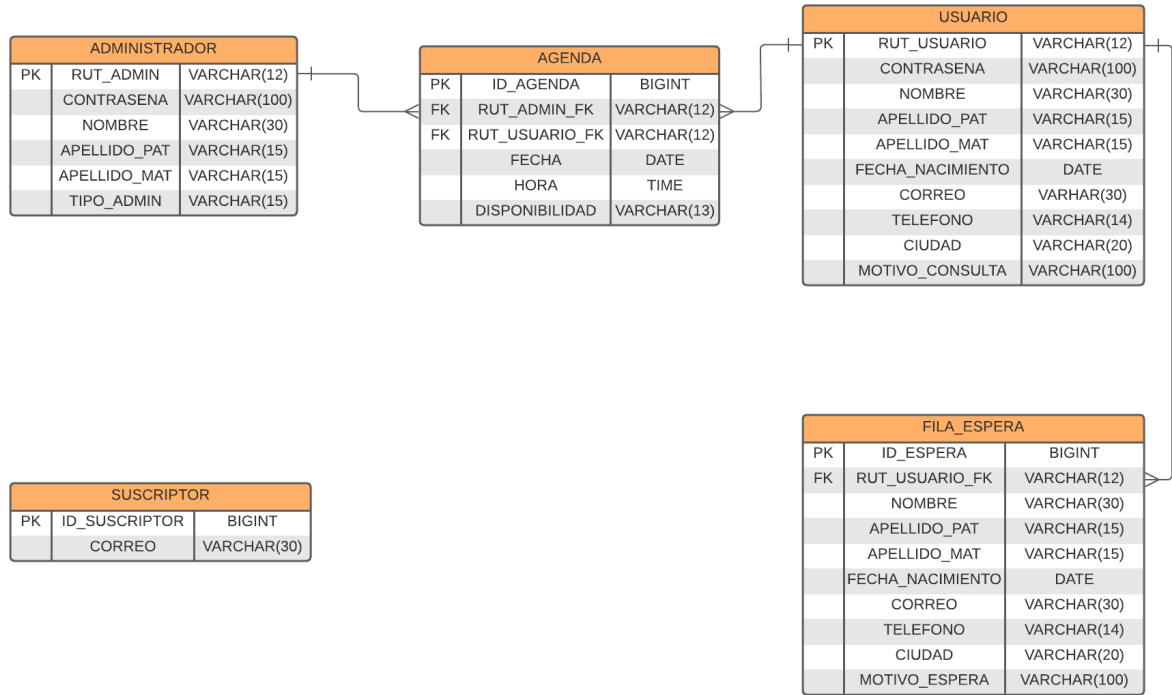


Figura 3.4: Modelo relacional de la base de datos de SIAP [Elaboración propia]

4.3. Sprint N°3

El tercer *Sprint* se centra en la investigación necesaria para implementar las tecnologías necesarias en SIAP. Tiene una duración de 2 semanas en las cuales se investigará la compatibilidad de las tecnologías y las distintas aplicaciones que pueden tener.

4.3.1. Sprint N° 3

4.3.1.1. 1° Weekly Meeting

En esta primera reunión se decide exactamente con que tecnologías se llevará a cabo el desarrollo de SIAP y, posteriormente, se le informa al *Scrum Master* cuales fueron las tecnologías seleccionadas.

4.3.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°3.

- **Investigar sobre Laravel:** El equipo de desarrollo debe realizar investigación sobre esta tecnología, la cual fue la seleccionada para trabajar el *backend* de SIAP.

- **Investigar sobre Vue.js:** El equipo de desarrollo debe realizar investigación sobre esta tecnología, la cual fue la seleccionada para trabajar el *frontend* de SIAP.
- **Investigar sobre SPA (Single-Page Application):** El equipo de desarrollo debe realizar investigación sobre las SPA, el cual es un método de trabajo de aplicaciones web muy popular en la actualidad.

4.3.1.3. 2° Weekly Meeting

El *Team* presenta la información de cada tecnología al resto del equipo, logrando un aprendizaje mayor de cada una de estas.

4.3.2. Sprint Review

En esta reunión se analizaron los problemas que se presentaron debido al periodo de evaluaciones de la institución, en donde el *Sprint* no se vio concretado en su totalidad, dejando *issues* pendientes para un futuro *Sprint*.

4.3.3. Sprint Retrospective

Debido a motivos externos al *Team*, este *Sprint* no pudo ser desarrollado en su totalidad. Es importante llevar un orden de las actividades para no retrasar el proceso completo del desarrollo de un trabajo. Sin embargo, esto permitirá al *Team* documentarse de mejor manera con respecto a las tecnologías deseadas a utilizar.

4.3.4. Increment

A diferencia de los *Sprints* anteriores, este, al ser netamente de investigación y documentación por parte del *Team*, no generó ningún resultado visible, escrito o presentable, sin embargo, aumentó los conocimientos del equipo.

4.4. Sprint N°4

El cuarto *Sprint* se centra solamente en la investigación de SPA, debido a que fue lo que quedo pendiente del *Sprint* anterior. Tiene una duración de 2 semanas en las cuales se investigará a profundidad el funcionamiento de una SPA y como aplicarla a SIAP.

4.4.1. Sprint N° 4

4.4.1.1. 1° Weekly Meeting

En esta primera reunión el *Team* presenta qué es lo que entiende cada uno por una SPA, discutiendo diferentes conceptos, sin embargo, no dejando en claro exactamente que son, debido a la falta de documentación actual.

4.4.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°4.

- **Investigar sobre SPA (Single-Page Application):** El equipo de desarrollo debe realizar investigación sobre las SPA, el cual es un método de trabajo de aplicaciones web muy popular en la actualidad.

4.4.1.3. 2° Weekly Meeting

En esta reunión el *Team* vuelve a discutir lo que han aprendido sobre SPA, discutiendo como puede ser aplicada a SIAP.

4.4.2. Sprint Review

En esta reunión se definió definitivamente entre el *Team* como se implementaría la SPA dentro de SIAP, dando espacio a posibles alteraciones.

4.4.3. Sprint Retrospective

El *Team* durante el desarrollo de este *Sprint* decidió solamente dejar 1 tarea, con la intención de dar cierre al *Sprint* anterior, sin embargo, no se siente que el trabajo se pudo haber visto retrasado al no agregar más *issues* al *Sprint*. Esto será algo a tener en cuenta para *Sprints* posteriores.

4.4.4. Increment

Al igual que con el *Sprint* anterior, este al ser solamente de investigación no generó ningún resultado visible, escrito o presentable, sin embargo, aumentó los conocimientos del equipo.

4.5. Sprint N°5

El quinto *Sprint* se centra en el diseño visual base que tendrá SIAP. Tiene una duración de 2 semanas en las cuales se espera realizar una primera visual a SIAP, para tener mayor claridad al momento de desarrollar.

4.5.1. Sprint N° 5

4.5.1.1. 1° Weekly Meeting

En esta primera reunión el *Team* decide el software a utilizar para el diseño base de las vistas y el cómo se verán estas.

4.5.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°5.

- **Diseñar mockup de vistas:** Antes de lanzarse al diseño de las vistas en la aplicación, el *Team* decide que es ideal realizar un *mockup*, con la intención de tener una referencia visual al momento de trabajar la aplicación.
- **Diseñar formularios:** Al momento de realizar los *mockups* de algunas vistas, se aprovecha de realizar un diseño previo de los formularios que se utilizaran dentro de SIAP, con la intención de que no falte ningún campo al momento de realizar el sistema.

4.5.1.3. 2° Weekly Meeting

Se revisan los *mockups* diseñados, dando visto bueno a los ya realizados y con los que el *Team* se siente satisfecho con ellos y decidiendo los que deben ser nuevamente diseñados.

4.5.2. Sprint Review

En esta reunión el *Team* decide comunicarse con el cliente, para presentarle los *mockups* realizados, para que pueda apreciar si se cumple con lo que se desea visualmente o si se desea realizar algún cambio.

4.5.3. Sprint Retrospective

Al momento de realizar un software para un cliente en concreto, se debe tener muy en cuenta sus solicitudes y los requerimientos que este presenta, con el fin de que el *Team* pueda realizar concretamente lo que se desea, sin embargo, se debe destacar que muchas veces se debe

guiar al cliente para obtener los resultados adecuados, debido a que muchas veces lo que expresa no es claro o ni siquiera él tiene claridad de lo que quiere o necesita.

4.5.4. Increment

Como resultado de este quinto *Sprint* se obtienen variados *mockups*, con la intención de solamente graficar este resultado, se presentará solo el siguiente *mockup*, el cual contiene un formulario, presentado en la subsección.

4.5.4.1. Mockup de registro de pacientes.

En el siguiente *mockup* se puede apreciar la distribución que tendrá la aplicación, tanto en su HEADER como FOOTER, así como también una idea de cómo se debería ver y los datos que debe presentar el formulario de registro de pacientes.

Logo

Inicio Sobre mí Agenda tu hora Usuario

Registro de pacientes

Rut: Nombre:

Apellido paterno: Apellido materno:

Fecha de nacimiento: Correo:

Teléfono: Ciudad:

Motivo de consulta:

Registrar

Preguntas frecuentes Redes sociales

Figura 3.5: Mockup de un formulario en SIAP [Elaboración propia]

4.6. Sprint N°6

El sexto *Sprint* se centra en la creación del esqueleto de SIAP. Tiene una duración de 2 semanas en las cuales se espera dar inicio al desarrollo visual de SIAP.

4.6.1. Sprint N° 2

4.6.1.1. 1° Weekly Meeting

En esta primera reunión se define que debe poseer tanto el HEADER como el FOOTER de SIAP, además de los colores finales de este, debido a que los presentados en el *mockup* pueden resultar molestos para la vista, sin embargo, esto se debe a la poca personalización de colores dentro del software utilizado.

4.6.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°6.

- **Diseñar vista de inicio:** Esta vista será la que cualquier persona verá al momento de abrir SIAP, ya sea el cliente u otra persona.
- **Diseñar HEADER:** El HEADER es la sección de arriba de cada sitio web, en donde podemos encontrar, por lo general, la navegación del sitio.
- **Diseñar FOOTER:** El FOOTER es la sección de más abajo de un sitio web, en donde por lo general se encuentran preguntas frecuentes de los visitantes, datos de interés o medios de contacto, como redes sociales o correos electrónicos.

4.6.1.3. 2° Weekly Meeting

El *Team*, debido a evaluaciones, decide disminuir el ritmo de trabajo dentro de SIAP, con el fin de poder rendir de mejor manera en todos los ramos que se están cursando.

4.6.2. Sprint Review

En esta reunión el *Team* definió como sería el ritmo de trabajo dentro de los siguientes *Sprints*, deseando apresurar el desarrollo de SIAP.

4.6.3. Sprint Retrospective

Nuevamente el *Team* se ve un tanto retrasado debido a las obligaciones dentro de la institución en la que están cursando su carrera, sin embargo, comprenden que esto se debe a

falta de organización dentro del mismo *Team* para enfrentar estos casos. A futuro se espera no tener que volver a retrasar el desarrollo ni que alguna tarea quede pendiente dentro de un *Sprint*, provocando que pase al siguiente, acumulando más tareas dentro del mismo periodo de tiempo.

4.6.4. Increment

Debido a que no se concretó al 100% ninguna de las tareas deseadas en el *Sprint Backlog*, no se posee evidencia visual respecto a lo realizado dentro de este *Sprint*.

4.7. Sprint N°7

En el séptimo *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de trabajar nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.7.1. Sprint N° 7

4.7.1.1. 1° Weekly Meeting

En esta primera reunión se busca trazar un plan de trabajo para finalizar el trabajo pendiente del *Sprint* anterior, con intenciones de dar inicio lo antes posible a las nuevas vistas.

4.7.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°7.

- **Diseñar vista de inicio:** Esta vista será la que cualquier persona verá al momento de abrir SIAP, ya sea el cliente u otra persona.
- **Diseñar HEADER:** El HEADER es la sección de arriba de cada sitio web, en donde podemos encontrar, por lo general, la navegación del sitio.
- **Diseñar FOOTER:** El FOOTER es la sección de más abajo de un sitio web, en donde por lo general se encuentran preguntas frecuentes de los visitantes, datos de interés o medios de contacto, como redes sociales o correos electrónicos.
- **Diseñar vista sobre nosotros:** SIAP, al ser una aplicación web destinada al uso tanto por el profesional como por sus pacientes, se debe presentar información respecto a este, en donde se podrá apreciar sus métodos de trabajo, tarifas, experiencia, etc.

- **Diseñar vista contacto:** Nuevamente, al ser SIAP una aplicación web destinada al uso de pacientes o gente en general, en este caso, estas personas deben poseer un medio de contacto directo con el profesional. Es en esta vista donde se presentará toda la información necesaria para el contacto a través de diferentes vías.

4.7.1.3. 2° Weekly Meeting

En esta segunda reunión semanal se tomaron diversas decisiones respecto al progreso de SIAP, debido a que el equipo de desarrollo actualmente estará con un poco mayor de libertad, se decidió apresurar el desarrollo, teniendo en cuenta la fecha límite, además de intentar no sobrecargar un *Sprint*, intentando que no queden tareas inconclusas, pasando a un próximo *Sprint*.

4.7.2. Sprint Review

En esta reunión se llegó a conclusiones respecto al futuro del proyecto, con la intención de agilizar su desarrollo y su documentación.

4.7.3. Sprint Retrospective

Nuevamente el equipo de desarrollo ha deliberado respecto al progreso del proyecto, llegando a conclusiones que se esperan aplicar dentro de los próximos *Sprints*, con la intención de lograr un desarrollo veloz y sin problemas.

4.7.4. Increment

Como resultado de este *Sprint* se obtuvo como resultado el diseño del HEADER y el FOOTER a utilizar por la aplicación. Sin embargo, estos pueden resultar modificados en caso de que se vea necesario, agregando o eliminando enlaces o funciones. Estos son presentados en las siguientes subsecciones.

4.7.4.1. HEADER

En la siguiente imagen se puede apreciar el diseño deseado para el **HEADER**, sin embargo, no se especifica que sea el diseño final, debido a que puede sufrir cambios a futuro.

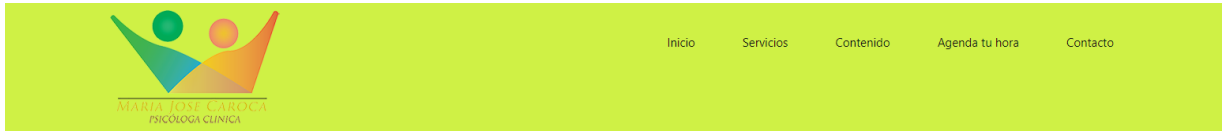


Figura 3.6: Diseño de HEADER [Elaboración propia]

4.7.4.2. FOOTER

En la siguiente imagen se puede apreciar el diseño deseado para el **FOOTER**, sin embargo, al igual que con el HEADER, no se especifica que sea el diseño final, debido a que puede sufrir cambios a futuro.



Figura 3.7: Diseño de FOOTER [Elaboración propia]

4.8. Sprint N°8

En el octavo *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de trabajar nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.8.1. Sprint N° 8

4.8.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior.

4.8.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°8.

- **Diseñar vista sobre nosotros:** SIAP, al ser una aplicación web destinada al uso tanto por el profesional como por sus pacientes, se debe presentar información respecto a este, en donde se podrá apreciar sus métodos de trabajo, tarifas, experiencia, etc.
- **Diseñar vista contacto:** Nuevamente, al ser SIAP una aplicación web destinada al uso de pacientes o gente en general, en este caso, estas personas deben poseer un medio de contacto directo con el profesional. Es en esta vista donde se presentará toda la información necesaria para el contacto a través de diferentes vías.
- **Diseñar vista de inicio:** Nuevamente, al ser SIAP una aplicación web destinada al uso de pacientes o gente en general, se desea presentar información de la profesional a toda persona que ingrese a SIAP. Es en esta vista en donde se presentará toda la información que se crea pertinente presentar a los posibles pacientes, con la intención de presentar a la profesional.
- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.

4.8.1.3. 2° Weekly Meeting

El equipo ha decidido realizar un cambio en las vistas, la información que se planteaba presentar en **Sobre nosotros**, finalmente se presentará dentro de la vista **Inicio**. Por otra parte, se agregará la vista **Servicios**, en donde se presentan los servicios que realiza la profesional, en conjunto con las enfermedades tratadas. Esto para mantener un mayor orden y presentar la

información de manera más similar a como se presenta en <https://www.doctoralia.cl>, a petición de la profesional.

4.8.2. Sprint Review

En esta reunión se discutió un cambio que solicitó la profesional durante el desarrollo de SIAP, esto no representó un gran problema, sin embargo, si retrasó de cierta manera el desarrollo de la aplicación. Se solicitó a la profesional que no hiciera más cambios, en caso de requerir alguno, debería solicitarlo de inmediato.

4.8.3. Sprint Retrospective

Se debe considerar siempre que pueden ocurrir cambios a último momento, ya sea por parte del equipo de desarrollo como por el cliente. En este caso, la profesional solicitó un cambio luego de que el equipo de desarrollo tuviese listo lo que originalmente se planteó, provocando un pequeño retraso respecto a lo planeado originalmente.

4.8.4. Increment

Debido al cambio presentado por la cliente, en este *Increment* solo se presentará la vista de **Inicio**, la cual fue modificada durante la reunión para poder seguir trabajando sin mayores inconvenientes a futuro. Fuera de lo que es programación, se realizó el bosquejo inicial del documento de memoria y el [Manual de Desarrollador](#). Estos son presentados en las siguientes subsecciones.

4.8.4.1. Vista inicio

En la siguiente imagen se puede apreciar el diseño final de la vista **Inicio**, luego de que se le aplicaran los cambios solicitados por la profesional.

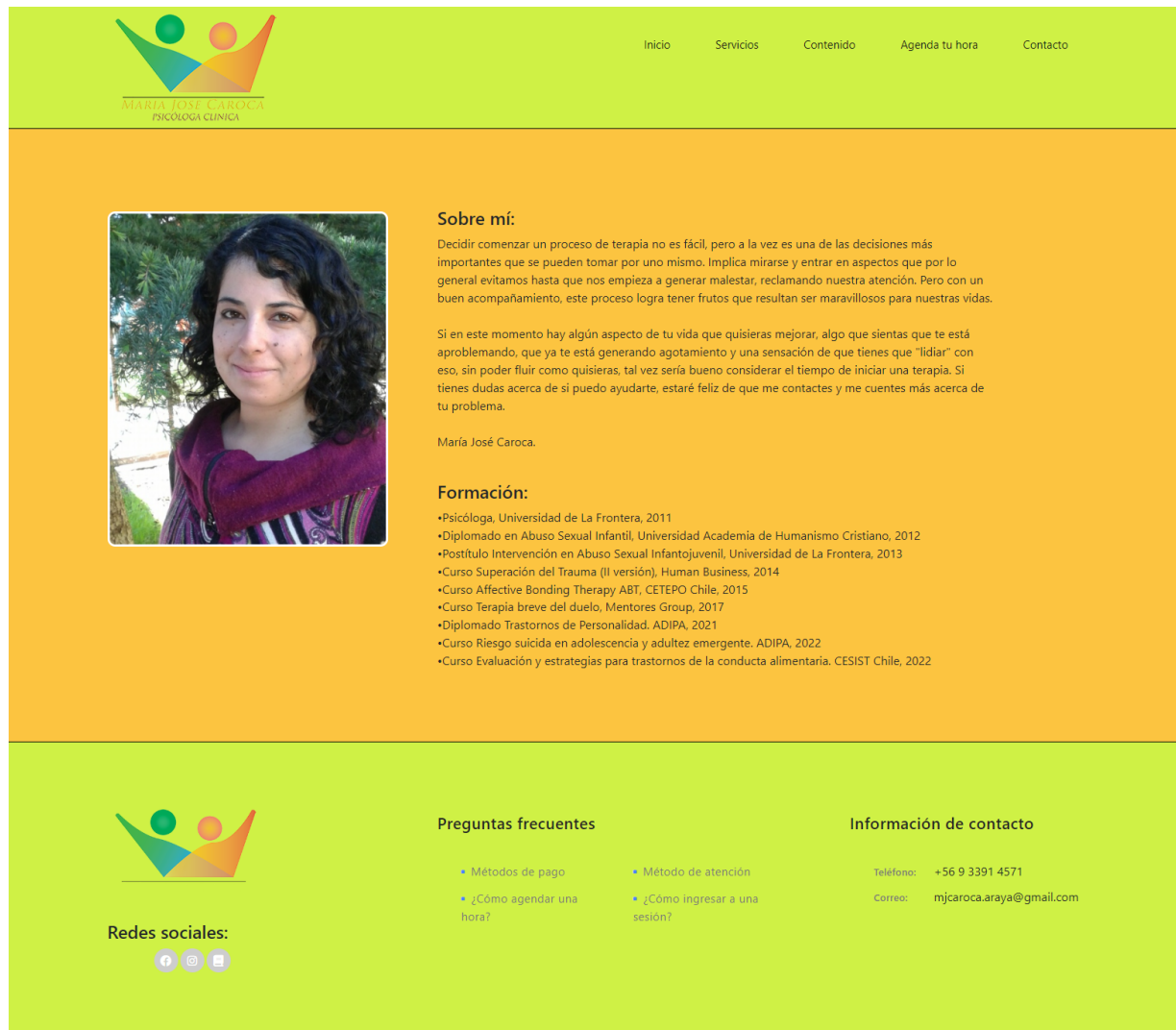


Figura 3.8: Vista inicio en SIAP [Elaboración propia]

4.8.4.2. Redacción de documento de memoria

Se deja constancia de que el documento de memoria ya ha sido iniciado, con su formato y todos los puntos solicitados para la evaluación.



UNIVERSIDAD CATÓLICA DEL MAULE

Facultad de Ciencias de la Ingeniería

Profesor Guía

Escuela de Ingeniería Civil Informática

Dr. Marco Toranzo

SIAP: SISTEMA DE AGENDA E INFORMACIÓN PARA PSICÓLOGA

LUIS JOAQUÍN CAROCA ARAYA

DAVID ALFREDO GONZÁLEZ ELGUETA

Memoria para optar al

Título Profesional de Ingeniero Ejecución en Computación e Informática

Talca, Octubre 2022

Figura 3.9: Portada del documento de memoria [Elaboración propia]

4.8.4.3. Redacción de Manual de Desarrollador

Se deja constancia de que el [Manual de Desarrollador](#) ya ha sido iniciado, indicando como obtener el proyecto, las tecnologías utilizadas y sus versiones exactas, para evitar problemas de compatibilidad.



Universidad Católica
Memoria

Manual de Desarrollador

SIAP
(Sistema de Agenda e Información para Psicóloga)

Luis Caroca Araya
David González Elgueta

Figura 3.10: Portada de manual de desarrollador [Elaboración propia]

4.9. Sprint N°9

En el noveno *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de continuar el trabajo de nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.9.1. Sprint N° 9

4.9.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior.

4.9.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°9.

- **Diseñar vista servicios:** SIAP, al ser una aplicación web destinada al uso tanto por el profesional como por sus pacientes, se debe presentar información respecto a este, en donde se podrán apreciar sus métodos de trabajo, enfermedades tratadas, entre otra información que se desee presentar.
- **Diseñar vista contacto:** Nuevamente, al ser SIAP una aplicación web destinada al uso de pacientes o gente en general, en este caso, estas personas deben poseer un medio de contacto directo con el profesional. Es en esta vista donde se presentará toda la información necesaria para el contacto a través de diferentes vías.
- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo

original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.

4.9.1.3. 2° Weekly Meeting

El equipo de desarrollo estuvo investigando la manera más óptima para la implementación de un calendario para la toma de horas de atención. Sin llegar a una conclusión debido a la falta de información, se continúa con otras vistas.

4.9.2. Sprint Review

En esta reunión el equipo, discutió como a lo largo de este *Sprint* priorizó el desarrollo de la documentación necesaria, debido a los retrasos que esta estaba sufriendo, dejando un poco de lado la programación, sin embargo. Además, se continúa discutiendo la búsqueda de la mejor manera de implementar un calendario de atención.

4.9.3. Sprint Retrospective

Algunas veces existen veces que el equipo de desarrollo debe priorizar una actividad sobre otra. En este caso, al ser 2 miembros en el equipo de desarrollo, uno pudo trabajar netamente la documentación, mientras que el otro además del documento, también pudo trabajar de menor manera en el proyecto.

4.9.4. Increment

El equipo de desarrollo continuó con la redacción de la documentación, corrigiendo errores de versiones, entre otras cosas. Además, se logró completar la vista **Servicios**, la cual había quedado pendiente desde el *Sprint* anterior.

4.9.4.1. Vista servicios

En la siguiente imagen se puede apreciar el diseño final de la vista **Servicios**, en donde se presenta la información de las enfermedades tratadas y los mismos servicios que entrega la profesional.



Figura 3.11: Vista servicios [Elaboración propia]

4.10. Sprint N° 10

En el décimo *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de cotinuar el trabajo de nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.10.1. Sprint N° 10

4.10.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior.

4.10.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°10.

- **Diseñar vista contacto:** Al ser SIAP una aplicación web destinada al uso de pacientes o gente en general, en este caso, estas personas deben poseer un medio de contacto

directo con el profesional. Es en esta vista donde se presentará toda la información necesaria para el contacto a través de diferentes vías.

- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.

4.10.1.3. 2° Weekly Meeting

El equipo de desarrollo continúa su investigación sobre la manera más óptima para la implementación de un calendario para la toma de horas de atención. Sin llegar a una conclusión debido a la falta de información, se continúa con otras vistas.

4.10.2. Sprint Review

En esta reunión el equipo, vuelve a discutir como a lo largo de este *Sprint* nuevamente se priorizó el desarrollo de la documentación necesaria, principalmente la de la documentación de la memoria de título.

4.10.3. Sprint Retrospective

Existen ocasiones en las que las actividades de un proyecto deben ser repartidas de mejor manera para que este mismo no se vea retrasado en su desarrollo. En este caso, se espera no tener que hacer cambios que puedan retrasar la planificación del proyecto, sin embargo, en caso de que esto sea necesario, el equipo debe resolver la situación de mejor manera.

4.10.4. Increment

El equipo de desarrollo continuó con la redacción de la documentación, corrigiendo errores ortográficos que puedan haber existido, además de agregar las imágenes necesarias para

dichos documentos, entre otras cosas. Además, se logró completar la vista **Contacto**, la cual se presenta en la siguiente subsección.

4.10.4.1. Vista contacto

La vista contacto presenta métodos de contacto con la profesional, tanto de manera como mensajería directa, como poder enviar un correo, así como la función de enviar un mensaje vía WhatsApp o un correo vía gestor de correo electrónico. Sin embargo, se espera que a futuro agregar otros medios de contacto, o trasladar la vista hacia el **FOOTER**, en vez de que se presente en el **HEADER**.

Figura 3.12: Vista contacto [Elaboración propia]

4.11. Sprint N°11

En el undécimo *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de continuar el trabajo de nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.11.1. Sprint N° 11

4.11.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior.

4.11.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°11.

- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.
- **Diseñar vista lista de espera:** SIAP, al estar orientada a la toma de horas de atención, se espera mantener un orden dentro de las mismas horas tomadas (bloqueadas) y disponibles. Sin embargo, la profesional plantea la idea de generar una lista de espera para sus pacientes, en caso de que estos crean requerir una atención lo antes posible o, por lo menos, antes que la primera disponible en el calendario.
- **Redactar manual de usuario** Todo proyecto debe considerar siempre un manual de usuario, el cual pueda ser entregado, de manera física y/o digital, al usuario final, indicándole la utilización del sistema, para que este pueda apreciar todas las funciones presentes en el sistema.

4.11.1.3. 2° Weekly Meeting

El equipo de desarrollo logra dar con una solución al problema que se presentaba respecto al calendario para la atención de horas, sin embargo, deben documentarse para la correcta implementación.

4.11.2. Sprint Review

En esta reunión, el *Team*, presentó como a lo largo de este *Sprint* estuvo investigando de manera continua la mejor manera de implementar el sistema de calendario para la toma de horas, ya que se consideró necesario para continuar el proyecto. Además, se logró poner al día con el estado de la documentación, sin embargo, esta debe ser actualizada a medida que el proyecto continúe su desarrollo.

4.11.3. Sprint Retrospective

Un buen proyecto debe mantener siempre su documentación al día, en caso de que exista algún problema o se requiera la búsqueda de cuando se realizó cierta actividad dentro del proyecto, se puede revisar todo esto dentro de su documentación. Además, la documentación permitirá a futuros desarrolladores realizar cambios al proyecto si es necesario.

4.11.4. Increment

Debido a que en este *Sprint* los mayores avances fueron tanto a nivel de investigación como de documentación, no se presenta ningún avance visible para este *Increment*.

4.12. Sprint N° 12

En el duodécimo *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de continuar el trabajo de nuevas vistas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.12.1. Sprint N° 12

4.12.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior.

4.12.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°12.

- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.
- **Diseñar vista lista de espera:** SIAP, al estar orientada a la toma de horas de atención, se espera mantener un orden dentro de las mismas horas tomadas (bloqueadas) y disponibles. Sin embargo, la profesional plantea la idea de generar una lista de espera para sus pacientes, en caso de que estos crean requerir una atención lo antes posible o, por lo menos, antes que la primera disponible en el calendario.
- **Redactar manual de usuario** Todo proyecto debe considerar siempre un manual de usuario, el cual pueda ser entregado, de manera física y/o digital, al usuario final, indicándole la utilización del sistema, para que este pueda apreciar todas las funciones presentes en el sistema.

4.12.1.3. 2° Weekly Meeting

El equipo de desarrollo se encuentra con el problema de que la librería seleccionada para la implementación del calendario, además de estar desactualizada, presentaba errores visuales, complicando el entendimiento para los usuarios, lo que provocaría la inutilización del sistema.

4.12.2. Sprint Review

En esta reunión el equipo da nuevamente con problemas para la implementación del calendario de atención, es por esto que deciden incluir un *DatePicker* básico simplemente, con la intención de progresar con otras tareas pendientes, además de dar por finalizada esta *issue*.

4.12.3. Sprint Retrospective

Algunas veces los proyectos pueden verse retrasados por tareas específicas, es por esto que se deben distribuir de mejor manera las actividades dentro de un proyecto, para que la realización no se vea interrumpida por completo por una tarea en específico.

4.12.4. Increment

Debido a que en este *Sprint* los mayores avances fueron tanto a nivel de investigación como de documentación, no se presenta ningún avance visible para este *Increment*.

4.13. Sprint N°13

En el décimo tercer *Sprint* se busca concretar los *issues* pendientes del *Sprint* anterior, además de continuar el trabajo de nuevas vistas, intentando dar por finalizada la creación de todas estas. Tiene una duración de 2 semanas, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.13.1. Sprint N° 13

4.13.1.1. 1° Weekly Meeting

En esta primera reunión se busca concretar el plan de trabajo para este *Sprint*, realizando un plan de trabajo con la intención de completar las tareas pendientes del *Sprint* anterior. Además, el equipo de desarrollo presenta la intención de dar por finalizada la programación del sistema dentro de este *Sprint*, sin embargo, no todas las tareas fueron consideradas dentro del **Backlog**, con la intención de no generar una sobrecarga de este mismo. A pesar de esto, en caso de concretar todas las actividades de programación, se realizarán actividades fuera del **Sprint Backlog**, las cuales serán representadas en los siguientes *Sprints* al momento de la documentación.

4.13.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°13.

- **Diseñar vista agendar hora:** SIAP, al ser una aplicación destinada a la toma y gestión de horas para atención psicológicas, debe permitir el poder registrar estas horas para que los pacientes puedan ser atendidos.
- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de desarrollador:** Todo proyecto debe considerar siempre un manual de desarrollador, con la intención de que si alguien, distinto al equipo de desarrollo original, desea realizar algún cambio al proyecto, este tenga una guía entregada, indirectamente, por el equipo original.
- **Diseñar vista lista de espera:** SIAP, al estar orientada a la toma de horas de atención, se espera mantener un orden dentro de las mismas horas tomadas (bloqueadas) y disponibles. Sin embargo, la profesional plantea la idea de generar una lista de espera para sus pacientes, en caso de que estos crean requerir una atención lo antes posible o, por lo menos, antes que la primera disponible en el calendario.
- **Redactar manual de usuario** Todo proyecto debe considerar siempre un manual de usuario, el cual pueda ser entregado, de manera física y/o digital, al usuario final, indicándole la utilización del sistema, para que este pueda apreciar todas las funciones presentes en el sistema.
- **Diseñar vistas adicionales necesarias:** SIAP, al estar dirigida principalmente al usuario promedio, se debe considerar que no todos sabrán como dar con la información requerida por ellos mismos. Es por esto por lo que se planea incorporar distintas vistas, en donde se presenten ayudas para los usuarios, ya sea con información adicional o indicando como tomar una hora o como hacer ingreso a estas.
- **Agregar base de datos a archivo .env:** Gracias a que se está trabajando con tecnología *Backend* (en este caso *Laravel*), podemos ingresar toda la información de nuestra base de datos en un archivo *.env*, el cual nos servirá para hacer la conexión a dicha base de datos. Además de esto, se debe crear la migración de la misma, con intención de que esta se cargue automáticamente y con el usuario base para la administración de dicha base de datos.

4.13.1.3. 2° Weekly Meeting

El equipo de desarrollo realiza la configuración deseada para el calendario de toma de horas de atención, además de continuar el diseño de las distintas vistas a utilizar dentro de SIAP.

4.13.2. Sprint Review

En esta reunión el equipo de desarrollo da por finalizada el diseño de todas las vistas de SIAP, quedando pendientes solamente las que requieren usar la base de datos para mostrarse. Luego de una detallada revisión, se da visto bueno y por finalizadas todas las vistas de SIAP.

4.13.3. Sprint Retrospective

Existen algunas ocasiones en las que los proyectos pueden presentarse como algo con un final inalcanzable, sin embargo, al finalizar un conjunto de tareas, este se ve mucho más realizable.

4.13.4. Increment

El *Team* en este *Sprint* decidió dar por finalizada el diseño de todas las vistas principales, quedando pendiente solamente pendientes las vistas de reportes. Además de esto, se completó documentación relacionada al proyecto. A continuación, se deja evidencia de las *issues* completadas.

4.13.4.1. Vista agendar hora

La vista agendar hora permite a los usuarios agendar horas para atención psicológica a través de un formulario que luego cargará la información a la base de datos.

Inicio Servicios Agenda tu hora Contacto Iniciar sesión

Rut:
Rut

Agendar Hora: Fecha de atención

Agendar Hora: Fecha de atención

Agendar

En caso de no haber horas disponibles, pero que sientas que requieres atención con urgencia, puedes añadirte a la [lista de espera](#) detallando tu caso.

Redes sociales:

Preguntas frecuentes

Métodos de pago
¿Cómo agendar una hora?

Método de atención
¿Cómo ingresar a una sesión?

Información de contacto

Teléfono: +56 9 3391 4571
Correo: mjcaroca.araya@gmail.com

Figura 3.13: Vista agendar hora [Elaboración propia]

4.13.4.2. Vista lista de espera

La vista lista de espera permite a los usuarios agregarse a una lista en caso de que no existan horas disponibles acordes a su disponibilidad o en caso de que el usuario requiera una atención de urgencia. Estos casos serán evaluados por la psicóloga y ella se contactará con los usuarios para acordar una hora.

Nombre Completo:
Nombres Apellido Paterno Apellido Materno

Rut: Teléfono:

Correo electrónico: Fecha de nacimiento:

Ciudad:

Motivo de espera:

Preguntas frecuentes

- Métodos de pago
- ¿Cómo agendar una hora?
- Método de atención
- ¿Cómo ingresar a una sesión?

Información de contacto

Teléfono: +56 9 3391 4571
Correo: mjcaroca.araya@gmail.com

Redes sociales:

Figura 3.14: Vista lista de espera [Elaboración propia]

4.13.4.3. Agregar base de datos a archivo .env

La base de datos a utilizar por SIAP se agregó al archivo .env, para poder empezar a realizar las conexiones necesarias. Además de esto, se crearon las migraciones de la misma base de datos, para que esta pueda ser cargada sin problemas en cualquier entorno. Los pasos de cargado de las migraciones están explicados en el [Manual de Desarrollador](#).

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=db_psicologa
DB_USERNAME=root
DB_PASSWORD=
```

Figura 3.15: Conexión de base de datos [Elaboración propia]

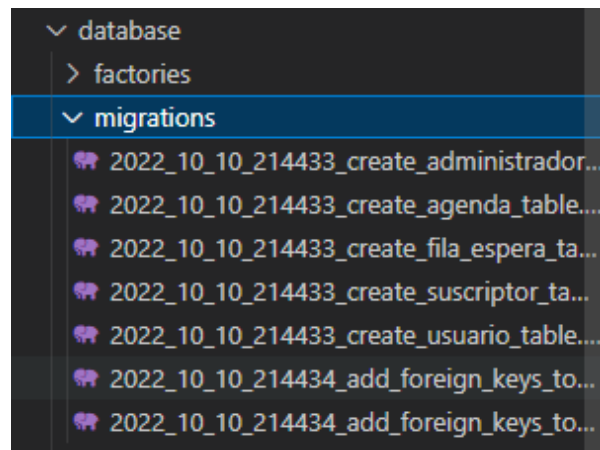


Figura 3.16: Archivos de migraciones de base de datos [Elaboración propia]

4.13.4.4. Redactar manual de usuario

Se inicio la redacción del [Manual de Usuario](#), el cual servirá para documentar al usuario sobre las funciones que posee SIAP y el cómo utilizar esta aplicación de manera correcta. A continuación, se presenta la portada de dicho manual, debido a su extensión.



Universidad Católica
Memoria

Manual de Usuario

SIAP
(Sistema de Agenda e Información para Psicóloga)

Luis Caroca Araya
David González Elgueta

Figura 3.17: Portada de manual de usuario [Elaboración propia]

4.13.4.5. Vistas adicionales necesarias

Se crearon las vistas necesarias adicionales que no fueron solicitadas en primera instancia por la profesional, si no que fueron solicitadas a lo largo del desarrollo, como ayuda a presentar información relevante para los usuarios que hagan ingreso a SIAP. A continuación, se presentan dichas vistas diseñadas con una breve explicación de la información que representan.

- La vista **Métodos de pago** presenta la información de los valores de las consultas psicológicas, además de los medios de pago de estas mismas.

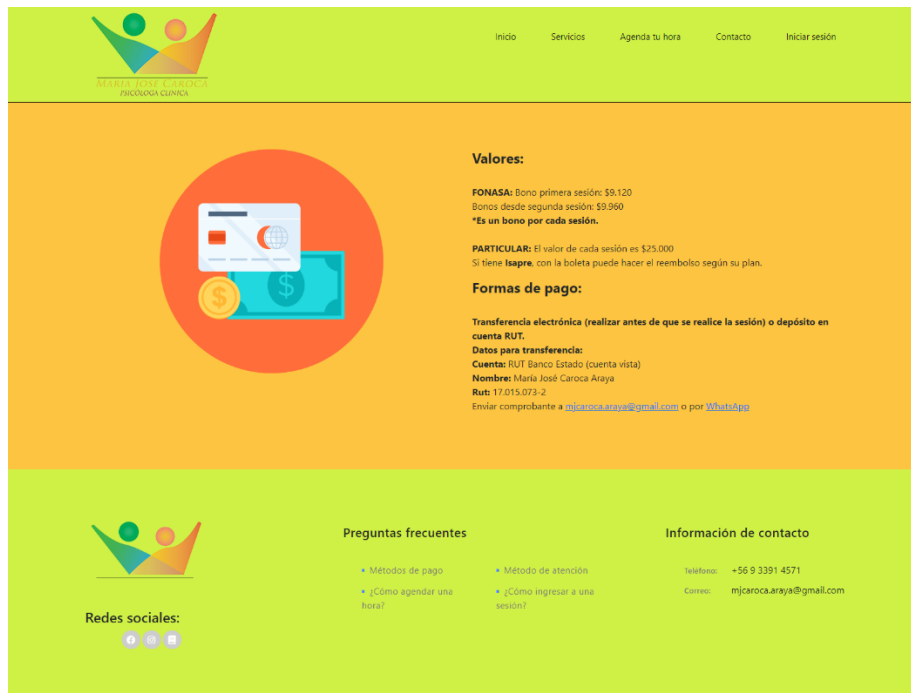


Figura 3.18: Vista métodos de pago [Elaboración propia]

- La vista **Métodos de atención** presenta la información de cómo se realizan las atenciones actualmente, esto variará a futuro.



Figura 3.19: Vista método de atención [Elaboración propia]

- La vista **¿Cómo agendar una hora?** presenta el paso a paso para agendar una hora con la profesional.

Primeros ve a **agendar tu hora** y rellena el formulario con los datos solicitados, al ya iniciarte sesión con tu cuenta, verifica que tus datos estén correctos.

En caso de no haber horas disponibles, pero que sientas que requieres atención con urgencia, puedes añadirte a la [lista de espera](#) contactando tu caso.

Debido a que en agenda se tiene una disponibilidad limitada de horas y días disponibles para la forma de horas. En caso de que no encuentres horas disponibles, o las disponibles no se acomoden a tu disponibilidad, puedes añadirte a la [lista de espera](#), donde podrás explicarme tu situación.

Una vez hayas rellena el formulario y seleccionado una hora disponible, haz clic en el botón **Agendar**. Si todo ha salido bien, aparecerá el siguiente mensaje en tu pantalla.

En caso de que no existan horas disponibles y sientas que requieres una atención con urgencia o si simplemente la agenda disponible no se acomode a tu disponibilidad, puedes agregarle a la [lista de espera](#) rellena el formulario que se puede apreciar en la siguiente imagen y detallando cuál tu caso. Yo evaluaré tu situación y me comunicare contigo para coordinar una atención dependiendo tu caso.

Una vez hayas rellena el formulario, haz clic en el botón **Agregar a lista de espera**. Si todo ha salido bien, aparecerá el siguiente mensaje en tu pantalla.

Hola! Ya tenemos agendada nuestra sesión online y quisiera contarte algunas consideraciones a tener en cuenta:

1. Te recomendamos hacer un lugar en que estás tranquila o tranquilo.
2. Si estás con más personas, avisa que estás ocupada u ocupado si tienes un letrero de "no molestar" para que no te interrumpen en la sesión.
3. La agenda se maneja desde este mismo sitio web, en donde se gestionan las horas. En caso de tener dudas, puedes contactar directamente o a través de la página de contacto.
4. Si día antes recibiste un recordatorio de nuestra sesión y ya estás listo para la sesión, te avisaremos a través de un correo.
5. Mientras no está cancelada la hora, ya asumirá que está confirmada y programará las videollamadas.
6. Usaremos la aplicación Zoom, que se puede usar desde el teléfono o el computador.
7. Si necesitas cancelar tu hora, te pido que lo hagas con la mayor antelación posible. Así me permites reprogramar la agenda y que alguien más pueda tomar ese horario. Pero si necesitas reprogramar, te pido que lo hagas con la mayor antelación posible. Así me permites reprogramar la agenda y que alguien más pueda tomar ese horario. Pero si necesitas reprogramar, te pido que lo hagas con la mayor antelación posible. Así me permites reprogramar la agenda y que alguien más pueda tomar ese horario.
8. Las sesiones son de 45 minutos.
9. En cuanto a los atrasos, yo hago todo lo posible para que esto no me afecte, así que también te pido lo mismo. Si yo me retrasara, con la tranquilidad de que tendría igual todo tu tiempo de sesión. Pero si luego me retraso, me respetar el horario según agenda, pero que tengas en cuenta y organízate antes tiempo que es bueno.
10. El pago de la sesión (por particular o con base en KINAVIA) será vía transferencia electrónica de dinero previa a iniciar la sesión. El comprobante debe ser enviado a mi correo mycarroa.amy@gmail.com o por whatsapp.

Nos encontramos pronto, un abrazo!

Preguntas frecuentes

¿Cómo agendar una hora?

¿Cómo agregar a una lista de espera?

Información de contacto


Teléfono: +56 9 6997 4077

Correo: mycarroa.amy@gmail.com

Redes sociales:

Figura 3.20: Vista ¿cómo agendar una hora? [Elaboración propia]

- La vista **¿Cómo ingresar a sesión?** presenta toda la información para ingresar a las sesiones en **Google Meet** con la profesional.

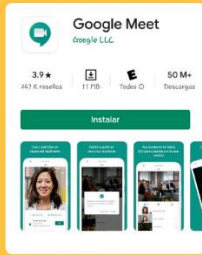


[Inicio](#)
[Servicios](#)
[Agenda tu hora](#)
[Contacto](#)
[Iniciar sesión](#)

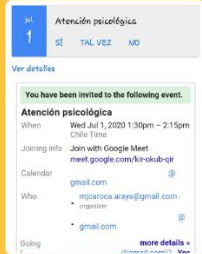
¿Cómo ingresar a las sesiones?

Ahora que ya tenemos agendada nuestra sesión, te explicaré cómo ingresar. Usaremos la aplicación Meet que es de gmail. Así que sólo necesitas tu correo para poder ingresar, ya que entrarás a la videollamada desde allí.

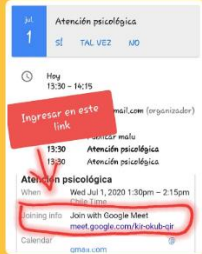
Se puede usar desde el computador o desde el teléfono. Desde el celular sí la tienes que tener instalada para ingresar. Esta es la aplicación para que la puedas descargar si es necesario:



Yo programaré la sesión y te enviaré una invitación por correo, que se verá algo así:



Y cuando sea la hora de tener la sesión, debes pinchar en este link:




Verás algo así, y tienes que pinchar en "Unirse a la reunión".


Yo ya estaré conectada. Y si dice como en la imagen "Eres el primero en llegar", no te preocupes, que ya estaré.

Para mí es muy importante la puntualidad, por eso cuido mucho los tiempos.


También puede ser que en el correo diga un horario distinto del que habíamos agendado, pero ese no lo tomes en cuenta, porque a veces se cambia por el huso horario.

Siempre la hora válida será la que hayamos agendado y que se recordará a través del mensaje de la página de doctoralia. ¡Nos vemos!





Redes sociales:



Preguntas frecuentes

- Métodos de pago
- ¿Cómo agendar una hora?
- Método de atención
- ¿Cómo ingresar a una sesión?

Información de contacto

Teléfono: +56 9 3391 4571

Correo: mjcaroca.araya@gmail.com

Figura 3.21: Vista **¿cómo ingresar a sesión?** [Elaboración propia]

4.14. Sprint N°14

En el décimo cuarto *Sprint* se busca finalizar el proyecto. Tiene una duración de 1 semana, en donde se espera poder realizar todo el trabajo planteado, además de que estos se vean aprobados en su totalidad en las *Weekly Meeting*.

4.14.1. Sprint N° 14

4.14.1.1. 1° Weekly Meeting

En esta primera reunión se busca dejar en claro todas las conexiones con la base de datos que se requieren, debido a que es lo único que está pendiente para completar el desarrollo de SIAP. Además de estimar el tiempo para finalizar la documentación pendiente y documentar los puntos presentados en este **Backlog**.

4.14.1.2. Sprint Backlog

A continuación, se describe la lista de tareas para el *Sprint* N°14.

- **Diseñar vista Perfil:** SIAP, el tener control de usuarios, debe poseer una vista de perfil con el cual los usuarios pueden visualizar su información o modificarla, si es que lo desean.
- **Diseñar vista Login:** SIAP, al tener control de usuarios, debe poseer un Login o sistema de inicio de sesión, con la cual los usuarios puedan hacer ingreso con sus respectivos datos.
- **Diseñar vista Recuperar contraseña:** SIAP, al tener control de usuarios, debe poseer la opción de recuperación de contraseña, con la finalidad de que, si un usuario no recuerda dicha contraseña, pueda volver acceder a su cuenta sin problemas.
- **Diseñar vistas adicionales necesarias:** SIAP, al estar dirigida principalmente al usuario promedio, se debe considerar que no todos sabrán como dar con la información requerida por ellos mismos. Es por esto por lo que se planea incorporar distintas vistas, en donde se presenten ayudas para los usuarios, ya sea con información adicional o indicando como tomar una hora o como hacer ingreso a estas. En el caso de este *Sprint* se consideran solamente las distintas vistas que presentarán reportes para los usuarios de tipo Administrador.

- **Redactar documento de memoria:** El documento de memoria debe ser presentado, en una primera instancia, al profesor guía, con la intención de recibir una evaluación al ramo asignado. Posteriormente, dicho documento debe ser presentado a las entidades evaluadoras para la defensa de la memoria de título.
- **Redactar manual de usuario** Todo proyecto debe considerar siempre un manual de usuario, el cual pueda ser entregado, de manera física y/o digital, al usuario final, indicándole la utilización del sistema, para que este pueda apreciar todas las funciones presentes en el sistema.
- **Realizar conexiones con base de datos:** Todo proyecto que deba almacenar información cuenta con su propia base de datos, es por esto que para SIAP se considera realizar las conexiones necesarias para hacer consultas requeridas para que SIAP funcione de manera óptima. Esto considera la creación de controladores, rutas de envío de información, modelos, etc.
- **Realizar consultas a la base de datos:** Todo proyecto que trabaje con una base de datos debe contener las consultas necesarias para trabajar de manera completa, ya sean consultas de inserción de datos, actualización, eliminación o desactivación.
- **Realizar pruebas de control a base de datos:** De igual manera, todo proyecto que trabaje con una base de datos debe recibir un control de pruebas para confirmar que estas funcionen sin problemas y no causen errores en la misma base de datos o en SIAP en sí.
- **Realizar correcciones a documento de memoria:** Se espera poder finalizar con tiempo de sobra el proyecto de SIAP, es por eso por lo que se cuenta con que la documentación podrá ser presentada al **Scrum Master** y **Product Owner**, con la intención de que este nos pueda dar una retroalimentación, con la intención de saber que corregir dentro de todo el proyecto SIAP.

4.14.1.3. 2° Weekly Meeting

Esta segunda reunión fue realizada en menos de una semana de desarrollo de SIAP, debido al tiempo disponible para su finalización. En esta *Meeting* el *Team* realiza presentación de sus últimos avances al resto del equipo, con el fin de poder incorporar todas las funcionalidades pendientes dentro del proyecto.

4.14.2. Sprint Review

En esta reunión se dio por finalizado el desarrollo de SIAP para la entrega del documento de la memoria. También se discutió entre el equipo de desarrollo el formato final de la documentación para los distintos documentos a entregar.

4.14.3. Sprint Retrospective

SIAP fue finalizado de manera visual y de manera funcional, sin embargo, aún no puede ser puesto en producción, debido a la falta de validación de campos que puedan generar problemas dentro de un sistema ya funcionando en la web. Esto es debido a que SIAP aún está en periodo de pruebas de control de usuarios, control de datos y soporte de carga de datos dentro de un servidor local. Sin embargo, ya se cuenta con el visto bueno de la profesional y una vez finalizadas las distintas pruebas de control de datos, se espera dar paso a SIAP para entrar en producción. También se espera que en el momento en que SIAP pase a ser una aplicación web de uso público, este se encuentre montado dentro de un servidor que pueda soportar una gran carga de datos, es decir, en un servidor de pago y con su propio dominio.

4.14.4. Increment

El *Team* durante este *Sprint* dio por finalizado el desarrollo de SIAP, dejando pendientes solamente algunos detalles antes de pasar el proyecto a producción, como pueden ser validaciones de datos y formularios dentro del proyecto en general. A continuación, se presentan las evidencias de las distintas *issues* realizadas durante este *Sprint*.

4.14.4.1. Vista perfil

La vista perfil permite al usuario visualizar los datos de su cuenta, además de tener la posibilidad de editar estos datos en casos de que se percate de tener un dato erróneo.

SECTOR DE SALUD
FARMACIA CLÍNICA

Inicio Servicios Agenda tu hora Contacto Iniciar sesión

Nombre Completo:
 Nombres Apellido Paterno Apellido Materno

Contraseña:
 Contraseña Confirmar contraseña

Rut:
 Rut

Teléfono:
 Teléfono

Correo electrónico:
 Correo electrónico

Fecha de nacimiento:
 Fecha de nacimiento

Ciudad:
 Ciudad

Motivo de Consulta:
 Motivo de consulta

[Editar](#)

Usted tiene una hora agendada para:

Usted está en fila de espera:

Usted está suscrito a nuestro boletín:

Redes sociales:
[Facebook](#) [Twitter](#) [Instagram](#)

Preguntas frecuentes

- Métodos de pago
- Método de atención
- ¿Cómo agendar una hora?
- ¿Cómo ingresar a una sesión?

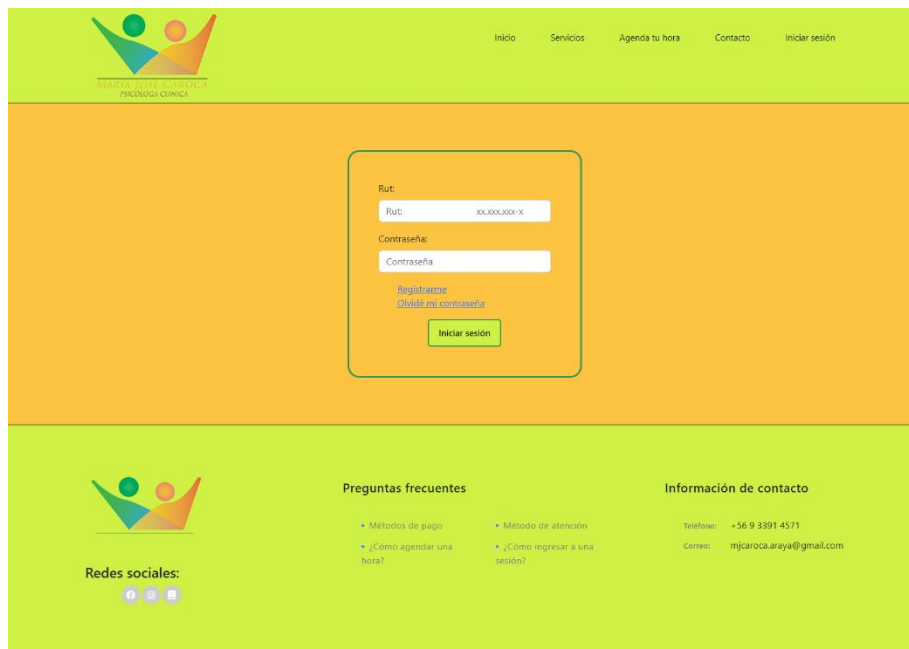
Información de contacto

Teléfono: +56 9 3391 4571
 Correo: mjcarroca.anysa@gmail.com

Figura 3.22: Vista perfil [Elaboración propia]

4.14.4.2. Vista login

La vista login permitirá a los usuarios de SIAP hacer ingreso con sus credenciales una vez que hayan hecho un previo registro de sus cuentas.

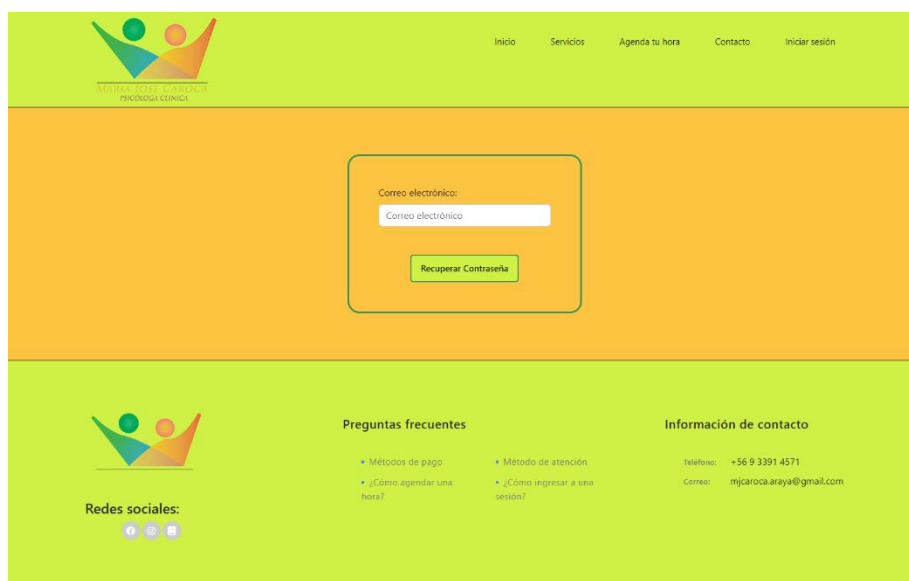


The login form is displayed on a light green background. At the top left is the logo for 'MIRASOL EDE S.A. CLINICA PROLOGO CLINICA'. The top navigation bar includes links for 'Inicio', 'Servicios', 'Agenda tu hora', 'Contacto', and 'Iniciar sesión'. The main form area has a light orange background and contains fields for 'Rut:' and 'Contraseña:'. Below these fields are links for 'Regístrate' and 'Olvidé mi contraseña', and a green 'Iniciar sesión' button. The footer section, also on a light green background, features the logo, 'Redes sociales' with Facebook, Twitter, and Instagram icons, 'Preguntas frecuentes' with links for payment methods, appointment scheduling, and login issues, and 'Información de contacto' with phone and email details.

Figura 3.23: Vista login [Elaboración propia]

4.14.4.3. Vista recuperar contraseña

La vista recuperar contraseña permitirá a los usuarios solicitar el reestablecer la contraseña de su cuenta a través de su correo electrónico. Esto es importante para el control de usuario en caso de que dicho usuario olvide su contraseña y pierda el acceso a su cuenta.



The password recovery form is displayed on a light green background. At the top left is the logo for 'MIRASOL EDE S.A. CLINICA PROLOGO CLINICA'. The top navigation bar includes links for 'Inicio', 'Servicios', 'Agenda tu hora', 'Contacto', and 'Iniciar sesión'. The main form area has a light orange background and contains a field for 'Correo electrónico:'. Below this field is a green 'Recuperar Contraseña' button. The footer section, also on a light green background, features the logo, 'Redes sociales' with Facebook, Twitter, and Instagram icons, 'Preguntas frecuentes' with links for payment methods, appointment scheduling, and login issues, and 'Información de contacto' with phone and email details.

Figura 3.24: Vista recuperar contraseña [Elaboración propia]

4.14.4.4. Vistas adicionales necesarias

Al igual que en el *Sprint* anterior, se crearon las vistas que se consideraron necesarias para la utilización correcta de SIAP, en este caso, enfocadas principalmente en su área de administración. A pesar de que la cliente especificó en un principio que no era necesario incluir en SIAP un sistema de listado de ningún tipo de información, se le recomendó incorporarlo, debido a que de igual forma ella necesitaba realizar distintas consultas a la información dentro de SIAP para sus atenciones. A continuación, se presentan las vistas creadas.

- La vista **Opciones de administrador** presenta las distintas opciones que tendrá el administrador dentro de SIAP. Esto se planeaba integrar como un componente de tipo *sidebar*, sin embargo, la cliente especificó que no deseaba la incorporación de este. Sin embargo, se espera que, en una futura iteración de proyecto o reunión con ella, esta permita realizar el cambio.



Figura 3.25: Vista Opciones de administrador [Elaboración propia]

- La vista **Reportes de Usuarios** presenta una lista de los usuarios registrados dentro de SIAP. Además, permite a los administradores el añadir un nuevo usuario, editar o eliminar uno existente.

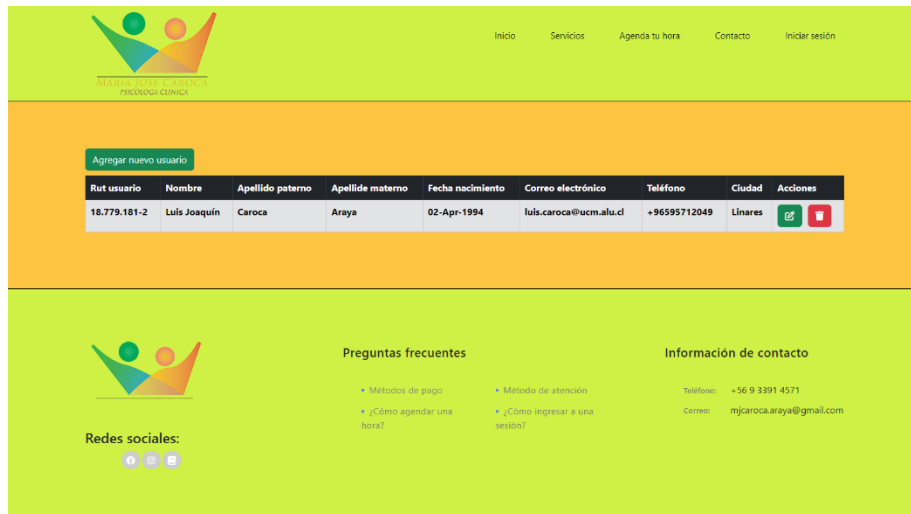


Figura 3.26: Vista Reporte de usuarios [Elaboración propia]

- La vista **Reporte de Suscriptores** presenta una lista de las personas que han realizado una suscripción vía correo electrónico al boletín de SIAP.

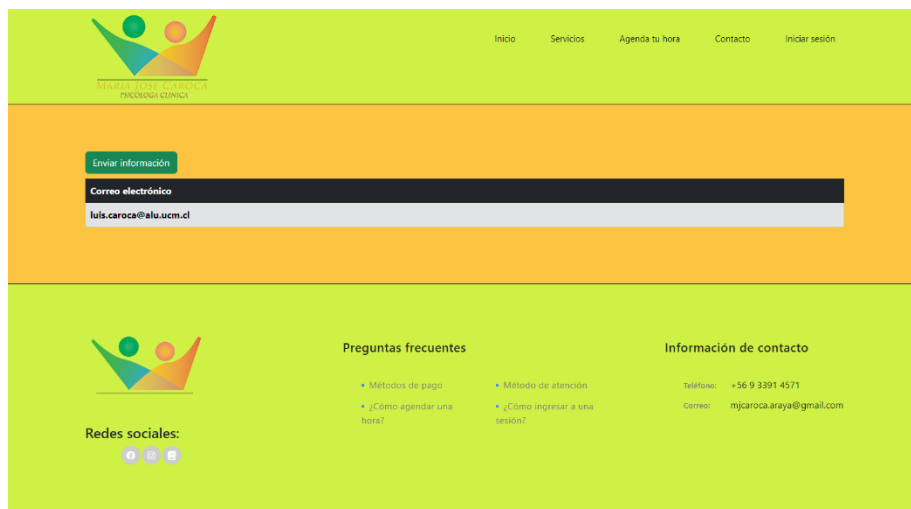


Figura 3.27: Vista Reporte de suscriptores [Elaboración propia]

- La vista **Reporte de Agenda** presenta una lista de las horas tomadas, presentando también el administrador a cargo de dicha atención (debido a que SIAP de momento solo cuenta con la psicóloga como administrador, no es necesario hacer una distinción actualmente), los datos del usuario que tiene registrada dicha hora, y los datos de fecha y hora de la sesión. Además, se presentan las acciones de editar y eliminar dicha hora en caso de que el administrador lo requiera. A futuro, previo a que SIAP sea puesto en producción, se deben ingresar los datos de la psicóloga con su respectivo cargo.

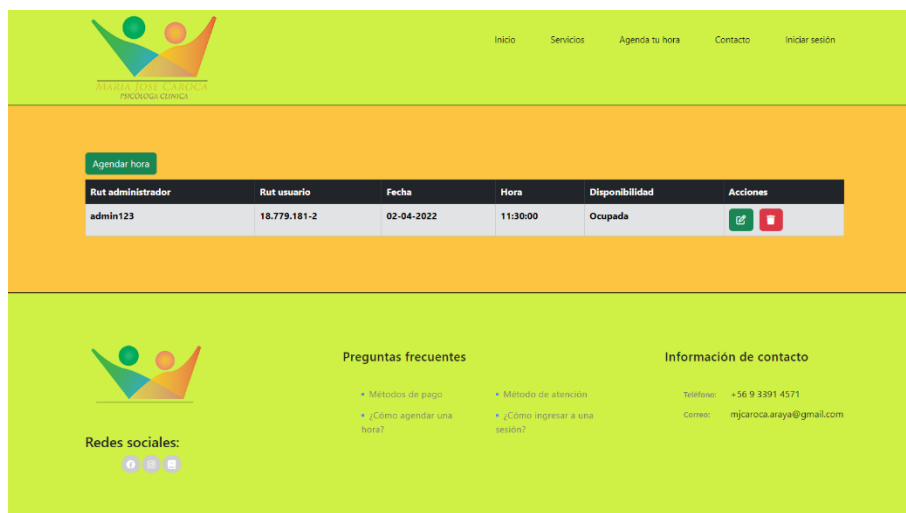

Rut administrador	Rut usuario	Fecha	Hora	Disponibilidad	Acciones
admin123	18.779.181-2	02-04-2022	11:30:00	Ocupada	 

Figura 3.28: Vista Reportes de agenda [Elaboración propia]

- La vista **Reporte de administradores** presenta una lista de todos los administradores que posee SIAP. Actualmente, como el sistema sigue en pruebas y se trabaja en un servidor local, se precargaron datos específicos para su testeo. Sin embargo, a futuro se deben precargar los datos específicos de la psicóloga, quien será la principal administradora de SIAP. Finalmente, dentro de esta vista se presenta la opción de editar, eliminar o agregar un nuevo administrador.

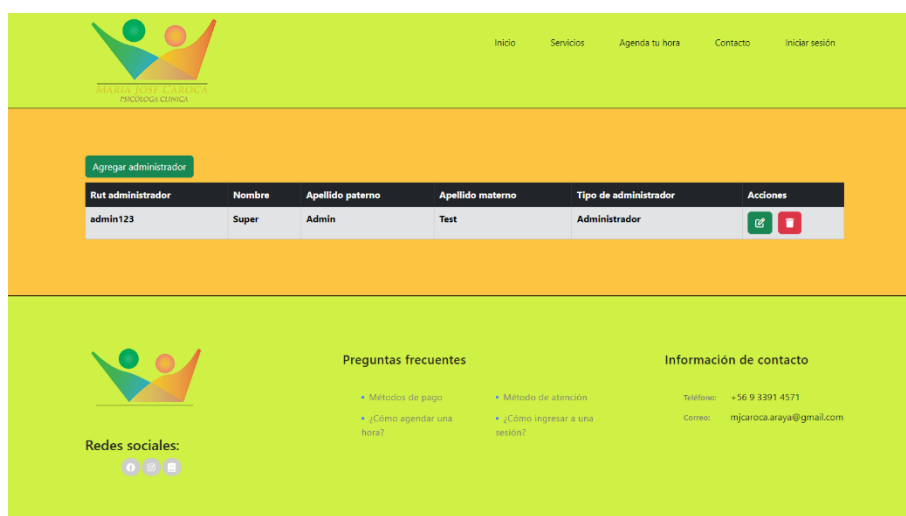

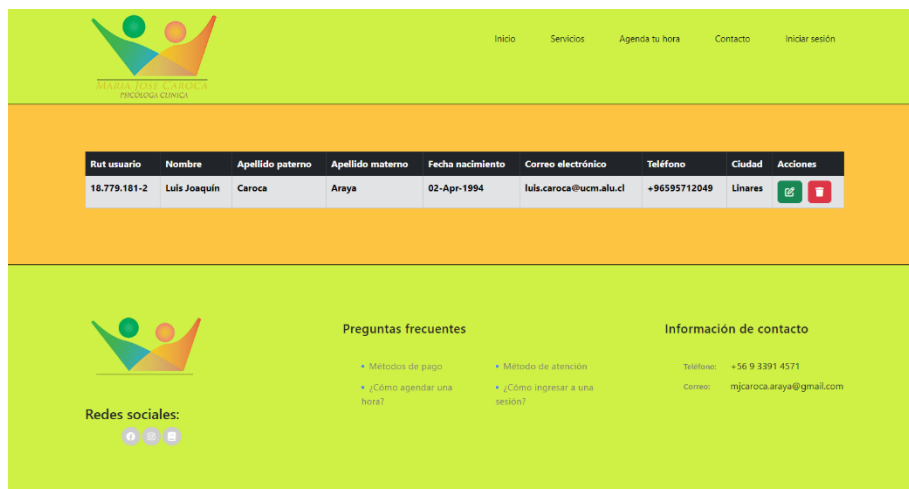
Rut administrador	Nombre	Apellido paterno	Apellido materno	Tipo de administrador	Acciones
admin123	Super	Admin	Test	Administrador	 

Figura 3.29: Vista Reporte de administradores [Elaboración propia]

- La vista **Reporte de Lista de espera** presenta todos los usuarios que hayan realizado un registro dentro de la lista de espera. La funcionalidad principal de este reporte es el poder asignar sesiones de atención a los usuarios registrados dentro de esta que más lo requieran, según criterio de la profesional. Además,

cuenta con la opción de eliminar registros, en caso de que se haya hecho la atención y agendado a través de otra vía.





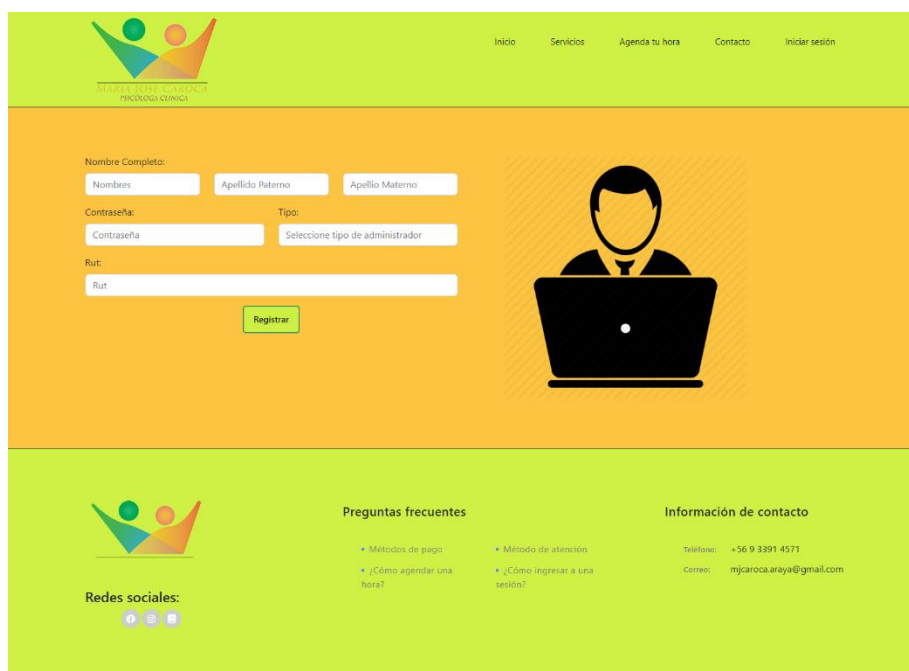
Rut usuario	Nombre	Apellido paterno	Apellido materno	Fecha nacimiento	Correo electrónico	Teléfono	Ciudad	Acciones
18.779.181-2	Luis Joaquín	Caroca	Araya	02-Apr-1994	luis.caroca@ucm.alu.cl	+96595712049	Linares	 

Figura 3.30: Vista Reporte de Lista de espera [Elaboración propia]

- La vista **Agregar administrador** permite el registro de un nuevo administrador dentro de SIAP. Se debe destacar que solo los administradores podrán realizar ingreso a esta vista, sin embargo, igual deberá ser protegida, con la intención de evitar que algún usuario malicioso pueda acceder a datos sensibles dentro de SIAP.



Nombre Completo:

Nombres
 Apellido Paterno
 Apellido Materno

Contraseña:

Contraseña
 Tipo: Seleccione tipo de administrador

Rut:

Rut

Figura 3.31: Vista Agregar administrador [Elaboración propia]

4.14.4.5. Redactar manual de usuario

Se finalizó la redacción del documento [Manual de Usuario](#), con el cual se espera presentar una guía de uso y navegación de SIAP al cliente, con intención de que pueda utilizarla sin problemas a futuro.



Universidad Católica
Memoria

Manual de Usuario

SIAP
(Sistema de Agenda e Información para Psicóloga)

Luis Caroca Araya
David González Elgueta

Figura 3.32: Portada Manual de usuario [Elaboración propia]

4.14.4.6. Realizar conexiones con base de datos

Toda aplicación que sirva para el almacenado y consulta de datos debe conexiones a la base de datos sin problemas. En este punto se volverá a mostrar la conexión realizada en el archivo .env de Laravel. Además, se mostrará un ejemplo de los **modelos** creados para el trabajo de Laravel a nivel **MVC** (Modelo Vista Controlador). Se debe especificar que dentro de los **modelos** debemos especificar los datos de nuestras tablas de la base de datos y las relaciones que estas presentan con otras tablas.

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=db_psicologa
DB_USERNAME=root
DB_PASSWORD=
```

Figura 3.33: Conexión a base de datos [Elaboración propia]

```
namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class usuarios extends Model
{
    use HasFactory;
    public $timestamps = false;
    protected $table = 'usuario';
    protected $fillable = [
        'rut_usuario',
        'contrasena',
        'nombre',
        'apellido_pat',
        'apellido_mat',
        'fecha_nacimiento',
        'correo',
        'telefono',
        'ciudad',
        'motivo_consulta'
    ];
    public function filas(){
        return $this->hasMany(fila_espera::class);
    }
    public function agenda(){
        return $this->hasMany(agenda::class);
    }
}
```

Figura 3.34: Modelo de tabla usuario [Elaboración propia]

En la figura 3.34 podemos apreciar el **modelo** de la tabla usuario. A continuación, se procederá a explicar los principales puntos de este archivo.

- **public \$timestamps = false:** esto indicará a nuestro **modelo** que no es necesario el almacenar los *timestamps*, los cuales hacen referencia a cuando dicho valor fue agregado a nuestra base de datos y cuando fue actualizado.
- **protected \$table = 'usuario':** esto indicará a nuestro **modelo** el nombre específico de la tabla que debe buscar dentro de la base de datos al momento de hacer consultas, debido a que Laravel, por defecto, busca dentro de la base de datos una tabla con el mismo nombre del **modelo**, pero con una letra **S** agregada al final. En este caso, de no declarar la tabla como *usuario*, Laravel buscaría la tabla *usuarios*.
- **protected \$fillable = []:** *fillable* es una propiedad que indica a los **modelos** los campos que son considerados al momento de hacer una consulta a la base de datos.
- **public function:** Las funciones declaradas dentro del **modelo** declaran las relaciones con otras tablas dentro de la base de datos. Por ejemplo, la función **agenda()** indica que un usuario puede tener muchas agendas registradas en la base de datos, es por esto que se utiliza la función **hasMany(agenda::Class)**, indicando que la relación se realiza con la **Class** (clase) que se encuentra dentro del **controlador** agenda. Por otra parte, dentro del **modelo** de la tabla agenda se realiza la acción contraria, gracias a la función **belongsTo()** (como se muestra en la figura 3.35), lo cual indica que las agendas pertenecen a un usuario en específico. Esta es la relación a nivel de codificación de las **Primary Key** (llaves primarias) con las **Foreign Key** (llaves foráneas) y Laravel por su cuenta resolverá que datos son los que deben compararse al momento de realizar las consultas.

```
public function usuarios(){  
    return $this->belongsTo(usuarios::class);  
}
```

Figura 3.35: Relación de la tabla agenda con la tabla usuario [Elaboración propia]

4.14.4.7. Realizar consultas a la base de datos

Las consultas a la base de datos a través de Laravel se realizan a través de los **controladores**, en los cuales se realizan todas las consultas pertinentes. En la figura 3.36 se pueden apreciar todas las funciones utilizadas dentro del **controlador** suscriptor.

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\suscriptor;
7
8  class suscriptorController extends Controller
9  {
10
11     public function index()
12     {
13         $suscriptor = suscriptor::all()->toArray();
14         return array_reverse($suscriptor);
15     }
16
17     public function store(Request $request)
18     {
19         $suscriptor = new suscriptor([
20             'correo' => $request->input('correo')
21         ]);
22         $suscriptor -> save();
23         return response()->json('suscriptor agregado');
24     }
25
26     public function show($id_suscriptor)
27     {
28         $suscriptor = suscriptor::find($id_suscriptor);
29         return response()->json($suscriptor);
30     }
31
32     public function update(Request $request, $id_suscriptor)
33     {
34         $suscriptor = suscriptor::find($id_suscriptor);
35         $suscriptor -> update($request->all());
36         return response()->json('suscriptor modificado');
37     }
38
39     public function destroy($id_suscriptor)
40     {
41         $suscriptor = suscriptor::find($id_suscriptor);
42         $suscriptor -> delete();
43         return response()->json('suscriptor borrado');
44     }
45 }
46
```

Figura 3.36: Controlador de tabla suscriptor [Elaboración propia]

A continuación, se procederá a explicar cada una de las funciones dentro del **controlador**. Se debe considerar que todos los **controladores** trabajan de manera similar, sin embargo, algunos tienen mayor cantidad de datos, dependiendo la tabla a la que quieran hacer referencia.

- **public function index:** esta función permite acceder a todos los datos de la tabla que maneja nuestro **controlador**.
- **public function store:** esta función permite almacenar datos dentro de la tabla que maneja nuestro **controlador** a través de la función **save()**. Dentro de esta función se deben agregar todos los campos que interactúan dentro de nuestro **modelo**.
- **public function show:** esta función permite buscar y mostrar un elemento seleccionado a través de su **Primary Key**, en este caso a través de la variable **\$id_suscriptor**.
- **public function update:** esta función permite buscar y editar un elemento seleccionado a través de su **Primary Key**, en este caso a través de la variable **\$id_suscriptor**, a través de la función **update()**.
- **public function destroy:** esta función permite buscar y eliminar un elemento seleccionado a través de su **Primary Key**, en este caso a través de la variable **\$id_suscriptor**, a través de la función **delete()**.

4.14.4.8. Realizar pruebas de control a base de datos

Al momento de realizar las consultas dentro de la base de datos, se debe considerar que se debe realizar un control de que estas estén realizando las consultas de manera correcta. Dentro de SIAP se está trabajando actualmente para mejorar las consultas a la base de datos y evitar problemas de usuarios indeseados dentro de áreas administrativas o que existan datos duplicados. Sin embargo, dentro de las primeras pruebas, el *Team* logró identificar que el **MVC** (Modelo Vista Controlador) fue implementado de manera correcta. A continuación, se presenta una inserción de datos a la base de datos a través de SIAP.

Figura 3.37: Ejemplo de inserción de datos a tabla usuario [Elaboración propia]

rut_usuario	contrasena	nombre	apellido_pat	apellido_mat	fecha_nacimiento	correo	telefono	ciudad	motivo_consulta
18.779.181-2	contraseña123456	Luis Joaquín	Caroca	Araya	02-Apr-1994	luis.caroca@ucm.alu.cl	+96595712049	Linares	Motivo test

Figura 3.38: Datos ya ingresados en gestor de base de datos [Elaboración propia]

En la figura 3.37 se puede apreciar que dentro de la vista de registro se están ingresando los datos para hacer un registro a la base de datos, mientras que en la figura 3.38 se puede apreciar que los datos fueron recibidos por el gestor de la base de datos.

4.14.4.9. Redactar documento de memoria

El *Team* da por casi finalizado el documento de memoria de título. Ya consideran que el documento está lo suficientemente completo para su presentación, en donde se abarca por completo todas las tareas realizadas y funciones incorporadas en SIAP, sin embargo, muchas muestras, como por ejemplo las distintas pruebas a la base de datos, **modelos** o **controladores** no fueron mostrados para no sobrecargar el documento con información repetitiva y solo se utilizó 1 ejemplo para cada caso, intentando ser lo más explicativos posibles para no dejar dudas abiertas.



UNIVERSIDAD CATÓLICA DEL MAULE

Facultad de Ciencias de la Ingeniería

Profesor Guía

Escuela de Ingeniería Civil Informática

Dr. Marco Toranzo

SIAP: SISTEMA DE AGENDA E INFORMACIÓN PARA PSICÓLOGA

LUIS JOAQUÍN CAROCA ARAYA

DAVID ALFREDO GONZÁLEZ ELGUETA

Memoria para optar al

Título Profesional de Ingeniero Ejecución en Computación e Informática

Talca, Octubre 2022

Figura 3.39: Portada final de documento Memoria de Título [Elaboración propia]

4.14.4.10. Realizar correcciones a documento de memoria

El *Team* vuelve a dar una revisión completa al documento, con intención de añadir imágenes que pudiesen faltar dentro del documento, corrigiendo posibles faltas de ortografías

presentes, además de comprobar si la redacción era la adecuada. Al ser solamente una tarea de revisión, no existe registro visual de esta *issue*.

Capítulo 5

Conclusiones y Trabajos

Futuros

Este capítulo presenta las conclusiones y trabajos futuros recomendados para el uso y mejora del sistema.

5.1. Conclusiones

Los objetivos fueron satisfechos durante toda la implementación de “SIAP” de la siguiente manera.

- **Especificar los requerimientos necesarios para el correcto funcionamiento de la solución:** Este objetivo fue cumplido en la primera etapa del proyecto, a través de conversaciones y una posterior toma de requerimientos en conjunto con la cliente, en donde se especificó todo lo necesario que debía incorporar SIAP para que este fuese aprobado por parte de ella.
- **Utilizar el Marco de Trabajo Scrum:** Este objetivo fue cumplido durante todo el trabajo de SIAP, en donde se especificaron todas las tareas en el *Product Backlog*, con la intención de que todo el *Team* tuviese conocimiento sobre las tareas que se debían realizar en SIAP. Posteriormente fueron cumpliéndose los distintos *Sprints*, en donde se fueron trabajando las distintas tareas en manera de *issue*, logrando completar el desarrollo de manera correcta.
- **Integrar tecnologías Frontend y Backend para el desarrollo de la plataforma:** Este objetivo fue cumplido al momento del desarrollo de SIAP, en donde de tecnología *Frontend* se trabajó con **Vue.js**, mientras que la tecnología *Backend* a elección fue **Laravel**, debido al conocimiento que se tenía de la compatibilidad de ambas tecnologías.
- **Validación del sistema:** Este objetivo fue cumplido al momento de presentar SIAP a la cliente, en donde ella aprueba que el trabajo realizado fue el esperado y que desea seguir trabajando SIAP u otras aplicaciones con el *Team* a futuro, ya sea incorporando nuevas funciones a SIAP o agregando módulos externos que puedan serle de utilidad.

5.2. Trabajos Futuros

En lo que sigue, se presentan algunas ideas de trabajos futuros derivados del desarrollo de este sistema y de conversaciones que se han mantenido con el cliente sobre agregados que podría tener el sistema a futuro:

- Desarrollar una versión móvil, con la intención de que, en caso de no poseer un computador, se pueda acceder sin problemas a SIAP a través de dispositivos móviles.
- Incluir dentro del mismo sitio la posibilidad de almacenar las fichas clínicas de los pacientes, que en la actualidad se guardan solamente en un software de escritorio que posee el cliente.
- Debido a que este es un sistema que podríamos llamar genérico dentro del mundo de la medicina, especialmente telemedicina, se puede modificar y presentar a otros profesionales de la salud, no necesariamente de la misma área que el cliente actual.
- Mejorar el entorno visual de SIAP es un trabajo que se presenta como el más próximo a realizar, esto debido a que la cliente desea contratar a un diseñador gráfico para que le realice el diseño. Además, se debe destacar que durante el proceso de desarrollo de SIAP, esta misma decidió realizar un cambio de imagen a su marca, por lo cual SIAP actualmente no responde a los colores institucionales.
- Investigar e implementar una relación entre la agenda generada dentro de SIAP con Google Calendar (Calendario de Google), con la intención de que dicha agenda se vea reflejada dentro del calendario de Google. Esto permitiría mayor automatización e incorporar recordatorios a través de programación de Google.

5.3. Referencias

[LGS2017] Schwaber, K. (2017). La guía definitiva de Scrum. Recuperado 16 octubre, 2019, de <https://www.scrumguides.org/docs/scrumguide/v2017/2017>.

[PSF2019] Kerkeb MOHAMED Larbi, "*A Comparative study of PHP frameworks performance.*" Recuperado 12 mayo 2022 de https://www.researchgate.net/publication/332596809_A_Comparative_study_of_PHP_frameworks_performance

[LAR2022] Laravel. "*Laravel – The PHP Framework for Web Artisans*", Recuperado 17 mayo 2022, de <https://laravel.com>.

[FIP2010] Llanos, Diego. "*Fundamentos de informática y programación en C*", 2010.

[MDB2022] MariaDB Foundation. "*About MariDB Server*": Recuperado 23 mayo, 2022, de <https://mariadb.org/about/>.

[MPD2016] Valdivia, José. "*Modelo de procesos para el desarrollo del front-end de aplicaciones web*", 2016.

[VJS2022] Vue.js. "*Introducción – Vue.js*". Recuperado 20 mayo 2022, de <https://es.vuejs.org/v2/guide/>.

[JRA2022] Jira. "*Jira | Software de seguimiento de proyectos e incidencias*", Recuperado de <https://www.atlassian.com/es/software/jira>, 2022.

[APR2016] "*Qué es y cuáles son sus ventajas en el desarrollo de proyectos*", Recuperado de <https://bbvaopen4u.com/es/actualidad>, 2016.

[DIS2019] "*Fundamentos JavaScript: Introducción a las API REST*", Recuperado de <http://developinginspanish.com/2019/11/09/fundamentos-javascript-introduccion-a-las-api-rest/>

[LCH2022] Lucidchart "*Lucidchart*", Recuperado de <https://lucidchart.com/>, 2022.

[RAM2019] Raml "*Acerca de Raml*", Recuperado de <https://raml.org/>, 2019.

[SLA2019] Slate "*Acerca de Slate*", Recuperado de <https://github.com/slatedocs/slate>, 2019.

[SWG2019] Swagger “*Acerca de Swagger.io*”, Recuperado de <https://swagger.io/solutions/api-design/>, 2019.

5.4. Anexos

Anexo A: Documento [Manual de Usuario](#).

Anexo B: Documento [Manual de Desarrollador](#).

5.5. Anexos Digitales

Anexo A: Documentación del Software desarrollado, Enlace: <https://github.com/Luiscaroca/documentacionSIAP>.

Anexo B: Historial de desarrollo en la herramienta Jira, Enlace: <https://luiscarocaucm.atlassian.net/jira/software/projects/SIAP/boards/1/backlog>.