

DESARROLLO DE SOFTWARE SOBRE LA GESTIÓN DE CASOS EN EL DERECHO CIVIL

Autor(es):

MARUAN ENRIQUE ARIAS CUCUNUBÁ

ANTONIO JOSÉ MATA RODRÍGUEZ

**UNIVERSIDAD EL BOSQUE
PROGRAMA DE INGENIERÍA DE SISTEMAS
FACULTAD DE INGENIERÍA
Bogotá, 2023**

DESARROLLO DE SOFTWARE SOBRE LA GESTIÓN DE CASOS EN EL DERECHO CIVIL

Autor(es):

**MARUAN ENRIQUE ARIAS CUCUNUBÁ
ANTONIO JOSÉ MATA RODRÍGUEZ**

**Trabajo de Grado presentado como requisito para optar al título de
INGENIERO DE SISTEMAS**

**Modalidad de Grado:
DESARROLLO TECNOLÓGICO**

Director(a)

CARLOS IGNACIO DELGADO ROMÁN

*Ingeniero de Sistemas, Especialista en Gerencia de Proyectos, Especialista en Gerencia Financiera,
Magíster en Administración*

**UNIVERSIDAD EL BOSQUE
PROGRAMA DE INGENIERÍA DE SISTEMAS
FACULTAD DE INGENIERÍA
Bogotá, 2023**

DEDICATORIA

Le dedico este proyecto a mi madre y a mi padre, que desde muy pequeño me enseñaron el valor del esfuerzo y la dedicación constante, además, de siempre hacer las cosas por motivación y no por obligación.
~ Antonio Mata

Dedico este proyecto a mi familia y en especial a mi abuela Mary por todo el apoyo brindado a lo largo de la carrera de Ingeniería y darme las bases de lo que soy como persona.
~ Maruan Arias

AGRADECIMIENTOS

Se da agradecimientos al señor Hernando Velez por su gran colaboración para el desarrollo del software para el Derecho Civil, aprendimos mucho de esa área de trabajo, su contexto y dificultades que se presentan actualmente.

~ Antonio Mata & Maruan Arias

Le agradezco a mi compañero de viaje Maruan, él no solo me acompañó en el desarrollo de este proyecto de grado sino en otra infinidad de trabajos de la carrera, gracias por tu compañía y todas las enseñanzas que me brindaste. No solo estuvo Maruan, sino también mis compañeros John Alejandro y Fabián Andrés, los cuales me brindaron su conocimiento y apoyo para superar múltiples obstáculos. También a mi mejor amigo Álvaro que es mi constante inspiración para perfeccionar mis habilidades como Ingeniero y ser un mejor profesional. Por último deseo agradecer infinitamente a Alejandra, la cual fue una gran compañía durante el desarrollo de este proyecto, sin su amor y apoyo incondicional, no alcanzaría la meta final.

~ Antonio Mata

Tabla de contenido

1. Introducción	4
2. Descripción Contexto y Justificación del Problema Desde el Modelo Biopsicosocial y Cultural	5
3. Marco Referencial	9
3.1. Antecedentes y Estado Del Arte	9
3.2. Marco Teórico	11
4. Descripción de la Solución Desde el Modelo Biopsicosocial y Cultural	12
4.1. Descripción del Artefacto	13
4.2. Componente de Análisis: Descripción de la Transformación Esperada del Contexto	14
5. Diseño Metodológico	15
6. Resultados y Discusión	18
7. Conclusiones	28
8. Lecciones Aprendidas y Trabajo Futuro	29
9. Referencias	29
Anexos	30
A. Anexo No. 1: Diagrama del Modelo Biopsicosocial desde la perspectiva del Problema	30
C. Anexo No. 3: Modelo BPMN AS - IS del Grupo Montserrat	30
D. Anexo No. 4: Modelo BPMN TO - BE del Grupo Montserrat	30
E. Anexo No. 5: Tablero de Control	30
F. Anexo No. 6: Metodología	30
G. Anexo No. 7: Diagrama C4	30
H. Anexo No. 8: Diagrama Arquitectura AWS	30
I. Anexo No. 9: Tabla costos - Productos AWS	30

DESARROLLO DE SOFTWARE SOBRE LA GESTIÓN DE CASOS EN EL DERECHO CIVIL *SOFTWARE DEVELOPMENT ON CASE MANAGEMENT IN CIVIL LAW*

Antonio J. Mata y Maruan E. Arias
amata@unbosque.edu.co, meariasc@unbosque.edu.co
Universidad El Bosque, Colombia

Resumen— La presente investigación muestra la detallada elaboración de un software de Gestión de Casos en el Derecho Civil que facilitará el manejo de la información de los casos en la empresa Grupo Monserrat. Dicho software fue desarrollado a partir de un híbrido de las metodologías SCRUM y Cascada, que, mediante una serie de Sprints, permitieron avanzar teniendo en cuenta las validaciones efectuadas en cada paso del desarrollo.

Lo anterior, fue contrastado mediante el uso de pruebas UEQ KPI, las cuales dieron lugar a la medición de la variable de interés para demostrar el antes y después de la implementación del mismo. Estas demostraron que, mediante el Software propuesto, se brinda una óptima experiencia al usuario, en la cual se puede mostrar y gestionar de forma sencilla y práctica los datos correspondientes a los casos a tratar por la compañía. Del mismo modo, los resultados obtenidos dan cuenta de cómo el uso de estas tecnologías facilitan la gestión de información y es completamente amigable con el usuario, esto en comparación al uso común de herramientas que no cuentan con pruebas de experiencia de usuario.

Abstract— This research shows the detailed development of a software for Case Management in Civil Law that will facilitate the management of case information in the company Grupo Monserrat. This software was developed from a hybrid of SCRUM and Waterfall methodologies, which, through a series of Sprints, allowed progress taking into account the validations made at each step of development.

The above, was contrasted through the use of UEQ KPI tests, which resulted in the measurement of the variable of interest to demonstrate the before and after the implementation of the same. These showed that, by means of the proposed software, an optimal user experience is provided, in which the data corresponding to the cases to be treated by the company can be displayed and managed in a simple and practical way. Likewise, the results obtained show how the use of these technologies facilitates the management of information and is completely user-friendly, compared to the typical use of tools that do not have user experience tests.

Palabras Clave— Engineering, Law, Software, User interfaces, Web technologies

◆

1. Introducción

La gestión de cobranzas en la parte civil y comercial del Derecho suele ser abrumadora, debido a que, contiene múltiples archivos y actualizaciones que pueden hacer difícil el retomar un caso o generar complicaciones al momento de delegarlo a otro abogado, ya que, toma tiempo contextualizarse si no se tiene una buena organización en el proceso y en los documentos involucrados en el mismo. Es por ello, que con el Grupo Montserrat, se presentó la propuesta de desarrollar un software intuitivo que permita la sistematización del proceso de cobros de los clientes, además de incluir la organización de documentos, filtrado y generación de informes.

El desarrollo de esta solución tecnológica es importante, debido a que, el cliente de Grupo Montserrat, ha manifestado que han intentado utilizar otras soluciones ofrecidas por diversas empresas dedicadas al Desarrollo de Software pero lastimosamente no se ha logrado dar algún resultado de las mismas, a la vez, se ha perdido bastante capital por el incumplimiento de la solución, además, la organización de archivos de los casos en la nube de Google Drive, genera que los documentos tengan la libertad de ser organizados por cualquier empleado de Grupo Montserrat, lo cual genera pérdidas de tiempo en buscar y dar seguimiento a los casos. Por todas las problemáticas anteriormente mencionadas, se decidió implementar una aplicación basada en la web que le facilite a los trabajadores del Grupo Montserrat, lo cual es una manera más sencilla de poder almacenar y dar seguimiento a los casos.

Al realizar una indagación a nivel internacional, nacional y local, se pudo encontrar varios casos en el ámbito jurídico, como por ejemplo Córdova [3] que en Lima - Perú, implementó un Web App en un Bufete de Abogados que ayudó a la organización de información en la cual puede ser accedida con bastante facilidad, a la vez que Novoa y Calderón [7], en Girardot - Colombia, donde el dueño de la empresa Orozco Ocampo Abogados reportaba que no podía dar seguimiento a los casos de los abogados y necesitaba generar un reporte del trabajo realizado, por lo cual se decidió desarrollar un Aplicativo de Escritorio, en cual se reportó un aumento de la productividad en un 80%.

Por último se encontró que en Bogotá - Colombia, Chacón et al [5], identificó en la firma de abogados Norton Rose Fulbright, que varios procesos realizados dentro de la empresa se realizaban de manera manual y el paso de esta información a otras áreas genera mala interpretación lo cual retrasa la evolución de los casos, se mejoró el software para los seguimientos de los casos, como resultado generó un incremento en la producción y precisión de la información.

Todos los problemas identificados en los anteriores casos se ajustan a los problemas identificados en el Grupo Montserrat, y que al implementar un aplicativo que apoye al organizamiento de los casos, visualizaron un incremento en la productividad y mejor seguimiento a los casos llevados a cabo por varios empleados de la firma. Un valor agregado que se quiere dar al Grupo Montserrat, que no ofrecen las soluciones encontradas, es el poder filtrar la información, generar y exportar informes para encontrar de manera más eficiente la información que requiera el trabajador y visualizar de una mejor manera el trabajo realizado.

El presente documento posee la siguiente estructuración, en la sección de *Descripción Contexto y Justificación del Problema Desde el Modelo Biopsicosocial y Cultural*, se puede encontrar el análisis realizado de los problemas identificados de la empresa utilizando el modelo biopsicosocial ofrecido por la Universidad El Bosque, junto con el objetivo general y los específicos, en la sección del *Marco Referencial* se encuentran los trabajos realizados en diferentes ubicaciones, en el cual se visualiza los trabajos implementados y los resultados que brindaron los mismos, además de sustentar las decisiones tomadas para brindar la mejor solución basándose en los casos encontrados en los *Antecedentes*.

Por otro lado, en la sección del *Diseño Metodológico*, se encuentra la metodología utilizada para la realización del presente trabajo, en los *Resultados y Discusión* se encuentran los resultados obtenidos al momento de desarrollar e implementar la solución y se discute sobre el impacto de los mismos en el proyecto y la empresa, en la sección de *Conclusiones*, se encuentra el cierre del proyecto donde se le da un enfoque el alcance de los objetivos planteados, en la sección de *Lecciones Aprendidas y Trabajo Futuro*, se realiza una serie de aprendizajes adquiridos a lo largo de la elaboración del proyecto, además, de brindar un panorama del trabajo futuro en el cual se puede continuar la solución planteada, en la sección de *Referencias*, se encuentran una serie de recursos de bibliografía digital la cual fue usada para sustentar algunas afirmaciones realizadas en el documento, por último, en la sección de *Anexos*, se encuentra el documento de anexos que aloja una serie de recursos visuales que apoyan algunas secciones, además de una explicación de los mismos.

2. Descripción Contexto y Justificación del Problema Desde el Modelo Biopsicosocial y Cultural

La organización Grupo Montserrat es una empresa de la ciudad de Bogotá especializada en resolver casos enmarcados dentro del Derecho Civil. Ha trabajado en esta área del derecho durante 40 años y actualmente se especializa en casos de recuperación de cartera los cuales se rigen bajo el derecho privado. Al trabajar bajo el derecho privado, las normas por las cuales se rigen se denominan inter partes, por lo que, como indica su nombre, solo afectan a las partes involucradas. Este tipo de derecho ha generado ciertos hábitos en la dinámica del Grupo Montserrat, ya que, deben comunicarse tanto con el cliente como con el deudor y estar atentos a los veredictos que dicten los juzgados para ir avanzando en el caso durante las distintas etapas.

Las etapas que existen en el proceso de recuperación de cartera se resumen en dos principales, la primera de ellas es la etapa prejudicial que es el primer acercamiento con el deudor, donde se busca llegar a un acuerdo de pago con el deudor o deudores sin la necesidad de una demanda; sin embargo, en el caso que no se pueda concretar un acuerdo de pago se procede a la segunda etapa, que es la judicial. La etapa judicial es cuando se inicia un proceso de demanda a partir de un título aportado por el cliente; este proceso judicial sufrió varios cambios a partir de la pandemia del COVID - 19, dado que, antes de la pandemia los procesos y documentos se llevaban de forma personal; lo anterior, implicaba ir a los juzgados de manera presencial y tener de

manera física todos los documentos que se requieran para el caso.

Por lo tanto, la pandemia cambió estos procesos forzando a la rama judicial a llevarlos a la virtualidad. Actualmente, las reuniones en los juzgados se realizan de manera virtual, por lo que todos los documentos deben digitalizarse, ya sea para subirlos a la página de la rama judicial o para enviarlos a una determinada persona para una revisión o aprobación, según sea el caso. Esto obligó a Grupo Monserrat a buscar herramientas de software para la gestión de dicha información; por consiguiente, tomaron la decisión de usar Google Drive, una herramienta con virtudes y limitantes que genera una serie de dificultades, analizadas más adelante. Por último, existen momentos dentro del caso denominados *extraprocesales*, los cuales hacen referencia a cuando uno de los deudores paga las obligaciones que son solidarias, por lo tanto, si uno de los deudores realiza el pago de las obligaciones, la deuda queda saldada.

Como se mencionó antes, el proceso que maneja Grupo Monserrat tiene dos etapas principales, la prejudicial y la judicial. En ambas etapas está presente la intervención de la herramienta de Google Drive para la gestión de los casos de los deudores, el uso de la misma es transversal a todo el proceso, ya que, en diferentes momentos se requiere la consulta de documentos (ver Anexo 3 - Figura No. 4). Por lo tanto, los procesos a intervenir son los siguientes: creación de la carpeta del deudor en Drive, notificación deudor, acuerdo de pago, presentación de la demanda y subsanar demanda.

Teniendo en cuenta lo anterior, la mayor necesidad para la empresa es el manejo adecuado y óptimo de uno de sus recursos más importantes, la información, dado que, a raíz de los diversos cambios en la rama judicial actual, se vieron obligados a digitalizar toda su documentación; es en este proceso de transformación tan drástico que se optó por herramientas como Google Drive que facilitarían la gestión de los documentos digitalizados. Esta es una buena herramienta, sin embargo, trajo consigo una serie de inconvenientes en la gestión de los documentos, lo que conlleva, a un manejo tedioso y desgastante de los casos en sus distintas etapas y a recurrir a procesos manuales que, en realidad, se pueden automatizar.

Es relevante resaltar que, dicho desgaste en la gestión documental provoca que los procesos a intervenir se ralenticen; de por sí, la naturaleza de los procedimientos que se llevan a cabo en la Rama Judicial son lentos, por lo que hay actividades que no se pueden agilizar, sobre todo, las que conllevan la decisión de jueces y otros actores. Por consiguiente, el manejo tedioso de la información digital que lleva Grupo Monserrat se da a partir de factores multicausales, los cuales se pueden visualizar en el árbol de problemas (ver figura 1).

Como se puede observar, existen causas biológicas externas, como lo fue la pandemia del COVID - 19, que obligó a la Rama Judicial a cambiar su funcionamiento, virtualizando sus procesos y, así mismo, a todas las empresas que trabajan de forma conjunta; entre dichas compañías está Grupo Monserrat, quienes presentaban resistencia al cambio causando que no hubiera una inversión en proyectos de tecnología y que se promoviera un desconocimiento en temas de Gestión de

Documentos, dos motivos que provocaron el uso de Google Drive como herramienta para la gestión de documentos; recurso que ha generado tendencias engorrosas en el manejo de la información de la firma y que es la primera causa de sus falencias documentales.

La segunda causa, derivada igualmente por los cambios implementados en la Rama Judicial, es su página web, en la cual se consultan, actualizan y descargan documentos de los casos manejados por la organización; dicha página presenta un bajo rendimiento en términos de accesibilidad en los horarios laborales, dificultad que aumenta el tiempo en sus procesos. Esto genera efectos negativos, tales como la demora en la toma de decisiones, lo que a su vez enlentece la ejecución de procesos y genera un uso prolongado de dispositivos tecnológicos. Lo anterior, causa entorpecimiento en la explicación del orden de las carpetas a personas externas e internas de la organización, y así, se produce estrés y deterioro del equipo de trabajo.

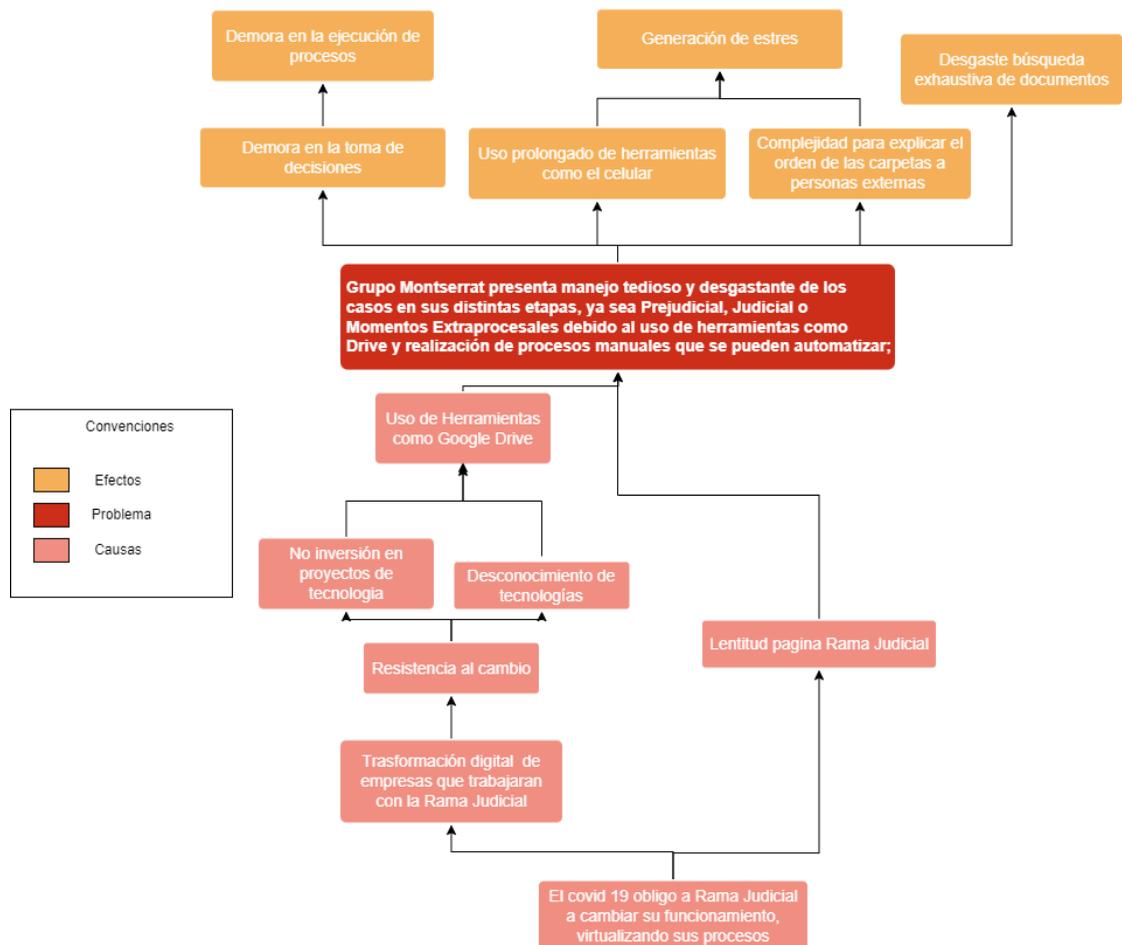


Figura No. 1 - Árbol de problemas
Fuente - Elaboración Propia

El análisis del modelo Biopsicosocial y cultural de la problemática surge desde la perspectiva de dos actores, está el primer actor, que son los trabajadores de la empresa de Grupo Monserrat, y están los deudores, aquellos a quienes se les hace el proceso de cobro de obligaciones. El medio para el personal de la firma se encuentra en la ciudad de Bogotá D.C, en una empresa que debe trabajar de la mano con la Rama Judicial; en el caso del deudor, el medio es la ciudad de Bogotá.

Adicionalmente, los trabajadores del Grupo Montserrat, bajo el medio ya descrito, usan una serie de artefactos para el desarrollo de sus actividades, dentro de los cuales se encuentra Drive, el celular, correo electrónico y la página de la Rama Judicial. En el caso del deudor se limita a celular y correo electrónico.

En cuanto a los componentes de creencias y hábitos, para el actor de Grupo Monserrat se encontraron una variedad de relaciones (ver Figura No. 2). Dentro de dichas relaciones hay cuatro principales; la primera es la creencia de que la implementación de la tecnología es tediosa y costosa, concepto generado a partir de intentos pasados que no tuvieron éxito; además, están las pérdidas de recursos económicos y de tiempo, las cuales generaron el hábito de la no exploración de nuevas tecnologías, por lo que, a su vez, existe una disminución considerable de la competitividad en el mercado frente a empresas del mismo rubro que están explorando tecnologías para la mejora de su proceso de negocio.

La segunda relación parte de que el artefacto, es decir, la página de la Rama Judicial, es lenta y tiene tendencia a caerse en horario laboral, esto provoca frustración, aburrimiento e irritación a los usuarios, además genera el hábito de espera, abandono y refresque continuo de la página lo que puede provocar impaciencia.

Por otro lado, la tercera relación surge a partir de la herramienta Google Drive, donde se generó la creencia de que el orden de las carpetas en dicha web solo lo entiende la persona que las creó, por lo que se da la sensación de sobrecarga de información, retrasando la toma de decisiones y ejecución de sus tareas.

Finalmente, la cuarta y última relación, se crea a partir del hábito de la revisión constante de notificaciones del dispositivo celular, esto puede crear adicciones como la de estar conectados todo el tiempo; del mismo modo, la creencia de que el trabajador debe estar atento al celular para notificar al deudor del pago de sus obligaciones provoca un cambio en la forma de relacionamiento cliente-deudor y puede generar estrés.

En concordancia, se debe realizar el análisis de las relaciones de creencias y hábitos pero desde la perspectiva del deudor (ver Anexo 1 - Figura No. 2).

Existen tres relaciones en total; primero está la creencia de que las obligaciones financieras no son su responsabilidad, lo que genera el hábito de evadir cualquier situación de cobro provocando fatiga y estrés mental, adicional al aplazamiento de las responsabilidades. Segundo, aparece la creencia de que solicitar ayuda para

pagar la deuda es signo de debilidad, desencadenando el hábito de nunca hablar sobre las obligaciones que se tienen por lo que habría un aplazamiento de las responsabilidades y se desarrolla un complejo de inferioridad frente a sus conocidos.

Finalmente, está la relación que se genera a partir del hábito de realizar exclusivamente los pagos mínimos, esto genera la creencia de que nunca podrá pagar la deuda, propiciando altos niveles de estrés e insomnio.

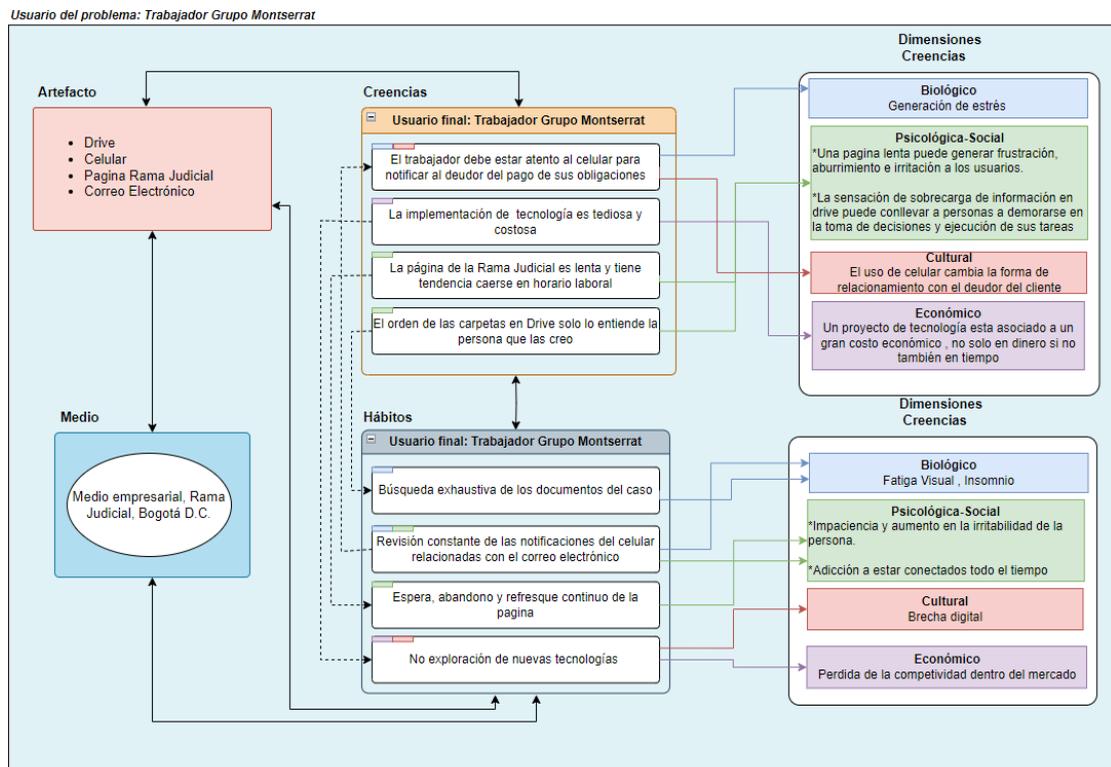


Figura No. 2 - Modelo Biopsicosocial Usuario del problema- Trabajador Grupo Montserrat
Fuente - Elaboración Propia

3. Marco Referencial

A continuación, se retomarán trabajos realizados con anterioridad a nivel local, nacional e internacional, apartado conocido como *Antecedentes y Estado del Arte*; asimismo, se tiene el *Marco Teórico*, donde se relacionan resultados obtenidos desde el *Estado del Arte* que sirven como apoyo a las sustentaciones realizadas en esta sección.

3.1. Antecedentes y Estado Del Arte

1) Tecnologías

En la investigación realizada por Pérez [2] es posible evidenciar como un Estudio Jurídico situado en Tarapoto - Perú presenta dificultades para atender de forma óptima y eficiente los casos recibidos por su personal; además de que genera situaciones problemáticas para los clientes al momento de buscar información relacionada al estado de sus procesos legales. Por ello, se decide desarrollar e implementar un sistema que permitiera gestionar de forma adecuada los procesos judiciales manejados por dicho Estudio Jurídico y que brindara mayor facilidad a sus clientes para mantenerse al tanto de sus procesos legales; de esta manera, se da uso de la metodología Extreme Programming durante cuatro meses y desarrollado en el lenguaje de programación PHP. Posterior a dicho proceso, fue posible concluir que, a partir de la implementación de dicha tecnología, el personal del Estudio Jurídico logró gestionar de forma más efectiva, rápida y organizada todos los procesos legales; a su vez, se logró que el servicio ofrecido a los clientes sea más ágil y eficiente.

En la ciudad de Raseborg - Finlandia, Rahikainen [9] desarrolló una web-app la cual tiene como objetivo poder visualizar vídeos de YouTube de manera simultánea con otro usuario, donde pueden acceder a todo el contenido disponible en la plataforma de vídeos, a la vez, de que el administrador puede pausar, adelantar, rebobinar un vídeo en tiempo real. Rahikainen utilizó el framework de Angular para realizar el aplicativo web en forma de una Single Page Application (SPA), el cual permite que se ejecute una sola vez toda la página web, y cuando el usuario interactúe con la aplicación, se reescribirá la web-app de manera gradual. El resultado obtenido de hacer una SPA como solución al problema expuesto fue todo un éxito, el autor destaca que aunque falta mejorar algunas cuestiones como la experiencia de usuario, considera que el objetivo planteado en el trabajo fue sobresaliente.

Desde otra perspectiva, en Lahti - Finlandia, Uotila [13] diseñó un prototipo de una aplicación, la cual tiene como finalidad servir datos para una aplicación de redes sociales basada en "feeds", donde se enfoca en dispositivos móviles. Los objetivos planteados fueron: incorporar autenticación, almacenamiento de fotos, lecturas y escrituras lo suficientemente efectivas de los datos de alimentación, así como conservar la modularidad para futuras escalas y pruebas. Uotila implementó la solución utilizando los servicios de Amazon Web Services (AWS) donde utilizó la tecnología de Lambda para construir una arquitectura sin servidor, a la vez, de que se implementó instancias de S3, RDS, EC2 y SQS, en lenguajes de programación se utilizó Python junto con Flask y para la base de datos se utilizó PostgreSQL. El resultado de la implementación de la solución fue un prototipo funcional, que se puede escalar.

Con base a los antecedentes encontrados, se puede tomar como preámbulo para conocer qué soluciones tecnológicas ya existen, así se puede incluir soluciones innovadoras, sin embargo, como lo justificó el cliente, desea un software el cual puedan administrar sus casos, por lo tanto, estos antecedentes nos sirven para aprender qué herramientas y tecnologías son las ideales para construir cada uno de

los componentes que constituyen un software para la gestión de casos.

2) Marco de Trabajo

A nivel nacional, se encuentran Novoa y Calderón [7], ubicados en Girardot - Colombia, quienes emprenden un proceso de investigación para la empresa Orozco Ocampo Abogados, el cual consiste en la implementación de un aplicativo de escritorio para automatizar la generación de informes, reportes de rendimiento y monitoreo de procesos realizados por los asesores. El propósito del desarrollo de software se genera a partir de que el gerente de la empresa detectó serios problemas de productividad por parte de los agentes, además, de que se desconoce el proceso en el que se encuentran los casos que están llevando a cabo, además, de expresar preocupación el rendimiento de sus agentes.

La metodología que llevaron a cabo para el desarrollo del software, fue de tipo evolutivo llamado *prototipado*, el cual se concentra en la fuerte relación entre el cliente y los desarrolladores de la solución; de igual manera, contribuye a ir entregando gradualmente prototipos de la solución, para que, de manera gradual se le otorgue el aval de lo ya desarrollado. Los resultados obtenidos por parte de Novoa y Calderón demuestran que, al momento de implementar la solución, se evidenció una disminución del 80% en los tiempos de ingreso, gestión, generación de informes, entre otros procesos.

Por otro lado, Córdova [3] realiza una investigación en Lima - Perú con la finalidad de implementar una Web App que permita el registro de órdenes de trabajo para un Bufete de Abogados, además busca conseguir que dicha tecnología cumpla el objetivo de calcular la factura de los clientes. El propósito de este estudio se da a partir de las inconsistencias presentadas por el Bufete de Abogados al momento de gestionar la información de los procesos que tienen a cargo, ya que, todo el manejo de datos se daba de forma manual, lo cual daba pie a múltiples errores, mayor tiempo de trabajo y menos eficiencia en el servicio prestado.

La metodología utilizada fue SCRUM, dado que permite agilizar el proceso de implementación y el correcto cumplimiento de los objetivos trazados, se plantean diversas fases de trabajo y se puntualiza en la meta a conseguir. Los resultados obtenidos permitieron observar que la Web App da lugar a un acceso sencillo y rápido a todo aquel que requiera la información de los procesos judiciales manejados por el Bufete, además de que tiene una óptima capacidad de organización y categorización de los datos almacenados para cada proceso; de igual forma, se consigue brindar la posibilidad a los abogados de administrar de manera efectiva el costo de sus tarifas, lo que facilita la elaboración de las facturas para los clientes, llegando así al objetivo final.

Así mismo, en Bogotá D.C - Colombia, se encuentra un emprendimiento para mejorar la arquitectura de software llevada a cabo por Chacón et al [5], con el cual se desean mejorar los procesos de negocio dentro de la firma de abogados Norton Rose Fulbright. La finalidad de esta investigación se debe a que, se detectaron severos

problemas dentro del negocio, generando así confusión en varios niveles de la empresa, esto se debe en su mayor parte a las inconsistencias de información debido a los sistemas internos y externos que presenta la organización, además, de que muchas gestiones que se realizan en algunas áreas, se elaboran de manera manual.

Para solucionar los problemas de procesos de la empresa, optaron por implementar una metodología BPM, dando así la facilidad de ordenar y ejecutar los procesos existentes de la misma. Para el tema de la construcción de software se implementó la metodología SCRUM, en el cual desarrollaron un sprint planning donde semanalmente ajustaban las tareas de cada integrante, a la vez de organizar reuniones bisemanales para revisar avances del equipo y manifestar dificultades presentadas por el equipo durante el desarrollo de sus actividades, por último, realizaron reuniones semanales con el cliente, para discutir si los progresos realizados se ajustan a la meta pactada con el cliente. Los resultados obtenidos fueron el incremento de la productividad y, a la vez, de la minimización de errores debido al trabajo manual gracias a la automatización implementada.

Teniendo en cuenta las múltiples consultas realizadas en documentos donde se desarrolla una solución a nivel tecnológico para un negocio de índole jurídico, se detectó que no se realiza un estudio de satisfacción mediante los parámetros de UEQ KPI, además, no se realiza un estudio sobre las decisiones tomadas al momento de realizar la maquetación del frontend, dejando de lado la experiencia del usuario y la teoría que conlleva el UX/UI. A continuación se explicará qué son los UEQ KPI.

Según Hinderks et al [1] en un negocio, para medir el rendimiento de varios departamentos, resulta más sencillo cifras clave que permitan entender cómo sus labores contribuyen al negocio, para ello se tienen los KPI, el cual significa Key Performance Indicator. El KPI va ligado a la experiencia de usuario, conocido comúnmente con las siglas de UX, consiste en la experiencia que tiene el usuario al momento de interactuar con un aplicativo, no es posible definir un estándar para concluir cuando un aplicativo cumple con todas las características de ser cien por ciento apto para cualquier usuario, debido a que, muchas empresas toman diferentes variables al momento de evaluar un aplicativo desde el rol del usuario, es por ello, que se utilizan las UEQ.

Las UEQ o por sus siglas, User Experience Questionnaire, tiene la finalidad de que los usuarios realicen una evaluación que cubra sus impresiones al momento de utilizar un software, según Hinderks, la prueba contiene 26 ítems que se agrupan en 6 escalas (Atractivo, Visibilidad, Eficiencia, Confiabilidad, Estimulación y Novedad), cada escala evalúa diferentes áreas del UX.

Finalmente, en Milagros - Ecuador, se encontró que Jimenez y Idrovo [12] utilizaron la metodología Cascada, de la mano de la metodología ágil XP Programming, para realizar un aplicativo web con la finalidad de automatizar el procesos relacionados a la odontología. Lo que realizó Jimenez y Idrovo, fue implementar esta metodología de acuerdo al cronograma que establecieron, con el fin de generar en cada paso que propone la metodología un resultado específico, en el cual podrá realizar las

respectivas pruebas para dar solución a errores que se puedan generar, una vez identificados, se procederá a corregir. El resultado obtenido al implementar la metodología Cascada fue entregar un aplicativo web que mejoró varios procedimientos que se realizaban de manera manual.

3.2. Marco Teórico

Las empresas relacionadas al ámbito jurídico, prefieren utilizar un aplicativo basado en la web para realizar sus tareas diarias; por eso, en algunos bufetes de abogados, suelen utilizar archivadores para almacenar los documentos de sus clientes, por lo que suele ser engorroso debido a que se tiene que organizar de forma regular para evitar que la información se pierda, además de asegurarse que solo las personas autorizadas puedan acceder a dichos documentos, por ello, para facilitar el trabajo e inclusive aumentar la productividad, se optan por soluciones tales como los aplicativos web.

Así fue el caso de Córdova [3], que buscaba implementar esta solución con el fin de optimizar el cálculo de las cuentas de sus clientes; como resultado se obtuvo que la productividad en la empresa aumentó debido a que los abogados podían acceder a la información de los clientes de una manera más eficiente, además, de optimizar el tema de pagos al poder cobrar de una manera más rápida y exacta por los trabajos realizados por sus empleados. En conclusión, es ideal la implementación de un aplicativo web, para agilizar el trabajo de la empresa Grupo Montserrat.

SCRUM y Cascada son dos metodologías que combinadas brindan una comunicación directa con el cliente al ofrecer un producto mínimo viable el cual puede ser medible por medio de la herramienta UEQ KPI. Para realizar el cuestionario UEQ y medir sus resultados KPI, es necesario que el usuario interactúe con el producto, tal como lo explican Hinderks et al [1], pero para alcanzar paulatinamente el objetivo planteado de brindar un producto que sea totalmente intuitivo es necesario utilizar SCRUM para que en cada iteración de trabajo se le brinde al cliente un producto mínimo viable el cual se obtenga el resultado de las pruebas y en la siguiente iteración se corrijan los resultados bajos.

Por otro lado, Cascada, como lo explica Jimenez y Idrovo [12] brinda un paso a paso para tener un camino claro de desarrollo de software, la desventaja es que no ofrece una comunicación directa con el cliente, es por ello que se realizó la combinación con la metodología de SCRUM. Para concluir lo anterior, las metodologías SCRUM y Cascada son fundamentales para tener una interacción directa con el cliente para alcanzar la necesidad de tener un software totalmente intuitivo y fácil de utilizar.

El framework Angular es una herramienta ideal para la implementación de Single Page Application (SPA), ya que permite realizar las SPA, el cual ofrece que el contenido esté disponible cuando el usuario lo requiera, lo cual hace que el aplicativo sea más eficiente y que se eviten los retrasos al momento de cargar todo el software como hacen otros Frameworks. Lo anterior se puede evidenciar gracias al trabajo realizado por Rahikainen [9], en el cual implementó Angular al desarrollar un

aplicativo web donde se pueden ver vídeos con otro usuario en tiempo real; por lo tanto, el uso de este framework dio como resultado una respuesta inmediata en el momento en el que el administrador avanzara o retrocediera un vídeo. Teniendo en cuenta esto, el uso de este framework favorecerá a la solución propuesta al cliente, debido a que, podrá obtener la información en tiempo real, como lo son las actualizaciones de los casos, sin tener que refrescar manualmente el navegador, a la vez que, será rápida la interacción con el aplicativo.

El prestador de servicios Amazon Web Services (AWS), ofrece servicios que permiten implementar una arquitectura en la nube basada en microservicios de manera sencilla y escalable. Según lo documentado por Uotila [13], AWS ofrece múltiples servicios de los cuales se pueden ver claramente los costos que incurren al momento de utilizarse, además, se implementaron las Lambdas que permiten ejecutar tareas específicas basadas en código, lo cual ahorra bastante tiempo al desarrollar un Bus de Servicios, sumado a esto, la escalabilidad es fácil de realizar. En conclusión, para el Grupo Montserrat, pasar todas sus actividades realizadas en Google Drive a los servicios en la nube, representados en un aplicativo web, les ahorrará en tiempo y costos; además, de que si se desea añadir una funcionalidad a la solución, será fácil realizar el escalamiento e implementación.

4. Descripción de la Solución Desde el Modelo Biopsicosocial y Cultural

En esta capítulo se nombran los objetivos del proyecto y se da explicación y descripción detallada de la solución y del artefacto que representa esa solución; asimismo, se abordará la transformación generada por la solución implementada a partir de una descripción sistémica del modelo Biopsicosocial y cultural que se vio afectada por el artefacto.

Objetivo General:

Desarrollar un sistema de información web bajo criterios de experiencia de usuario que permita la gestión de los casos de recuperación de cartera que maneja la empresa Grupo Montserrat para facilitar el seguimiento de los mismos.

Objetivos Específicos:

1. Analizar el proceso actual de recuperación de cartera de la empresa Grupo Montserrat comprendiendo sus limitantes, generando así historias de usuario que reflejen los requerimientos del sistema de información de gestión de casos.
2. Determinar la arquitectura del sistema de información de gestión de casos de recuperación de cartera definiendo un documento SAD que abarque componentes al igual que sus relaciones y dependencias para soportar la construcción del mismo.
3. Construir el sistema de información de gestión de casos de recuperación de cartera de acuerdo con la arquitectura definida, garantizando así el cumplimiento de los requerimientos definidos en las historias de usuario.

4. Evaluar el sistema de información en las distintas etapas del CTT para generar un documento de pruebas que evidencien la calidad, la aceptación y el cambio del contexto mediante la metodología UEQ KPI.

4.1. Descripción del Artefacto

El artefacto propuesto para resolver la problemática es un sistema de información web que permita administrar los casos de derecho civil de Grupo Montserrat. Este artefacto tendría implicaciones dentro de los hábitos de la empresa e impactos sobre los empleados que intervienen dentro del proceso de recuperación de cartera, debido a que cambiaría la forma en que se comunican con deudores, todo gracias al módulo de mensajería; asimismo, se realizará una transformación dentro de determinados procesos que estén relacionados con la gestión de los casos.

Para visualizar el artefacto en diferentes niveles se toma la decisión de usar un diagrama de Arquitectura C4, este diagrama proporciona 4 niveles para dar perspectiva del artefacto; cada nivel va dirigido a una audiencia específica con el fin de que les sea útil y teniendo en cuenta que el conocimiento técnico entre estos va aumentando. Se realizan dos diagramas, uno de contexto y uno de componentes, esto para visualizar las tecnologías que se usarán en el proyecto, ya que con el diagrama de AWS se visualiza el artefacto a más alto nivel.

El primer diagrama es el de contexto (ver Anexo 7 - Figura No. 11), en él se identifica a los usuarios que usan el sistema, es decir, los trabajadores de Grupo Montserrat y abogados externos a los cuales se les asigna un número de casos. En segundo lugar, está el sistema de información web que interactúa con los usuarios y con el módulo de mensajería; por último, el módulo de mensajería interactúa con el sistema de información y con el deudor.

El segundo diagrama es el de componentes, donde se visualizan las tecnologías a usar dentro del proyecto. Desde la parte de frontend se usará Angular como framework principal junto con el sistema de diseño de Bootstrap; por otra parte, desde la parte de backend se usará como proveedor de servicios en la nube a AWS. Se usará el API gateway de Amazon debido a que permite manejar una arquitectura de microservicios, una base de datos Aurora PostgreSQL debido a su flexibilidad y su modelo de pagos bajo demanda, por último una instancia de S3 para el almacenamiento de los documentos.

Como se mencionó anteriormente, se usará una arquitectura de microservicios, esta se centra en la creación de software compuesto por servicios independientes que son orquestados por un bus de servicios o un API [6]. La elección de esta arquitectura va de la mano con la elección de AWS como proveedor de servicios en la nube, dado que AWS ofrece el servicio de funciones AWS Lambda, el cual permite ejecutar código sin la necesidad de un servidor, estas serán gestionadas por el API gateway y

estarán conectadas a su vez con la base de datos de Postgres, que está relacionada con la instancia de S3 donde se guardan los documentos.

Lo anterior, genera un patrón de arquitectura sin servidor ya que la gestión de la infraestructura la realiza AWS (Figura No. 3). Asimismo, se usará un bucket de S3 para almacenar los archivos correspondientes al frontend para poder realizar su despliegue a través de Amazon Cloudfront creando una distribución que se conecte al bucket para poder generar su despliegue.

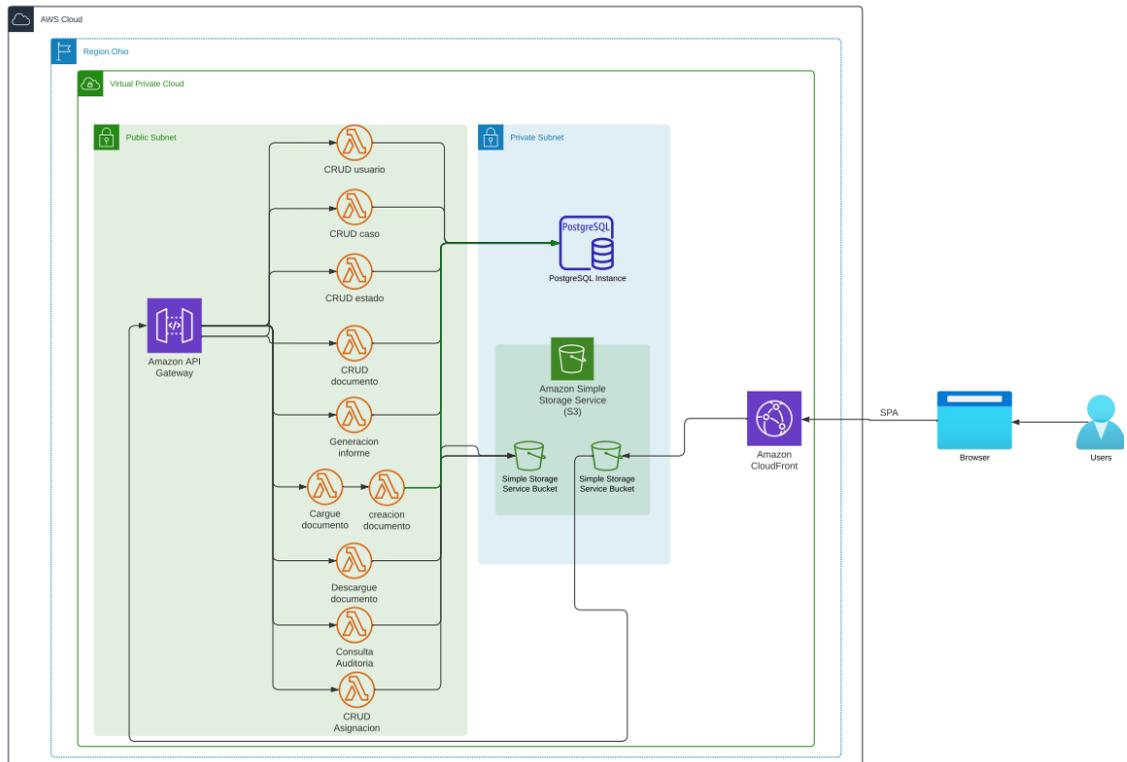


Figura No. 3 -Diagrama Arquitectura AWS
Fuente - Elaboración Propia

Por otro lado, dentro de los patrones de diseño que se implementan debido a la arquitectura y tecnologías seleccionadas, se tienen tres patrones principales. En primer lugar, está el patrón Proxy, el cual consiste en la creación de una capa intermedia entre el cliente y los servicios, usualmente se implementa por medio de un objeto que sustituye a otro objeto controlando el acceso [7]. En este caso la capa intermedia entre el cliente y los servicios de la aplicación, que son las funciones Lambda, es API Gateway, esto mejora la seguridad, la escalabilidad de la aplicación y su rendimiento.

El rendimiento se debe a que API Gateway puede actuar como un proxy caché, lo que significa que puede almacenar respuestas a peticiones que se repitan y entregar directamente al cliente si este hace la misma petición en un futuro. Por otro lado, el

tema de seguridad con API Gateway, permite manejar temas de autorización y autenticación para que solamente los usuarios autorizados tengan acceso a los servicios.

En segundo lugar, se encuentra el patrón de diseño denominado como la Capa de Servicios [8]. Este patrón se implementa dentro la arquitectura debido a que la capa de servicios se centra en las funciones Lambda, que son las que contienen la lógica de negocio.

Concluyendo, dentro del modelado de la base de datos que representa el negocio se decide no guardar los documentos dentro de la base de datos sino la ruta que tenga dentro de la instancia de S3; la razón de esto es, principalmente, por un tema monetario y de escalabilidad, dado que dentro de las variables que se tienen en cuenta para el cobro de RDS PostgreSQL está la instancia, la cual hace referencia a la capacidad computacional y de almacenamiento disponible para ejecutar los procesos de la base de datos como consultas y almacenar temporalmente los datos.

Lo anterior quiere decir que, a mayor capacidad mayor costo, por lo que si se guardan los documentos dentro de la base de datos se forzaría a seleccionar instancias de mayores capacidades que se adapten al tamaño de los archivos que se carguen. Por otro lado, está el almacenamiento de la base de datos que es el almacenamiento permanente, si se guardan los documentos dentro de la base de datos requerirá al menos 130 GB, pero debido a que se almacenan en el bucket de S3 solamente se usarán 10 GB, por lo que esto reduce considerablemente los costos.

4.2. Componente de Análisis: Descripción de la Transformación Esperada del Contexto

La solución consiste en el desarrollo de un sistema de información para la gestión de casos enmarcados en el derecho civil que maneja la empresa Grupo Monserrat. Esta solución que se diseñará y desarrollará bajo las variables de experiencia de usuario y tiempo del proceso, permitirá búsquedas sencillas de los documentos y una estructura sencilla para el orden de los casos, lo cual facilitará el entendimiento y uso de los usuarios.

El sistema contará con un módulo de mensajería para la automatización de envío de mensajes, este artefacto, introducido dentro del modelo Biopsicosocial y Cultural del actor que representa al trabajador de Grupo Monserrat, genera un cambio en una creencia y en dos hábitos (ver figura 4). Por un lado, transforma el hábito de las búsquedas exhaustivas, debido a que, por medio del sistema nombrado SGDC, el usuario realiza búsquedas sencillas de los documentos; de igual modo, se brinda una estructura de gestión de documentos para que cualquier usuario de la empresa y externo, puedan llevar la gestión de los casos de una manera práctica, todo gracias a la UX del sistema. El cambio en este ámbito transforma la creencia de que el orden de las carpetas solo lo entiende quien las creó, ya que el sistema SGDC está diseñado

bajo la variable de UX y permitirá una interacción sencilla y práctica con el sistema para el entendimiento del usuario promedio.

En concordancia, el segundo hábito que se transforma es la revisión constante de las notificaciones del celular relacionadas con el correo electrónico; por ende, este hábito disminuye debido a que el sistema SGCDC cuenta con un módulo para la automatización del envío de mensajes. A su vez, esto cambia la creencia del trabajador de que debe estar atento al celular todo el tiempo, ya que el sistema permite al trabajador delegar la carga de trabajo de esa tarea al sistema.

Finalmente, está la transformación en la creencia de que la implementación de tecnología es tediosa y costosa. El sistema contará con funcionalidades que se ajusten al proceso de negocio de Grupo Monserrat, obteniendo mejores resultados que los ofrecidos por herramientas como Drive en temas de tiempo y disminución de complejidad del proceso de negocio; por ello, este cambio en la creencia genera un cambio en el hábito de exploración tecnológica ya que, a futuro, se requerirán actualizaciones para que el sistema se mantenga funcional y competitivo, por lo que será necesaria una vigilancia tecnológica del mercado.

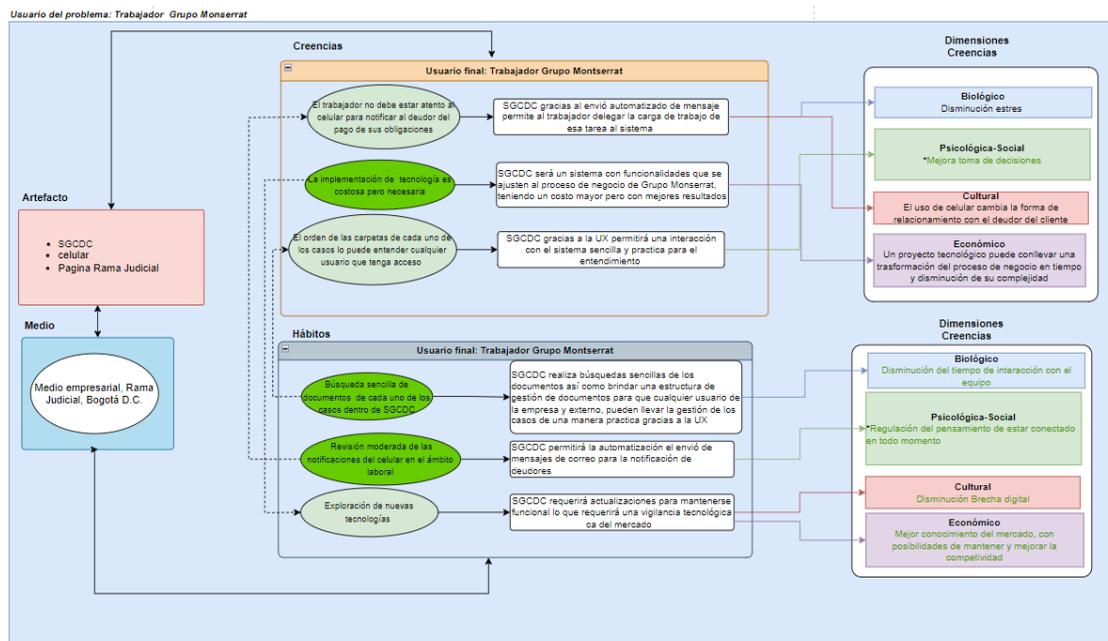


Figura No. 4 - Diagrama del Modelo Biopsicosocial desde la perspectiva de la Solución
Fuente - Elaboración Propia

De igual forma, como resultado de la implementación del sistema, se espera que el impacto de los efectos planteados en la Sección 2, sea inexistente o disminuya en el proceso; también, se espera que la complejidad para la explicación del orden de carpetas a personas externas e internas desaparezca y que esto permita un descenso de los niveles de estrés. Además, se espera una mejora en la toma de decisiones y en el tiempo de ejecución del proceso, puesto que el sistema permite encontrar los

documentos fácilmente, erradicando el efecto de desgaste de las búsquedas exhaustivas gracias a su orden estructurado. Para concluir, gracias al módulo de mensajería se espera disminuir el uso prolongado del celular respecto al proceso de envíos de correos.

5. Diseño Metodológico.

Para esta sección, se realiza el análisis de los problemas identificados, por lo que el uso de la Estructura de Desglose del Trabajo (EDT) facilita el establecimiento de tareas a realizar en conjunto con las actividades para lograr la culminación de las mismas. El planteamiento de estas tareas que se encuentran en el primer nivel, están atadas para cumplir con éxito el objetivo general establecido. A continuación se mostrará el diagrama realizado en la *Figura No. 5*.

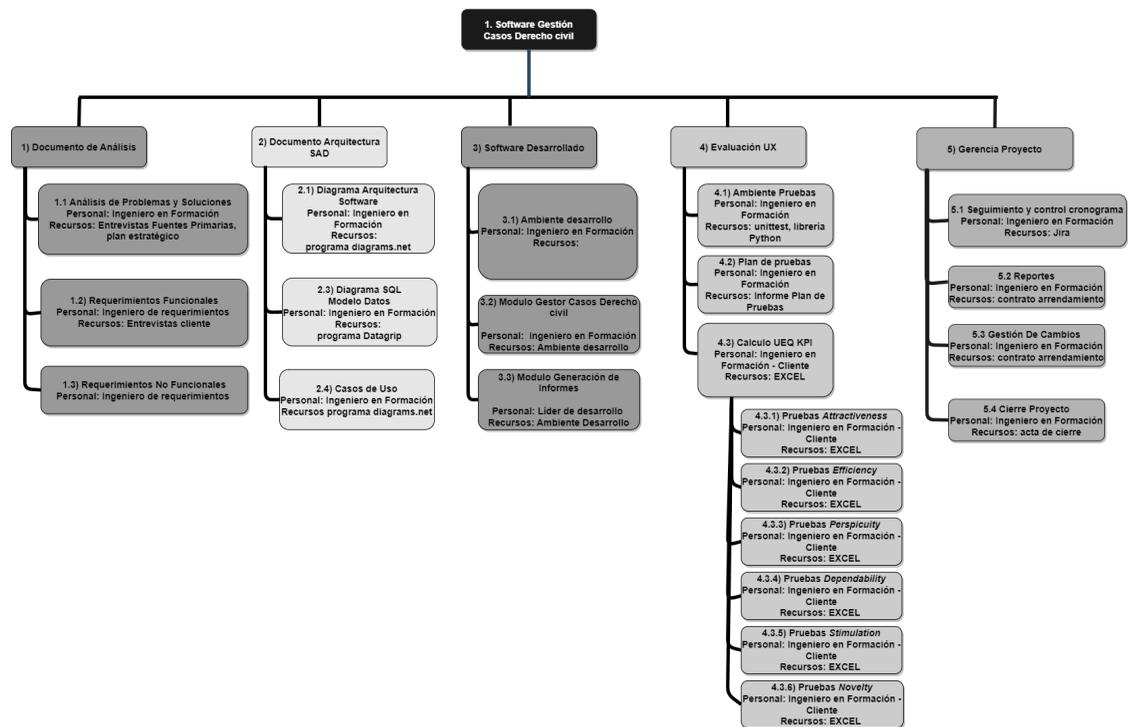


Figura No. 5 - Análisis EDT del Grupo Montserrat

Fuente: Elaboración propia

En el primer nivel se tiene el *Documento de Análisis*, el cual consiste en la realización de entrevistas al cliente, esto con el fin de conocer el estado actual de la empresa y, así, detectar falencias que pueden ser solucionadas a partir de la solución a plantear; por lo tanto, para proponer una solución, es necesario elaborar un análisis desde el modelo Biopsicosocial con el que se pueda graficar el estado actual de la empresa. Este es conocido como el *AS-IS* y el estado de la empresa después de implementar la solución, conocido como el *TO-BE*, a la vez, se realiza un documento de árbol de

problemas para visualizar de manera clara el origen del problema y sus causas, esto con el fin de buscar posibles soluciones para abordar el problema desde varias perspectivas.

Una vez realizada la entrevista, se procederá a investigar qué otras soluciones se encuentran en el mercado, en este punto se indagan tecnologías implementadas, metodologías aplicadas, sistemas de diseño, entre otros. Para el levantamiento de requerimientos funcionales y no funcionales, fue necesario realizar un levantamiento de Historias de Usuario, en el cual, dependiendo de cada rol que va a interactuar en la solución, se definirán sus funcionalidades. Por último se realiza un estudio de cuáles tecnologías son óptimas para la solución.

En el segundo nivel se tiene el *Documento Diseño*, en el cual se realizarán los respectivos diagramas de software para ilustrar las infraestructuras que se implementarán en la solución, como lo son los diagrama SQL y C4, a la vez de la realización del Mockup para definir las pantallas que verán los usuarios en la solución final.

En el tercer nivel, se encuentra el *Software Desarrollado*, en el cual se realizará la codificación de los diagramas y mockups realizados en el segundo nivel, aquí se establecen los módulos a desarrollar y entregar en cada iteración que indica la metodología ágil.

En el cuarto nivel, se realiza el *Reporte de Pruebas*, donde se propone un plan de pruebas y se adecua el ambiente para realizar las mismas, esto con el fin de probar las funcionalidades implementadas, a la vez que se realiza una prueba con el usuario final en el cual va a interactuar con la solución y realizar tareas que espera que se logren, con esto se logra verificar con el usuario que todas las funcionalidades se estén ejecutando con total normalidad y que se elaboren para identificar posibles errores que no se identificaron en las anteriores pruebas.

En el quinto nivel, se realiza la *Puesta en Producción*, el cual consiste en poner en *producción* el producto final, en este punto el usuario final podrá hacer uso del software con total libertad. Es necesario realizar pruebas de *post-producción*, para verificar que la solución esté funcionando de manera correcta, y que el usuario final no presente problemas al momento de utilizarlo para eso se usaron la pruebas de caja negra.

En el último nivel, se encuentra la *Gerencia del Proyecto*, en donde se elabora un seguimiento de la solución por medio de reportes conocidos como *logs* para verificar que el software esté funcionando correctamente, a la vez, de realizar una gestión de cambios, el cual consiste en si se desea mejorar alguna funcionalidad se haga la evaluación de que al momento de implementar este cambio no genere una incompatibilidad que desencadene en que la solución se afecte de manera negativa.

Una vez establecido el EDT para abordar la solución, es necesario encaminar las tareas establecidas por medio de una metodología ágil, para el caso de la implementación de la solución se realiza un híbrido entre SCRUM y Cascada, en

donde se usa Cascada para utilizar las fases que ofrece, como son *Análisis - Diseño - Desarrollo - Testing - Implementación*; sin embargo, como lo plantea Jimenez y Idrovo [12] en cada etapa se realizan las tareas correspondientes a cada fase, y hasta que no se completen no se continúa a la siguiente fase, esto es ideal para minimizar los errores.

A pesar de lo anterior, un punto negativo es que no existe una intervención directa del cliente durante el desarrollo de la solución y no hay entregas parciales antes de la entrega de la solución final, por ende, puede generarse un descontento por parte del cliente, produciendo un regreso a la fase donde se presentó dicho descontento, lo que implica una desviación de la metodología establecida, además de que el despliegue del producto final se retrase. Para solventar las falencias identificadas de Cascada, se decidió utilizar las características de SCRUM, como lo explicó Córdova [6], se establecen las tareas de acuerdo a la fase brindada por Cascada para realizarse dentro de las fechas acordadas con el equipo de desarrollo, una vez se llega a esta fecha, se genera el conocido *producto mínimo viable*. Este *producto mínimo viable* se obtiene al finalizar cada SPRINT, que es el ciclo de trabajo donde se establecen las tareas en fechas establecidas, que en el caso del Desarrollo Tecnológico es cada 12 días, lo anterior aplica para cualquier fase en la que se encuentre de Cascada.

Es importante resaltar que el cliente está presente en todas las fases de Cascada, con la finalidad de solventar dudas durante el desarrollo del artefacto o simplemente para que pueda visualizar el avance del mismo; no obstante, es importante que el cliente también sea un participante activo, en especial en la etapa de *Desarrollo*, ya que, es donde se aplican las pruebas de UEQ KPI al producto mínimo viable, con el fin de que el cliente de manera imparcial diligencia una encuesta para evaluar si la aplicación es intuitiva, generando en cada entregable un KPI que sirve para medir la evolución del producto, por lo tanto, dependiendo del resultado de la prueba, se generarán estrategias para resolver las falencias en las áreas donde se presentaron bajos resultados.

En la *Figura No. 6*, se puede evidenciar de manera gráfica la unión entre *Cascada* y *SCRUM* generando así la metodología a implementar en el desarrollo de la solución. A continuación, se puede encontrar diferentes tipos de validación del ciclo de transferencia tecnológico, tales como: la validación dinámica, donde se aplica el UEQ KPI, donde se realizará la medición e interpretación de los resultados generando un KPI por cada entrega de un producto de valor, además se implementará la metodología de investigación-acción la cual según [14] permite a los investigadores realizar ciclos iterativos para evaluar el desarrollo de su solución y así generar reflexiones para implementar cambios sobre la marcha, la metodología permite evaluar usando diferentes instrumentos, en este caso se utilizaron cuestionarios, como por ejemplo el de UEQ KPI el cual es importante para medir la variable interés en cada producto mínimo viable, gracias a que es cíclica se acomoda con SCRUM, lo cual permite que se le entregue al cliente y se evalúe en conjunto con él en cada iteración; se tiene la validación estática, en la cual se realizarán pruebas de las funcionalidades, con el fin de verificar que cumplan con lo establecido en la

Historias de Usuario; por último se tiene la validación académica, en donde se realizan las pruebas de rendimiento del artefacto, a la vez de pruebas de carga, entre otros, con la finalidad de que se garantice que el software sea totalmente estable para el usuario final.

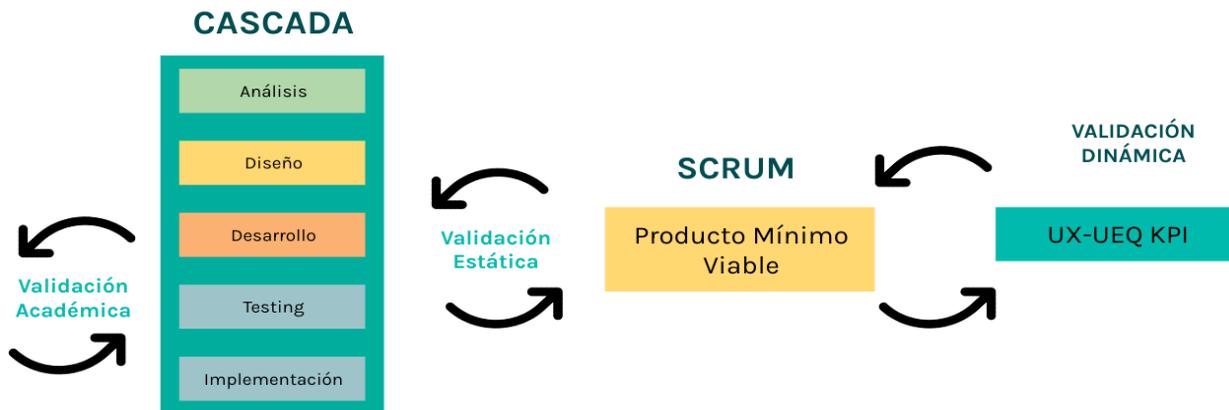


Figura No. 6 - Modelo gráfico de la Metodología

Fuente: Elaboración propia

Una vez se tiene la metodología definida, se procede a la organización de tareas por medio de un cronograma distribuido por Sprints, este cronograma se puede visualizar en la *Figura No. 7*. Gracias a la metodología de Cascada, se pueden categorizar las epicas con colores, por ejemplo, el color azul es la etapa de *Análisis*, el color verde es la etapa de *Diseño*, el color naranja es la etapa de *Desarrollo*, el color morado es la etapa de *Testing* y el color azul oscuro es la etapa de gestión de cambios que aunque no esté estipulado en la metodología de Cascada, sirve para realizar pequeños cambios que se presenten en el proyecto y dar un cierre al mismo.

En conclusión, la implementación de la metodología mencionada, en conjunto con el cronograma propuesto, ha resultado en un progreso significativo en la ejecución de las tareas establecidas en cada uno de los sprints. Es importante destacar que se ha observado un avance notable en aspectos clave del proyecto, como los Mockups, las Historias de Usuario, los Casos de Uso y el Diagrama C4, los cuales han sido fundamentales para su desarrollo. Esto se debe en gran medida a que, en cada iteración, se han recibido validaciones, sugerencias y correcciones por parte del cliente, lo cual ha enriquecido aún más el desarrollo del proyecto.

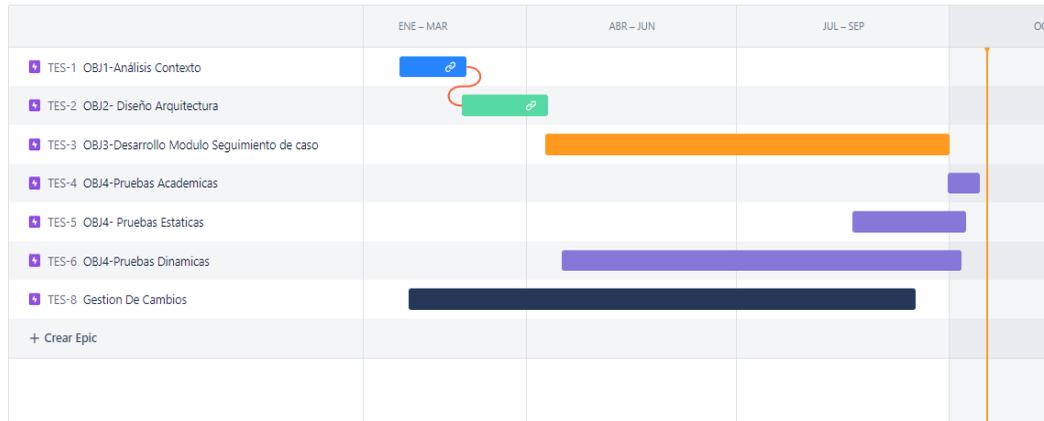


Figura No. 7. Cronograma del proyecto en la plataforma Jira
Fuente: Elaboración propia

6. Resultados y Discusión

En esta sección se presentan los resultados obtenidos del desarrollo del proyecto SGCDC, se maneja por medio de las épicas que proporciona la metodología de SCRUM; estas épicas, se dividen en historias de usuario realizadas con anterioridad, las cuales se dividen en tareas para cumplir cada historia de usuario. En este caso, se nombrará la épica según el objetivo específico al que apunta y los sprint's que se requieren llevar a cabo para completar esa épica, en cada sprint se realizan una serie de entregables que contribuyen al cumplimiento de cada objetivo específico.

6.1 Épica Análisis del Contexto

La primera épica a desarrollar corresponde al análisis del contexto, esta inicia desde el 5 de febrero hasta el 5 de marzo del presente año, esta va dirigida a cumplir con el objetivo específico número 1 (ver apartado 4 - Objetivos específicos 1). Se realizó un sprint extenso, de 4 semanas, orientado a desarrollar los entregables relacionados con el análisis del contexto de la empresa Grupo Montserrat. La herramienta utilizada para realizar la gestión de los sprints fue JIRA (ver Anexo 5 - Figura No. 8).

Se llevaron a cabo Scrum Meetings para establecer las tareas del sprint donde se traza como primera tarea el desarrollo de los diagramas del modelo BPCS desde la problemática y solución, así como el desarrollo del árbol de problemas para el entendimiento de las causas y los efectos de la problemática que se detecta dentro del Grupo Montserrat.

Durante las primeras semanas se desarrollaron los entregables anteriormente mencionados; inicialmente, se realizó una reunión con el director del proyecto el día 10 de febrero de 2023 (ver Anexo 11 - Figura No. 18) donde se sugirió realizar un diagrama BPMN AS IS para visualizar los distintos procesos de negocio que realiza la empresa Grupo Montserrat. Se solicitó a la empresa Grupo Montserrat por medio de un correo electrónico, brindar documentación acerca de un ejemplo del cómo

con el sistema de información; en cambio, los dos diagramas de BPMN tanto el AS IS (ver figura 8) como el TO BE (ver Anexo 4- figura 7) que se realizaron van encaminados a apoyar el documento de arquitectura SAD, que es el entregable final del objetivo específico número dos (ver apartado 4 - Objetivos específicos 2).

Al final del sprint se realizó el sprint review para reflexionar sobre el desarrollo de las tareas. Y se realizó un primer borrador de las historias de usuario para el entregable del objetivo específico número 1 (ver figura 7).

Enunciado de la historia					Criterios de aceptación			
Identificador (ID) de la historia	Rol	Característica / Funcionalidad	Razón / Resultado	Número (#) de escenario	Criterio de aceptación (Título)	Contexto	Evento	Resultado / Comportamiento Esperado
00-0000-0001	Como usuario tipo Administrador	Debo poder crear casos	Con la finalidad de, crear los casos de los clientes	1	Correcta creación del caso	En caso de que se desee crear un nuevo caso	Cuando se selecciona el botón de agregar un nuevo caso	Se mostrará un pop-up donde se notifique el éxito de la operación, así mismo se podrá apreciar la publicación del caso
				2	Caso no creado	En caso de que no sea posible agregar el nuevo caso; debido a errores al llenar campos requeridos	Cuando se seleccione el botón de "Crear"	El botón se encontrará inactivo y en un color tenue, al igual que se resaltarán en la parte inferior de los campos incompletos un mensaje indicando su necesidad de validez para realizar la acción
				3	Error al crear el caso	En caso de que no se haya podido modificar el progreso del respectivo caso, debido a, errores internos del servidor	Cuando se selecciona el botón de agregar un nuevo caso	Se mostrará un Pop-up donde se notifica el desacierto al intentar agregar el caso respectivo por problemas internos en el servidor

Figura No. 7. Historia de usuario creación caso

Fuente: Elaboración propia

6.2 Diseño Arquitectura

La segunda épica a desarrollar corresponde al diseño de arquitectura, esta inicia desde el 4 de marzo hasta el 9 de abril, va dirigida a cumplir con el objetivo específico número 2 (ver apartado 4 - Objetivos específicos 2). Dentro de la misma, se realizaron dos sprints de 2 semanas cada uno. Uno del 13 al 24 de marzo del 2023 y otro del 27 de marzo al 9 de abril. Acorde al EDT se inicia el proceso de diseño, por lo tanto, se establece como Sprint Goal para el segundo sprint realizar un adelanto en esta etapa para establecer la arquitectura, tecnologías y costos del sistema de información.

Se realizaron SCRUM Meetings para establecer las tareas del sprint 2, por lo que se establece realizar tres entregables; en primer lugar, un diagrama de arquitectura C4 en dos de sus niveles, para visualizar el contexto donde estará el sistema y las tecnologías que se usarán. En segundo lugar, un diagrama de arquitectura de AWS para determinar los servicios que se usarán y visualizar cómo estos se acoplan a la arquitectura de microservicios y, en tercer lugar, en base al diagrama de arquitectura AWS establecer los costos de los servicios seleccionados.

La primera semana se realizan los diagramas C4 desde la perspectiva del contexto (ver anexo 7 - Figura 11) y desde la perspectiva de componentes (ver Figura 8). Desde la perspectiva del contexto, se identificaron los actores que intervienen con el sistema o con algún módulo en específico; además, se obtuvieron dos actores principales: los trabajadores de Grupo Montserrat y los abogados externos a las cuales se les asigna un caso. Los trabajadores y abogados interactúan con el sistema de información.

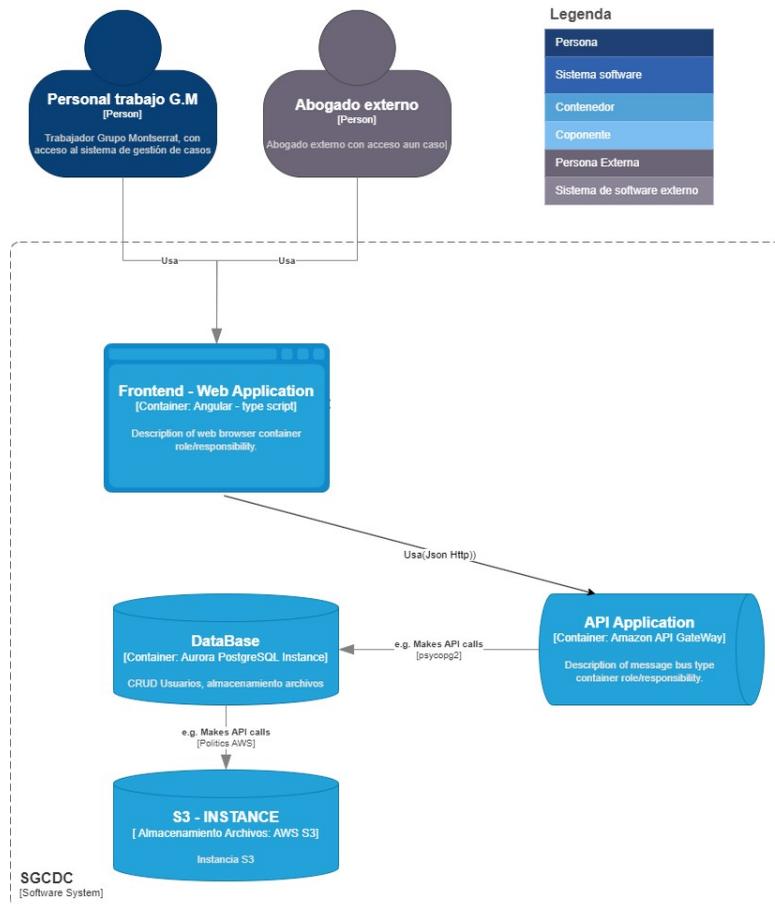


Figura No. 8. Diagrama C4 - contenedores del sistema

Fuente: Elaboración propia

Por otro lado, desde el diagrama a nivel de componentes se visualizan las tecnologías a usar; inicialmente, desde la parte del Frontend, se establece usar el framework Angular con el sistema de diseño de Bootstrap. Se escogió como proveedor en la nube a AWS, debido a que poseen servicios que permiten realizar una arquitectura de microservicios de manera simple; también, dentro de los productos de AWS se seleccionaron los siguientes: API Gateway, para orquestar los microservicios, de base datos a PostgreSQL y S3 para el almacenamiento de los documentos. A su vez, se seleccionaron ciertos patrones de diseño que se nombran en la sección del artefacto (ver apartado 4.1).

Del mismo modo, se realizó el diagrama de arquitectura de AWS, en este se visualizan como los productos de AWS interactúan bajo la arquitectura de microservicios (ver figura 3). Por un lado, se tiene Amazon API Gateway para la gestión de los servicios, es el intermedio entre la capa de servicio y las peticiones de los usuarios; igualmente, estos servicios se realizan por medio de funciones Lambda, un servicio que permite ejecutar código sin la necesidad de un servidor, dichas funciones se usarán para realizar las operaciones que requieren la lógica de negocio.

Asimismo, se visualiza la base de datos la cual es RDS PostgreSQL y el servicio de S3 para el almacenamiento de archivos.

Se realizó la gestión de los costos de los productos teniendo en cuenta cada una de las variables que se requieren para su cobro (ver figura 9). Esto se realizó con el fin de aclararle al cliente cuánto costaría mantener el sistema de información en funcionamiento en la nube.

Producto AWS						
Configuración - API						
Amazon API Gateway	tipo Locacion	Region		US East (Ohio)	Precio US	Tiempo
	Capa Gratuita	Llamadas Api -HTTP		1000000	\$ -	12 meses
	REST API - request units	millions	Request	1	\$ 3.50	mensual
				2	\$ 7.00	mensual
	HTTP API - request units	millions	Request	1	\$ 2.00	mensual
		Average size of each request	MB	1	\$ 4.00	mensual
		millions	Request	2	\$ 4.00	mensual
Average size of each request		MB	2	\$ 8.00	mensual	
AWS Lambda	tipo Locacion	Region		US East (Ohio)	Precio U	Tiempo
	Capa Gratuita	Arquitectura x86		1 million solicitudes- 1000000	\$ -	
	Arquitectura	x86 -arm		400 000 GB/segundos Tiempo Computo	\$ -	
	Number of requests -PER MOUTH	2000000				
	Duration of each request (in ms)	100			\$ 0.20	mensual
	Amount of memory allocated (GB)	2				
	Amount of ephemeral storage allocated(GB)	1				
PostgreSQL instance	tipo Locacion	Region		US East (Ohio)	Precio U	Tiempo
	Instancia	db.t4g.medium	vCPU:2 Memory: 4 GB Network Performance: Up to 5 Gigabite			
	Pricing model (modelo precios)	OnDemand				
	Utilización	%Utilized/Moth	100	Hours/Day -- 24	\$ 54.82	mensual
			70	Hours/Day -- 17	\$ 38.83	mensual
			50	Hours/Day -- 12	\$ 28.18	mensual
			40	Hours/Day -- 9.6	\$ 22.00	mensual
	Database Storage	GB	10			
	Baseline IO rate	per second	1			
	Peak IO rate	per second	1			
Duration of peak IO activity	Hours per month	1				
Amazon simple store S3	S3 Standard storage	GB	130			mensual
	PUT, COPY, POST, LIST requests to S3 Standard	Number Request	1000000			mensual
	GET, SELECT, and all other requests from S3 Standard	Number Request	1000000			mensual
	Data returned by S3 Select	GB	10		\$ 3.88	mensual
	Data scanned by S3 Select	GB	65			mensual
	Data Transfer					mensual
	Inbound Data Transfer	Internet Free	TB	1		mensual
Outbound Data Transfer	Internet 0.05 USD-0.09 per	GB	15		mensual	

Figura No. 9. Costos productos al mes de AWS

Fuente: Elaboración propia

Por un lado, se gestionaron los costos para Amazon API Gateway donde se identificó la capa gratuita del producto que se realiza por medio del número de peticiones, en este caso son hasta un millón de peticiones al mes, durante 12 meses. API Gateway tiene tres tipos de API, se compararon las dos primeras opciones que se ajustaban mejor al proyecto; por lo tanto se tiene REST API y HTTP API, en ambas opciones los parámetros de cobro son los mismos: número de peticiones y tamaño promedio de las peticiones. Se tomó para ambos casos un estimado de un millón y dos millones de peticiones al mes, con un tamaño promedio de la petición de 1 MB o 2 MB. Ambos modelos de API se distinguen en los servicios extras que ofrece AWS, por lo que decide usar REST API debido a los servicios de seguridad y autenticación que ofrece AWS.

El siguiente servicio y tecnología son las funciones Lambda, estas tienen una capa gratuita de un millón de peticiones, por lo que dentro de las variables que se tienen en cuenta para su cobro se encuentra el tipo de arquitectura, el número de peticiones al mes, la duración de la petición y el almacenamiento asignado.

De igual forma, se encuentra PostgreSQL, que dentro de los parámetros principales para su cobro está la instancia, que hace referencia a la capacidad de cómputo para realizar las operaciones de la base de datos y almacenar datos de forma temporal; es importante tener en cuenta que, una mejor instancia conlleva mayores cobros. El modelo de precios seleccionado fue de "bajo demanda", ya que, permite ajustar los costos de la base de datos acorde a su utilización. Además, se realizaron estimaciones con un 100%, 70%, 50% y 40% de uso, al frente, cada uno tiene un estimado del uso de horas diario que representa cada porcentaje. Por último el parámetro con mayor relevancia es el almacenamiento permanente de la base de datos, para este se seleccionaron 10 GB, ya que, se busca guardar la ruta del archivo en la base de datos (ver apartado 4.1 - Diseño Artefacto).

En concordancia, está la instancia S3 de AWS, la cual tiene tres principales variables para su cobro; la primera variable, hace referencia al tamaño de almacenamiento, por lo que se seleccionó 130 GB debido a que Grupo Montserrat maneja actualmente un plan con Google Drive con esa capacidad de almacenamiento. El segundo tipo de variable hace referencia al número de solicitudes, que se agrupan en dos grupos; el primer grupo son solicitudes de tipo PUT, COPY, POST, LIST y el segundo grupo son solicitudes de tipo GET, SELECT y demás clasificaciones. Para ambos se estableció un rango de un millón de solicitudes al mes.

Siguiendo con lo anterior, el otro tipo de variables están relacionadas con el tamaño de solicitud de retorno y el escaneo de la instancia de S3, si la instancia de S3 ya tiene documentos cuando se realiza una solicitud de tipo SELECT, AWS escanea la instancia; por lo tanto, a mayor GB almacenados mayor será el escaneo. El último tipo de variable está relacionada con el tamaño del canal para la transferencia de datos, la variable de *inbound* tiene una capa gratuita de 1 TB y para la variable de *outbound* se establece un valor de 15 GB teniendo en cuenta el tamaño de ciertos documentos que maneja Grupo Montserrat.

A su vez, se plantearon 4 escenarios para los costos donde influye principalmente la utilización de la base de datos (ver Anexo 9 - Tabla 4) por lo que se comienza con un 100% de utilización con un valor mensual de 331,172 pesos colombianos mensuales hasta 163,881 pesos colombianos, con una utilización del 40%. Este entregable de gestión de costos tuvo una repercusión importante dentro del modelo de la base de datos, ya que, se buscó minimizar sus costos e igualmente en la selección de ciertos lenguajes para las funciones Lambda como fue el caso de Python para el cargue y descargue de documentos.

Finalmente, para la base de datos se realizó un modelo entidad - relación (ver Anexo 10 - Figura 16), así que, se establecieron las entidades que representan el negocio, sus atributos y sus relaciones. Se pasó este diagrama a un modelo relacional (ver Anexo

10 - ver Figura 17) para establecer las relaciones de las tablas y sobre todo las llaves primarias y foráneas. Posteriormente se realizó el diagrama SQL para visualizar el primer diseño de la base datos (ver figura 10).

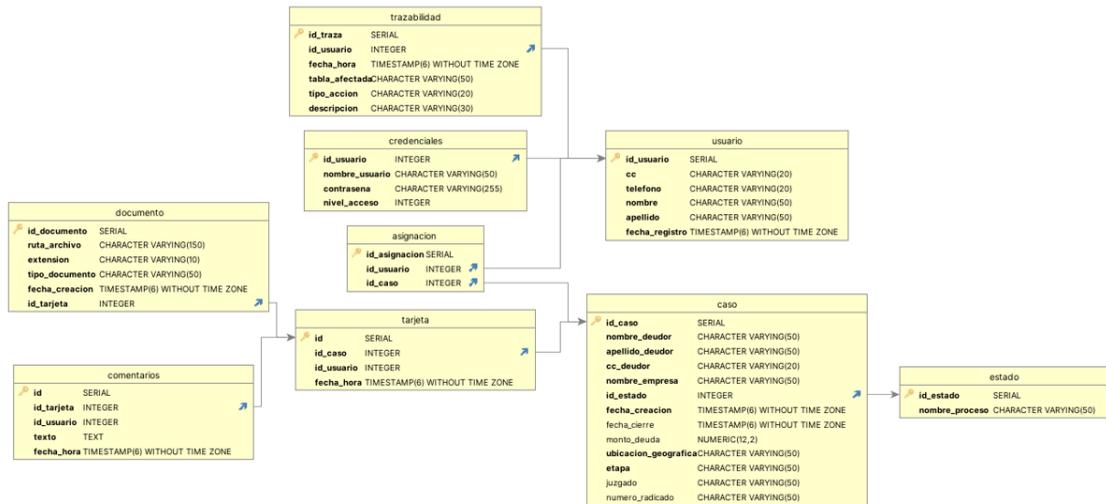


Figura No. 10. Diagrama relacional DB corregido

Fuente: Elaboración propia

En la segunda semana del sprint se realiza una reunión con el director el día 22 de marzo de 2023 (ver Anexo 11 - Figura No. 20) en la cual se muestran los avances realizados tanto a nivel de arquitectura, tecnologías seleccionadas y un primer avance del modelado de la base de datos. Del mismo modo se discutió el tema del cambio de los objetivos específicos, debido a que, hacía falta especificar el “para qué” de estos; dentro de la reunión, se determinó el “para qué” de los objetivos y lo que está relacionado con el entregable esperado, aunque el entregable no es el fin último del objetivo, permite alinear con el cumplimiento del objetivo general.

Por este motivo, se determinaron los entregables de historias de usuario finalizadas, documento SAD, el sistema de información y el documento de pruebas, entregables que reflejan el cumplimiento de los distintos objetivos específicos. Por otro lado, se mostró la primera versión del modelo de la base de datos donde el director sugirió la creación de una tabla de trazabilidad, que permita realizar una auditoría del sistema.

Se realizó el sprint review para reflexionar sobre el desarrollo de las tareas, las cuales, se cumplieron en su totalidad, pero en el caso del modelo de la base de datos, se deben realizar cambios dada la recomendación del director y teniendo en cuenta los variables para el cobro en AWS.

El sprint número tres se plantea para realizar un ajuste a los entregables del modelo SQL y realizar un adelanto en la parte de diseño y análisis del sistema de información. Acorde a ese objetivo, en la parte de análisis, gracias a los entregables

del primer sprint y a las reuniones realizadas, se otorga la tarea del ajuste de las historias de usuario, que es el entregable final del objetivo específico número 1 (ver apartado 4 - Objetivos Específicos 1).

Del mismo modo, en la parte de diseño, se decide realizar el primer borrador de mockups y la modificación del modelo de la base datos, así como la creación del script del mismo y su trigger para la tabla de trazabilidad. Este sprint, debido a la diversidad de sus entregables, apoya los objetivos específicos número 1, 2 y 3 (ver apartado 4 - Objetivos Específicos 1, 2, 3).

Asimismo, para el módulo SQL de la tabla de datos, se agrega la tabla de asignación que permite al usuario administrador asignar 1 o n casos a un abogado; también, se cambia el tipo de dato del atributo archivo para almacenar la ruta del archivo en la instancia S3 y se agrega la tabla de trazabilidad. Para los mockups se realiza la vista para el login (ver anexo 15 - figura 37), una sección para visualizar todos los casos (ver figura 11) y la sección de creación del caso (ver figura 12).

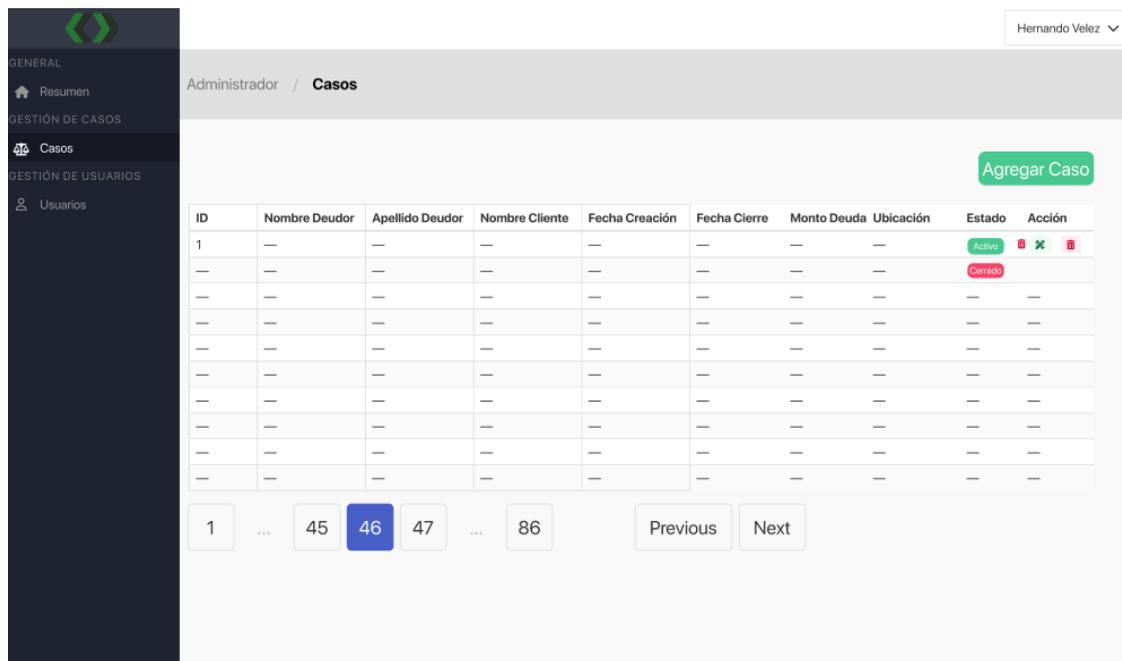


Figura No. 11. Mockup sección casos
Fuente: Elaboración propia

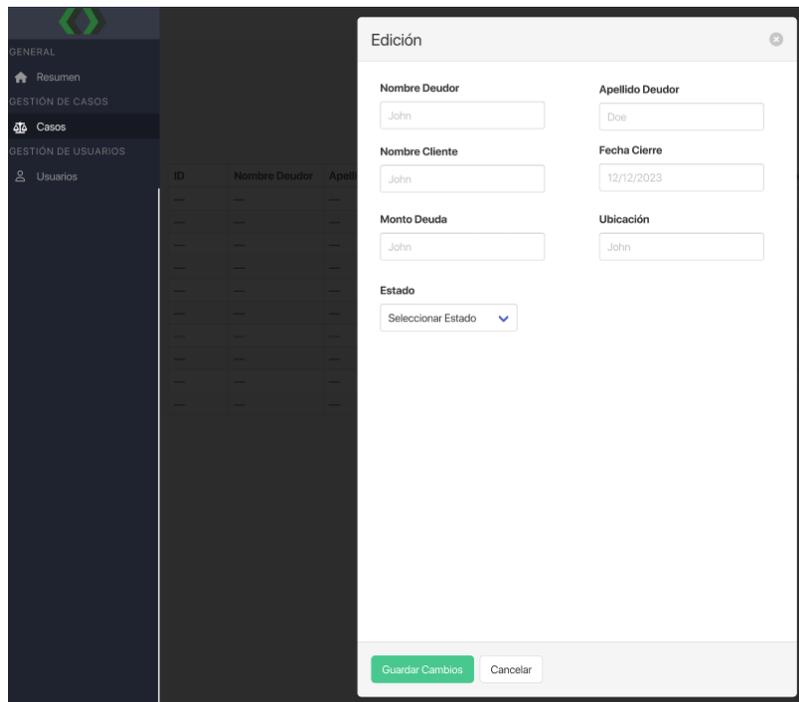


Figura No. 12. Mockup creación casos
Fuente: Elaboración propia

Se realiza una reunión con el director del proyecto el día 1 de abril de 2023 (ver Anexo 11 - Figura No. 21), se trataron 3 temas fundamentales; primero, los ajustes realizados a los objetivos específicos, segundo, los cambios del modelo SQL con la inclusión de la tabla trazabilidad y asignación, y finalmente, tercero, el borrador de los primeros mockups. Al finalizar la reunión se hace la sugerencia de que en la próxima reunión con Grupo Montserrat se exponga la información que se va almacenar en el sistema de las distintas tablas y que estos hagan una retroalimentación para conocer si es necesario agregar más información a las tablas propuestas. Asimismo, se propuso mostrarles los mockups para conocer su opinión y recomendaciones del diseño.

El día 4 de abril de 2023 se realiza la reunión con Grupo Montserrat (ver Anexo 11 - Figura No. 23), en ella se discuten los atributos de las tablas donde se realizan varias modificaciones a la tabla de caso (ver anexo 9 - tabla 3). De igual forma, se discuten funcionalidades del sistema, donde se llegó al acuerdo de agregar funcionalidades de exportación de informes en vez del módulo de mensajería y se acordaron compromisos para los mockups.

Posteriormente, se realizó el script de la base datos (ver anexo 10-figura 16) y del trigger ver anexo 10-figura 17) para la tabla de trazabilidad. Los arreglos de la base datos van dirigidos al documento SAD y estos últimos entregables que ya van realizados con el desarrollo del sistema van encaminados a apoyar el objetivo

específico 3 (ver apartado 4 - Objetivos específicos 3).

Los entregables generados durante este sprint número 3, van encaminados a apoyar el entregable del objetivo específico 2 documento SAD (ver apartado 4 - Objetivos Específicos 2). En el caso de los scripts de la base de datos van orientados a apoyar el objetivo específico número 3 (ver apartado 4 - Objetivos Específicos 3) y, además, se realizó el sprint review para reflexionar sobre el desarrollo de las tareas.

6.3 Desarrollo Módulo de seguimiento

La tercera épica a desarrollar corresponde al desarrollo del módulo de seguimiento de caso, que fue el que se establece como alcance para este proyecto esta etapa inicia desde el 9 de abril hasta el 30 de septiembre del 2023, orientada a cumplir con el objetivo específico número 3 (ver apartado 4 - Objetivos específicos 3). Para esta épica se realizaron 8 sprints.

El sprint número cuatro empieza el día 12 de abril del 2023 y termina el 24 de abril del 2023 se establece como sprint Goal la configuración y conexión de los distintos servicios de AWS para realizar la arquitectura de microservicios, ajustes de los mockups y configuración de Angular.

Se realizó la configuración de la base de datos PostgreSQL bajo los estándares de la capa gratuita (ver Anexo 13 - figura 24), se continuó con la elaboración del código de la función Lambda en donde se realiza la conexión a la base de datos por medio de la biblioteca de python aws-psycopg2 (ver Anexo 14 - figura 25) y se prosigue con el cargue del código en la función Lambda y su testeó en la consola de AWS (ver Anexo 14 - Figura 26).

Por último, se realiza la conexión de la función Lambda con el API Gateway (Anexo 14 - Figura 27), se testeó la función Lambda dentro del API Gateway (ver figura 13) y finalmente se configura y testeó el endpoint del servicio (ver Anexo 14 - Figura 29). Por lo que se puede observar la configuración base de la arquitectura de microservicios y, que, se cumplió acorde a lo planteado en la descripción del artefacto (ver apartado 4.1 - Descripción del artefacto).

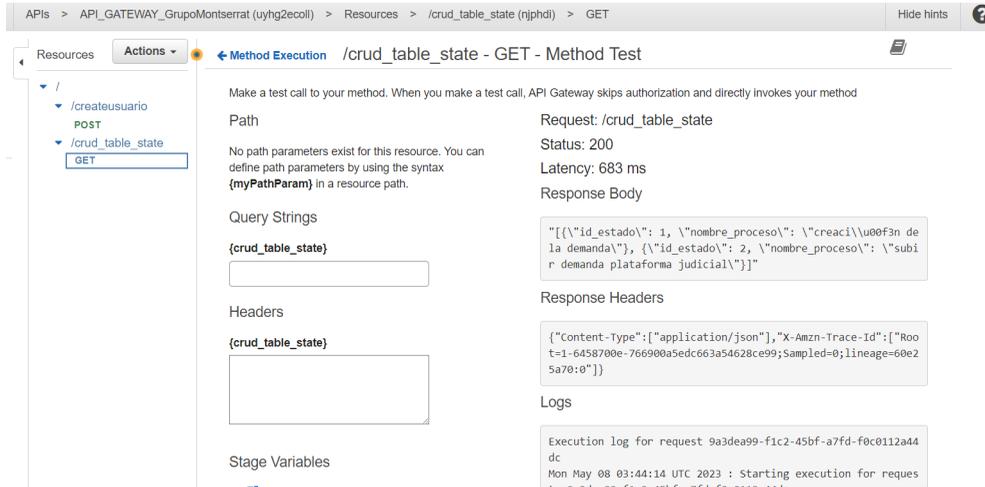


Figura No. 13. Configuración API Gateway con el servicio Lambda

Fuente: Elaboración propia

Para los sprints número cinco y número seis, se realizan los CRUD's de las tablas de usuario, credenciales y asignación teniendo en cuenta la configuración de las lambdas y el Api Gateway. Para finalmente en el Postman organizar cada una de las peticiones (ver figura 14).

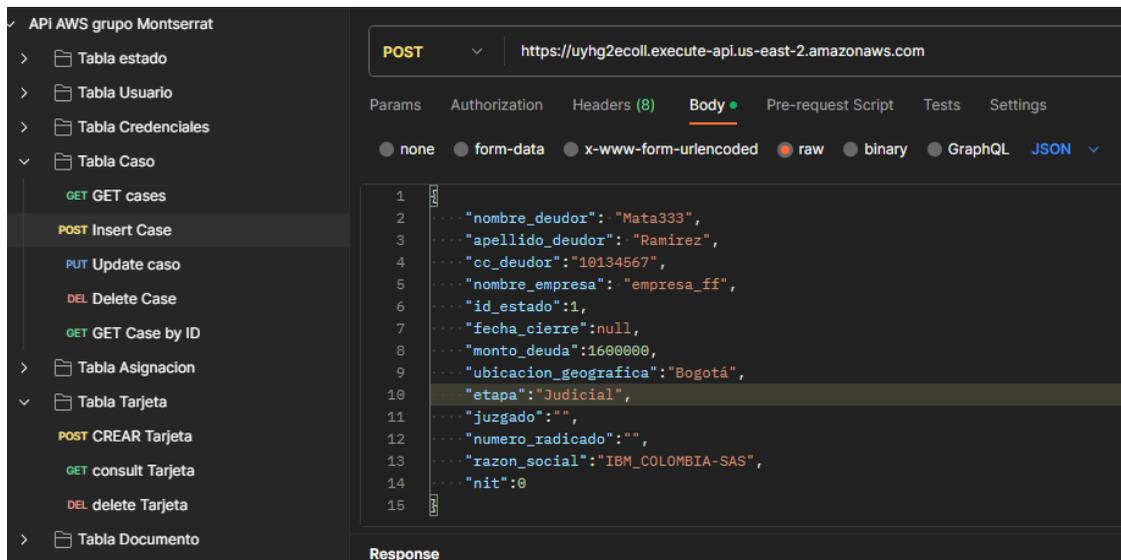


Figura No. 14. Organización Postman de las distintas peticiones

Fuente: Elaboración propia

Asimismo, se realizaron ajustes a la base de datos, donde se modificaron las relaciones de la tabla de documentos para que ésta estuviera asociada a la tarjeta y no al caso, así se tiene un mejor manejo de las actualizaciones que se realizan del caso, viéndose reflejadas en las tarjetas que contienen los documentos y los comentarios

(Figura 10).

El sprint número siete empieza del 9 de junio de 2023 al 23 de junio de 2023, en este se establece como sprint Goal la maquetación del módulo de *Casos* del usuario administrador. Se realizó únicamente la maquetación en el framework de Angular (Ver Anexo 15 - Figura 36). Durante este sprint se le plantea al beneficiario dejar el módulo de generación de reportes para trabajo futuro, la razón de esto se debe a que no se cuenta con la suficiente información para realizar este módulo, por lo cual se llegó a la conclusión con el cliente que la mejor decisión es aplazarla.

El sprint número ocho empieza el día 7 de agosto del 2023 y termina el 18 de agosto del 2023. Se realizó el consumo de los servicios que ya se encontraban implementados (Anexo 15 - Figura 38). Donde por parte del backend se habilitaron los respectivos CORS de cada uno de los servicios para su respectivo consumo. Así mismo se desarrolló el CRUD de la Tabla Tarjeta y Comentarios, se implementaron las Lambdas y su configuración del API Gateway, aparte se realizó un cambio para poder implementar una consulta paginada en SQL (ver Anexo 16 - Figura 49 y Figura 50). Finalmente, para la sección de Comentarios, se realizó la configuración de los métodos de cargue y descargue de Documentos, donde su almacenamiento se realiza en el servicio de S3 (ver Anexo 16 - Figura 48 y Figura 49) y se testeó el código de las Lambdas (ver Anexo 16 - Figura 50 hasta la Figura 51) donde para el cargue y descargue del documento se usó codificación en base64 y se configuró las políticas dentro de AWS para el uso del Bucket de S3 que almacena los documentos. Para el Frontend, se realizó el consumo para el módulo de *Casos*, en el cual se realizó el consumo de los datos del API para disponerlas en la tabla de este módulo (ver Anexo 15 - Figura 38), una vez realizado este consumo, se realizó el método para agregar un caso por medio de un formulario, este formulario se mejoró seccionando por información del cliente y el deudor, una vez se llenen los datos obligatorios, se envía esta información al backend (ver Anexo 15 - Figura 39), paso seguido se realizó el método de editar, el cual trae los datos del caso a editar, una vez los datos hayan sido modificados se envía al backend para que puedan ser actualizados (ver Anexo 15 - Figura 40), por último, se implementó el método eliminar caso, por medio de un aviso de confirmación que envía al backend la acción de eliminar el caso especificado (ver Anexo 15 - Figura 41).

El sprint número nueve empieza el día 20 de Agosto del 2023 y termina el 4 de Septiembre del 2023, se otorga como Sprint Goal la realización de una consulta paginada para el manejo de las tarjetas y el seguimiento de los casos. La consulta paginada permite manejar el número de registros que se solicitan a la base de datos, permitiendo al frontend optimizar la petición de los datos y su visualización (ver Anexo 16 - Figura 52). Una vez finalizada la implementación de esta consulta paginada y la disposición de la información por medio de un API, se realizó el consumo de la información en el Frontend del Módulo de Seguimiento, en el cual se realizó el consumo de la información de las tarjetas en su debido estado, si es el estado *Creación Demanda* se mostrará las tarjetas relacionadas a ese estado, en el caso contrario que sea *Subir Demanda Plataforma Judicial*, se visualizarán las tarjetas relacionadas a dicho caso, a la vez, se puede visualizar un pequeño cuadro en el cual

contiene información relacionada al caso que se le desea hacer el seguimiento, esto con el fin de que el usuario tenga presente a cuál caso le está realizando el seguimiento (ver Anexo 16 - Figura 53). Una vez que se haga el consumo de las tarjetas de cada caso, se añade la funcionalidad de *Agregar Actualización Caso* en las cuales el usuario le podrá colocar un título a la tarjeta, y el estado al que va ligado, dependiendo del estado que seleccione, se actualizará el estado del caso (ver Anexo 16 - Figura 54). Al tener la tarjeta creada, se tiene la posibilidad de agregar un documento, el cual se convierte a base64 y se envía al backend para su respectivo almacenamiento (ver Anexo 16 - Figura 55), una vez subido se podrá visualizar en una pequeña tabla el documento subido, si se desea eliminar, se le da click al botón "Eliminar" de la sección *Documentos*, el cual se desplegará una ventana de confirmación preguntándole al usuario si desea eliminar el caso (ver Anexo 16 - Figura 56). Para la sección de *Documentos*, está la posibilidad de que el usuario digite un comentario en dicha sección, si se desea editar el comentario, se le da click a *Editar Comentario*, el cual se desplegará una ventana con el comentario para editar, una vez editado se confirma la misma y se envía al backend para ser sobrescrito (ver Anexo 16 - Figura 57). Si se desea eliminar el comentario, se le da click al botón *Eliminar Comentario*, el cual se despliega una ventana de confirmación para que el usuario de su aprobación para que el comentario sea eliminado (ver Anexo 16 - Figura 58). Por último, si se desea eliminar la tarjeta, se da click al botón *Eliminar Tarjeta* el cual desplegará una ventana solicitando la confirmación del usuario, si el usuario acepta, se eliminará la tarjeta en conjunto con los comentarios y los documentos (ver Anexo 16 - Figura 58).

6.4 Pruebas Académicas

La tercera épica a desarrollar corresponde al desarrollo de pruebas académicas. Esta etapa inicia desde el 12 de septiembre hasta el 10 de octubre del 2023, orientada a cumplir con el objetivo específico número 4 (ver apartado 4 - Objetivos específicos 4). Para las pruebas académicas se realizaron de tres tipos. Primero pruebas unitarias donde se usaron para probar determinadas lambdas involucradas en el módulo de seguimiento de los casos; de igual forma, para el testeo de las lambdas, se usó la biblioteca *unittest.mock* que permite crear objetos simulados (mocks) los cuales simulan el comportamiento de las unidades de código permitiendo su testeo de forma aislada. El resultado de las pruebas se ven en la tabla de la Figura 15.

Lambda	Método	Registro	Resultado
consult_case	GET	ID:10	OK - 0.006S
consult_comentario	GET	ID:39	OK - 0.004S
consult_tarjeta	GET	ID:4	OK - 0.004S
Insert case	POST	ID:25	OK - 0.002S

Insert Comment	POST	ID:40	OK - 0.005S
Insert Tarjeta	POST	ID:9	OK - 0.003S
update_case	PUT	ID:23	OK - 0.006S
update_Tarjeta	PUT	ID:23	OK - 0.006S
update_comment	PUT	ID:16	OK - 0.004S
delete_case	DELETE	ID:45	OK - 0.003S
delete_comment	DELETE	ID:20	OK - 0.003S
delete_tarjeta	DELETE	ID:12	OK - 0.004S

Figura No. 15. Pruebas unitarias realizadas a las lambdas del módulo de seguimiento de casos.

Fuente: Elaboración propia

Segundo se continuó con las pruebas de rendimiento. Donde se tomaron métricas de las distintas Lambdas el API Gateway y de la base de datos, montadas en AWS. Para esto, se usó la herramienta de Cloudwatch, la cual permite recopilar y generar métricas de los distintos servicios usados en AWS.

Periodo medición	Lambda	Max Duración (ms)	Min Duración(ms)	Promedio (ms)	Invocaciones
1 semana	consult_case	20.737	2.05	5.3714	91
1 semana	update_case	16.5	3.77	8.1452	28
1 semana	delete_case	42.93	3.845	9.8187	23
1 semana	consultaPaginada	398.83	2.93	19.576	55
1 semana	UploadFile	2'855	210.62	1'948	30
1 semana	DownloadFile	2'299	91.57	1'262	30
1 semana	Delete_file	2'581	282.85	1'781	29
1 semana	insert_comentario	408.8	3.97	77.989	49

1 semana	update_comentario	457.03	4.08	67.558	34
1 semana	delete_comentario	388.88	4.41	106.92	23
1 semana	insert_tarjeta	22.11	4.06	9.873	26
1 semana	delete_tarjeta	61.45	5.17	17.046	24
1 semana	ConsultState	18.81	1.63	2.82	120

Figura No. 16. Pruebas rendimiento lambdas

Fuente: Elaboración propia

De las variables de las Lambdas, se tomó la duración y número de invocaciones, principalmente, ya que estas variables están involucradas directamente en el costo del servicio. Como se puede ver en los resultados, las Lambdas que más tiempo de duración registraron fueron aquellas relacionadas al manejo de carga (ver figura 17), descarga y eliminación de documentos, donde el registro máximo registrado fue en la Lambda de cargue con 2'855 milisegundos.

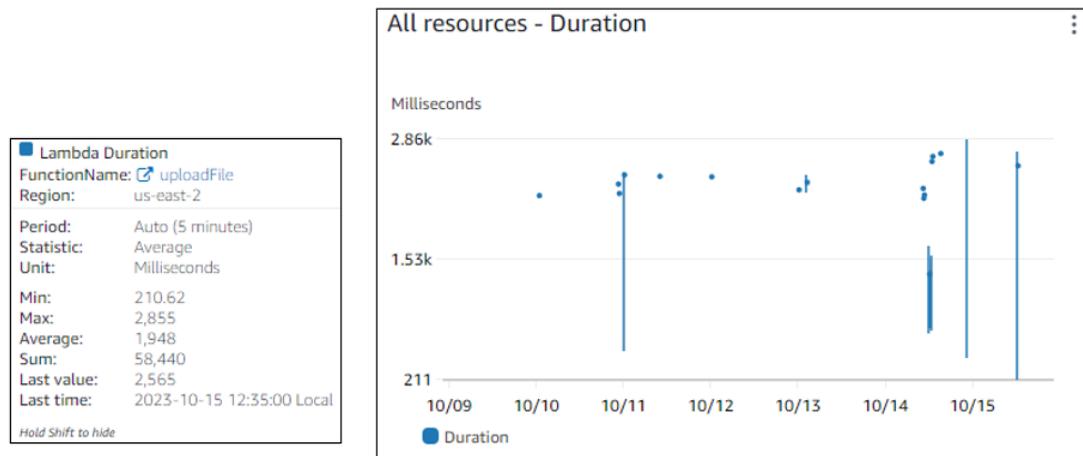


Figura No. 17. Duración Lambda cargue de documento

Fuente: Elaboración propia

Por último se realizaron pruebas de caja negra donde en la figura 17 se muestra las funcionalidades y módulos que se testean(ver figura 18).

Componente a Testear	Actividad de prueba	Tipo de Prueba
Módulo de Gestión de	Agregar un nuevo Caso	Caja negra

Casos	Editar un Caso	
	Eliminar un Caso	
Módulo de Seguimiento de Casos	Agregar una tarjeta	Caja negra
	Subir un documento	
	Descargar documentos subidos	
	Eliminar un documento	
	Crear un comentario	
	Editar un comentario	
	Eliminar un comentario	

Figura No. 18. Aplicaciones prueba caja negra

Fuente: Elaboración propia

6.5 Pruebas Estáticas (Aceptación)

Para las pruebas de aceptación, se continuó con la metodología de Alpha Testing, Beta Testing y User Acceptance Test. Se realizó la medición de un cuestionario que evalúa la navegación, contenido, accesibilidad, seguimiento y fácil entendimiento del software a evaluar. Dicha evaluación se llevó a cabo realizando 10 afirmaciones las cuales tienen cinco opciones de respuesta acorde a la Escala Likert, se encuentran las siguientes opciones: Totalmente De acuerdo (SA), De acuerdo (A), Neutro (N), Desacuerdo (D), Totalmente en Desacuerdo (DS); estas puntúan del 1 al 5 siendo Totalmente en Desacuerdo un valor de uno y Totalmente De acuerdo un valor de cinco. A continuación, en la figura 20 se muestran los resultados obtenidos del cuestionario dando un porcentaje de las respuestas obtenidas en cada una de las afirmaciones.

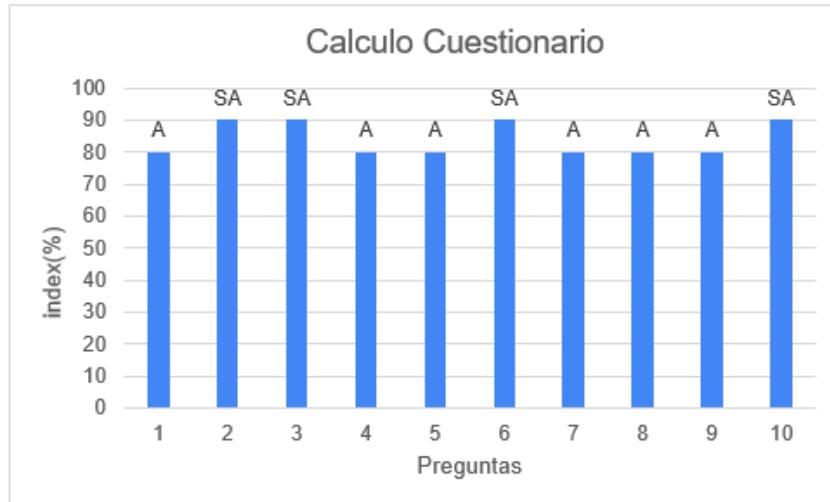


Figura No. 20. Resultados del cuestionario de evaluación mediante escala Likert
Fuente: Elaboración propia

De acuerdo a los resultados obtenidos se puede concluir que, se obtuvo un nivel de aceptación del 84% por parte del usuario lo cual es óptimo, y se toma como un buen resultado. Lo anterior indica que el software desarrollado cumple con los criterios de navegación, contenido, accesibilidad, seguimiento y fácil entendimiento, planteados para estas pruebas de aceptación.

6.6 Pruebas Dinámicas (UEQ KPI)

Para la realización de las pruebas dinámicas, se le aplicó una encuesta que comprende de 26 preguntas dirigidas hacia el UEQ, adicional a ello, se realizó una encuesta que complementa a la de UEQ que es la de KPI, que consta de 6 preguntas. Estas preguntas se miden por medio de una escala Likert, donde, dependiendo de la pregunta, responde con la afinidad que lo represente, por ejemplo, está la pregunta de “Un sistema debe ser fácil/difícil” si la persona siente mayor afinidad con *fácil*, contesta con un valor cercano a la izquierda. La prueba se realizó en tres momentos, la primera fue evaluando su sistema actual que es Google Drive, la segunda es con la primera entrega de valor del artefacto, y la tercera es la segunda entrega de valor. A continuación se mostrará los resultados de cada una:

6.6.1 Aplicando UEQ KPI en Google Drive

Se ingresaron las respuestas del Gerente del Grupo Montserrat en un Excel que provee los investigadores del UEQ KPI:

Description of data set 1: Data Set 1																										
Items																										
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
5	6	5	5	5	5	5	5	6	7	5	3	5	4	2	4	5	5	5	5	5	5	4	5	5	3	

Figura No. 21. Datos ingresados en la herramienta 'Data Analysis Tools'

Una vez agregados, se dirige a la pestaña de *Benchmark* la cual aporta los siguientes resultados:

Scale	Mean	Comparisson to benchmark	Interpretation
Atracción	0,00	Bad	In the range of the 25% worst results
Transparencia	0,25	Bad	In the range of the 25% worst results
Eficiencia	0,00	Bad	In the range of the 25% worst results
Controlabilidad	0,00	Bad	In the range of the 25% worst results
Estimulación	0,00	Bad	In the range of the 25% worst results
Novedad	-1,75	Bad	In the range of the 25% worst results

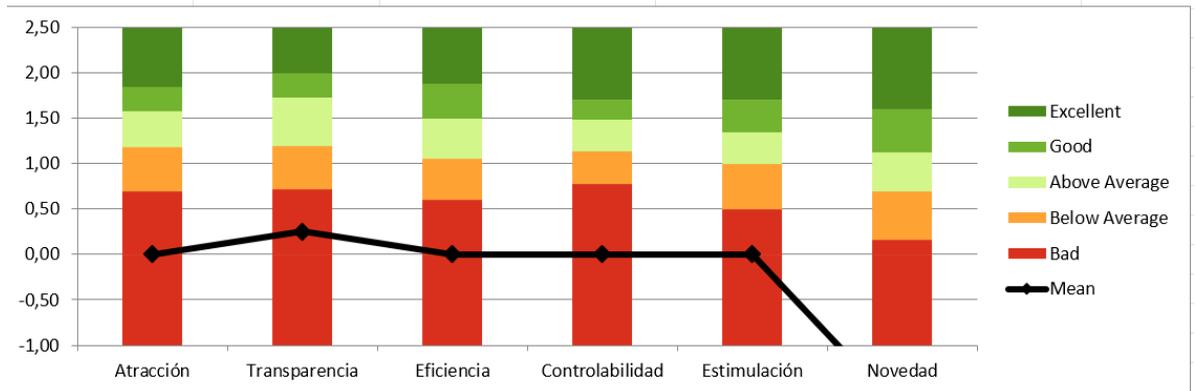


Figura No. 22. Resultados de la encuesta UEQ sobre Google Drive

La Figura No. 22 revela que Google Drive sacó una mala puntuación en las 6 dimensiones que evalúa el UEQ, de manera gráfica se puede apreciar el resultado bajo que se obtuvo.

Paso seguido, para poder evaluar la variable de transformación, es necesario utilizar la herramienta Excel para calcular los anteriores resultados en un único KPI. Para ello se realizó otra encuesta que adiciona 6 preguntas de User Experience, donde se registran las respuestas del beneficiario en el mismo Excel en el que se realizó el cálculo del UEQ, para ello se realizó lo siguiente:

concerning the importance of the scales below!						
Transparencia	Eficiencia	Controlabilidad	Estimulación	Novedad	SUM	
7	7	7	7	7	42	

Figura No. 23. Resultados de la encuesta KPI sobre Google Drive

Una vez ingresados los resultados de la encuesta en la herramienta de Excel, la herramienta procede a realizar el siguiente cálculo:

Calculated relative importance							
Atracción	Transparencia	Eficiencia	Controlabilidad	Estimulación	Novedad	KPI	
0,17	0,17	0,17	0,17	0,17	0,17	-0,25	

Figura No. 24. Operaciones realizadas para obtener el KPI de Google Drive

Una vez que la herramienta realiza los cálculos, se obtiene que el valor de la Experiencia de Usuario del beneficiario en Google Drive, es de **-0,25**, [1] ofrece la siguiente escala para entender en qué posición se encuentra el KPI:

UEQ KPI value range between -0.286 and 2.143

Según la escala anterior, se define que si el KPI está entre los dos valores anteriormente mostrados, está dentro del rango, en este caso, se encuentra por fuera del mismo.

Como un dato adicional que brinda esta herramienta al momento de medir el KPI, realiza un gráfico resaltando las dimensiones más importantes para el cliente, a continuación se dejará esta gráfica:

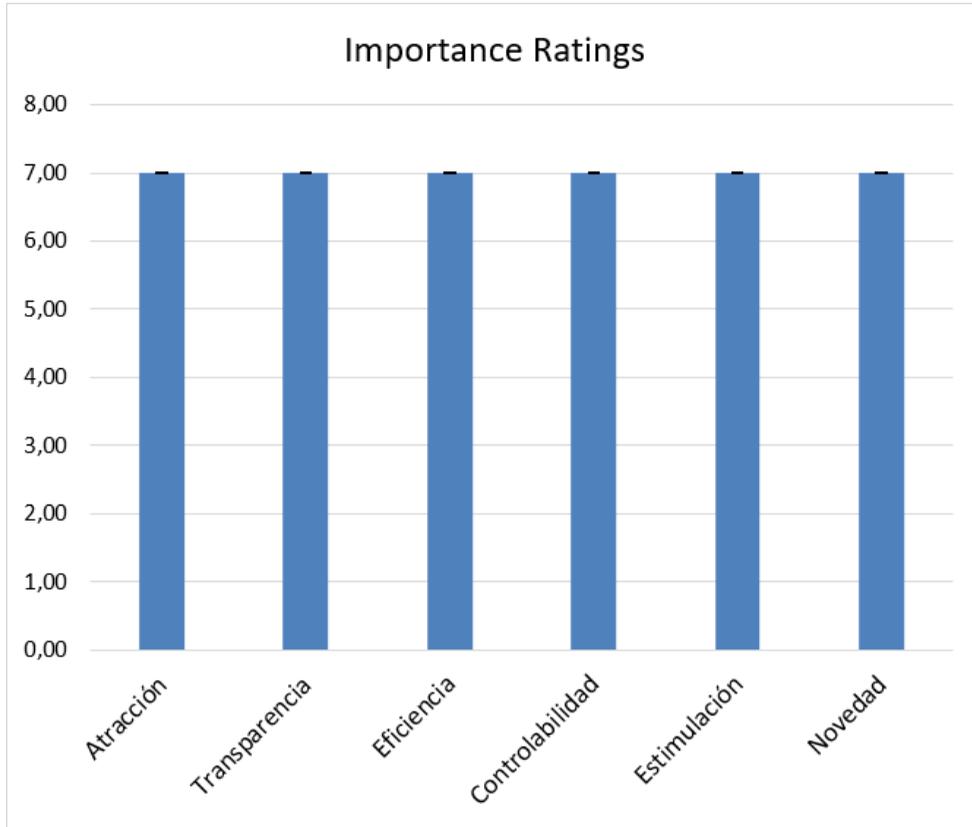


Figura No. 25. Gráfica que muestra la importancia de cada área del beneficiario

Según la anterior gráfica que se basa en las respuestas del cuestionario KPI, el beneficiario evidencia una afinidad a todas las áreas del UX, por lo tanto para el artefacto lo ideal es cubrir esas 6 dimensiones.

6.6.2 Aplicando UEQ KPI en la primera entrega del artefacto

En un segundo momento, se realiza la misma prueba, ahora con la primera entrega del artefacto, aquí el usuario previamente interactuó con el mismo, posteriormente se dedicó a diligenciar el cuestionario UEQ, donde se ingresaron los siguientes resultados:

Items																									
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
7	7	2	1	2	5	6	6	2	1	6	1	7	6	7	6	1	2	2	6	1	6	1	2	2	6

Figura No. 26. Inserción de los datos en la herramienta de Excel UEQ para ser calculados

Como se puede apreciar en la Figura No. 26, se agregaron los resultados de la encuesta practicada al beneficiario, ahora se dirige a la sección de *Benchmark* para obtener de manera gráficas los resultados del UEQ, que se mostrarán a continuación:

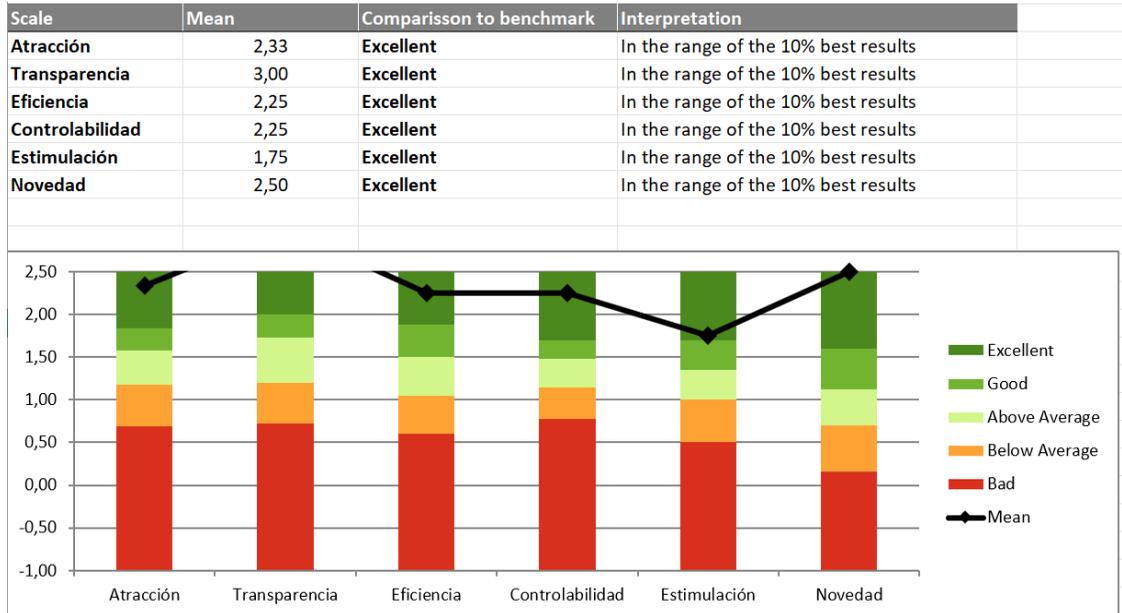


Figura No. 27. Inserción de los datos en la herramienta de Excel UEQ para ser calculados

Una vez calculados los resultados del UEQ, se puede visualizar que el artefacto abarca el valor máximo en todas las áreas, a continuación se utilizan los resultados del KPI realizado anteriormente para calcular los valores obtenidos en el UEQ de la primera entrega del artefacto, a continuación se mostrará el resultado obtenido:

Calculated relative importance							KPI
Atracción	Transparencia	Eficiencia	Controlabilidad	Estimulación	Novedad		
0,17	0,17	0,17	0,17	0,17	0,17		2,35

Figura No. 27. Inserción de los datos en la herramienta de Excel UEQ para ser calculados

Como se puede apreciar en la Figura No. 27, se obtuvo como resultado un valor KPI de **2,35**, si se sigue la escala anteriormente mencionada, se puede visualizar que la primera entrega del artefacto supera el rango estimado.

6.6.2 Aplicando UEQ KPI en la segunda entrega del artefacto

En este tercer momento, se realiza una segunda entrega, en donde se realizaron pequeños cambios como el agregar más información a las tablas, y mostrar la información del caso en el seguimiento. Se volverá a aplicar la misma encuesta UEQ, a continuación se mostrará la inserción de los resultados en la herramienta de Excel:

Items																									
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
6	7	2	1	2	5	6	5	1	2	6	2	7	5	6	6	1	2	2	6	1	6	1	2	2	5

Figura No. 28. Inserción de los datos en la herramienta de Excel UEQ para ser calculados

A continuación se mostrará la sección de *Benchmark* donde se ilustrará los resultados de la encuesta UEQ:

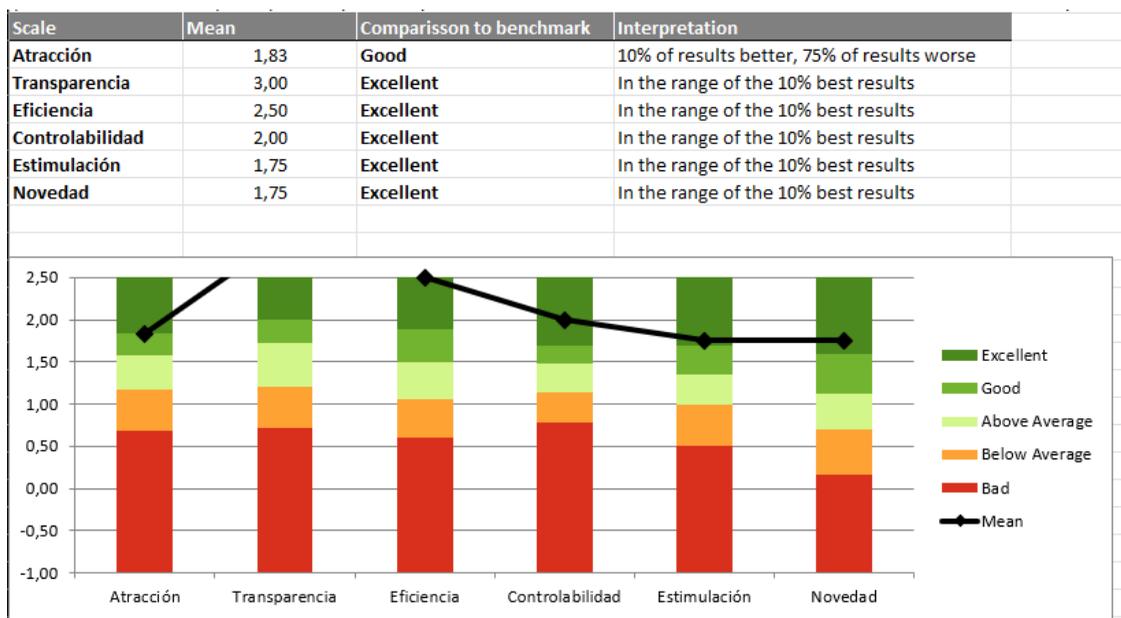


Figura No. 29. Resultados del cálculo de los valores del UEQ de la tercera entrega del artefacto

Se puede apreciar en la Figura No. 29. que la *Atracción* bajó un poco, pero esto no afecta en gran medida la percepción de Experiencia de Usuario del beneficiario. A continuación se mostrará el cálculo del KPI para esta tercera entrega:

Atracción	Transparencia	Eficiencia	Controlabilidad	Estimulación	Novedad	KPI
0,17	0,17	0,17	0,17	0,17	0,17	2,14

Figura No. 30. Resultados del cálculo de los valores del KPI de la tercera entrega del artefacto

Como se puede apreciar en la Figura No. 30, el resultado del KPI es de **2,14**, varió solo *0,21*, lo cual sigue estando en el máximo de la escala.

7. Conclusiones

Al momento de llevar a cabo la épica de análisis del contexto, se pudo determinar que, la técnica de historias de usuario facilitaron la comprensión del contexto bajo el cual se encuentra la organización Grupo Montserrat, esto permitió la definición de los requerimientos para el sistema de información. Dichos requerimientos permitieron delimitar el alcance del proyecto, generando así una alineación directa con el Grupo Montserrat para que se tenga conocimiento del desarrollo actual y el alcanzado hasta la fecha de cierre. Esto anterior se alinea con el objetivo específico número 1, lo cual permite tener el cumplimiento del mismo.

De acuerdo a los resultados de aceptación obtenidos de las pruebas realizadas con el Grupo Montserrat llevada a cabo en la épica de pruebas estáticas, los resultados de las mismas indican que la arquitectura definida para el sistema de información fue la adecuada para el desarrollo e implementación del módulo de gestión y seguimiento de casos, dando así, continuación del proyecto a futuro con el cliente. El excelente resultado obtenido en estas pruebas permitieron cumplir con éxito el objetivo específico número 2.

En la épica de pruebas estáticas se realizó en forma paralela la prueba de aceptación por medio de un cuestionario según la metodología *Alpha Testing, Beta Testing y User Acceptance Test*, que provee ciertos criterios de experiencia de usuario puntuados bajo la escala likert. Al evaluar el sistema de información desarrollado, se obtuvo como resultado un índice de aceptación del 84%, lo cual indica que el sistema posee una buena navegación, contenido, accesibilidad, seguimiento y fácil entendimiento. Esto da a entender que el buen resultado obtenido permite dar el cumplimiento del objetivo específico número 3.

Por último, se realizó el cuestionario UEQ KPI en la épica de pruebas dinámicas, el cual permitió analizar la percepción de la experiencia del usuario por medio de seis dimensiones. El cliente diligenció este cuestionario enfocado a la herramienta de Google Drive, la cual obtuvo un resultado de *-0,25*, indicando que dicha herramienta no cumple con las expectativas del cliente según los criterios KPI. Sin embargo, cuando el cliente evaluó el sistema de información desarrollado, se obtuvo un resultado de **2,14**, lo cual representa un cambio sustancial en el contexto. Las pruebas UEQ KPI reflejaron la gran transformación que generó el software desarrollado a comparación con el sistema Google Drive, lo cual permite dar como cumplido con éxito el objetivo específico número 4.

8. Lecciones Aprendidas y Trabajo Futuro

Al recabar información de la organización y de su contexto se encontró que el proyecto era más grande de lo que se había previsto. Por lo tanto se realizó un acuerdo con el cliente para delimitar el alcance del proyecto al módulo de gestión y seguimiento de los casos. Esto permitió que se desarrollara con mayor calidad estos dos módulos, generando así, un buen punto de partida para continuar con el desarrollo de otros módulos para realizar un sistema de información más completo.

Para trabajo futuro se acordó con el cliente seguir trabajando en el modelo de seguimiento de caso para expandir sus funcionalidades, realizar ajustes al diseño e implementarlo en la página de Grupo Montserrat, lo anterior con el propósito de brindar mejoras en la experiencia de aquellos usuarios que den uso al software propuesto. Se deja abierta la posibilidad de dar continuidad al proyecto en caso de que ambas partes estén de acuerdo y que sea de utilidad para todos los interesados.

Con respecto al sistema de diseño de Frontend, se obtuvo como aprendizaje que no se deben utilizar sistemas de diseño que no provean un buen soporte, en este caso se utilizó *ng-bootstrap*, aunque ahorró tiempo para no diseñar cosas desde cero, si se deseaba implementar herramientas como las notificaciones (llamadas *Toasts* en la documentación oficial), la documentación no es lo suficientemente clara, es por ello, que se aprendió a elegir con mejor criterio herramientas que posean una mejor documentación de la mano con ejemplos que estén alineados a las últimas versiones de Angular.

9. Referencias

- [1] Hinderks, A., Schrepp, M., Domínguez Mayo, F. J., Escalona, M. J., & Thomaschewski, J. (2019). Developing a UX KPI based on the user experience questionnaire. *Computer Standards & Interfaces*, 65, 38–44. doi:10.1016/j.csi.2019.01.007
- [2] R. P. Pérez Salas, "Diseño e implementación de un sistema informático con tecnología web para la gestión de procesos legales en el estudio Jurídico Jiménez Troya S.R.L.," Universidad César Vallejo, 2019, Accessed: Oct. 20, 2022. [Online]. Available: <https://repositorio.ucv.edu.pe/handle/20.500.12692/28999>
- [3] Á. J. Córdova Jaramillo, "Implementación de un web app para registro de orden de trabajo para bufete de abogados," Repositorio Institucional - UTP, 2021, Accessed: Oct. 20, 2022. [Online]. Available: <https://hdl.handle.net/20.500.12867/5100>
- [4] L. E. Novoa Enciso and C. A. Calderón Barragán, "Aplicativo para el control y monitoreo de procesos de la agencia Orozco Ocampo Abogados 'CONTROLSOFT,'" repository.unipiloto.edu.co, 2018. <http://repository.unipiloto.edu.co/handle/20.500.12277/6113> (accessed Nov. 17, 2022).

- [5] A. K. Chacón Mata, D. A. Flórez Rincón, L. A. Medina González, C. E. Quimbay Cunalata, and M. J. Ramos Pacheco, "Arquitectura empresarial para la firma de abogados Norton Rose Fulbright, un enfoque BPM-SOA," repository.javeriana.edu.co, Jun. 2018, Accessed: Nov. 17, 2022. [Online]. Available: <http://hdl.handle.net/10554/40914>.
- [6] Andrawos, Mina, and Martin Helmich. Cloud Native Programming with Golang: Develop microservice-based high performance web apps for the cloud with Go. Packt Publishing Ltd, 2017.
- [7] Debrauwer, Laurent. Patrones de diseño en Java: los 23 modelos de diseño: descripciones y soluciones ilustradas en UML 2 et Java. Ediciones ENI., 2018.
- [8] Yellavula, Naren. Hands-On RESTful Web Services with Go: Develop elegant RESTful APIs with Golang for microservices and the cloud. Packt Publishing Ltd, 2020.
- [9] M. Rahikainen, "Web Application in Angular and Ionic," www.theseus.fi, 2021. <https://urn.fi/URN:NBN:fi:amk-2021121626394> (accessed Apr. 09, 2023).
- [10] M. Khan, "Distributed and scalable parsing solution for telecom network data," aaltodoc.aalto.fi, Jan. 2020, Accessed: Apr. 09, 2023. [Online]. Available: <https://aaltodoc.aalto.fi/handle/123456789/42779>
- [11] V. S. Venkata Murali Krishna, T. S. Rama Vamsi, and S. Kavitha, "Automation of Forensic Analysis for AWS Aurora using EventBridge and Athena," 2022 6th International Conference on Electronics, Communication and Aerospace Technology, Dec. 2022, doi: <https://doi.org/10.1109/iceca55336.2022.10009126>.
- [12] K. E. Jiménez Arellano and L. N. Idrovo Román, "DESARROLLO DE UNA APLICACIÓN WEB MEDIANTE LA METODOLOGIA DE DESARROLLO EN CASCADA PARA AUTOMATIZAR LA GESTION ODONTOLOGICA DEL CANTON MILAGRO," Repositorio de la Universidad Estatal de Milagro, Jun. 2022, Accessed: Apr. 09, 2023. [Online]. Available: <http://repositorio.unemi.edu.ec/xmlui/handle/123456789/6357>
- [13] T. Uotila, "Designing scalable microservices : Case: AWS with Python," www.theseus.fi, 2019. <https://urn.fi/URN:NBN:fi:amk-201905139574> (accessed May 06, 2023).
- [14] B. J. Oates, *Researching information systems and computing*. Los Angeles: Sage, 2006.
- [15] Z. Peng, X. Lin, and N. Niu, "Unit Tests of Scientific Software: A Study on SWMM," Lecture Notes in Computer Science, vol. 12143, pp. 413–427, Jun. 2020, doi: https://doi.org/10.1007/978-3-030-50436-6_30.
- [16] S. Lukasczyk, F. Kroiß, and G. Fraser, "Automated Unit Test Generation for

Python,” arXiv:2007.14049 [cs], vol. 12420, pp. 9–24, Jun. 2020, doi: https://doi.org/10.1007/978-3-030-59762-7_2.

[17] I. Afrianto, A. Finandhita, A. Heryandi, and S. Atin, “User Acceptance Test For Digital Signature Application In Academic Domain To Support The Covid-19 Work From Home Program,” *International Journal of Information System & Technology*, vol. Vol. 5, No. 3, Oct. 2021, Available: https://www.researchgate.net/profile/Irawan-Afrianto/publication/355584483_User_Acceptance_Test_For_Digital_Signature_Application_In_Academic_Domain_To_Support_The_Covid-19_Work_From_Home_Program/links/617794880be8ec17a9305870/User-Acceptance-Test-For-Digital-Signature-Application-In-Academic-Domain-To-Support-The-Covid-19-Work-From-Home-Program.pdf

[18] A. Hinderks, M. Schrepp, and J. Thomaschewski, “User Experience Questionnaire (UEQ),” *Ueq-online.org*. <https://www.ueq-online.org/>

[19] S. Calzavara, R. Focardi, M. Nemeč, A. Rabitti, and M. Squarcina, “Postcards from the Post-HTTP World: Amplification of HTTPS Vulnerabilities in the Web Ecosystem,” *IEEE Xplore*, Sep. 16, 2019. <https://ieeexplore.ieee.org/document/8835223> (accessed Sep. 04, 2023).

[20] A. Vázquez-Ingelmo, A. García-Holgado, and F. J. García-Peñalvo, “C4 model in a Software Engineering subject to ease the comprehension of UML and the software,” *IEEE Xplore*, Apr. 01, 2020. <https://ieeexplore.ieee.org/abstract/document/9125335> (accessed Jun. 22, 2023).

Anexos

A continuación se encontrará el [documento de anexos](#) para visualizar figuras, gráficas, entre otros.

A. Anexo No. 1: Diagrama del Modelo Biopsicosocial desde la perspectiva del Problema

Se encontrará a continuación el Diagrama del Modelo Biopsicosocial, donde se analiza la perspectiva desde el *Problema* identificado de la empresa Grupo Montserrat

B. Anexo No. 2: Diagrama del Modelo Biopsicosocial desde la perspectiva de la Solución

Se encontrará a continuación el Diagrama del Modelo Biopsicosocial, donde se realiza un análisis para dar Solución a los Problemas encontrados.

C. Anexo No. 3: Modelo BPMN AS - IS del Grupo Montserrat

Se encontrará a continuación el Modelo BPMN AS - IS del Grupo Montserrat, donde se describe el estado actual de la empresa antes de implementar la solución.

D. Anexo No. 4: Modelo BPMN TO - BE del Grupo Montserrat

Se encontrará a continuación el Modelo BPMN TO - BE del Grupo Montserrat, donde se describe el estado actual de la empresa después de implementar la solución.

E. Anexo No. 5: Tablero de Control

A continuación se puede apreciar el [Tablero de Control](#), en el cual se encuentran listadas las tareas a realizar en cada ciclo de trabajo respectivo.

F. Anexo No. 6: Metodología

En este anexo, se visualiza de manera gráfica la metodología usada para el desarrollo del artefacto.

G. Anexo No. 7: Diagrama C4

En este anexo se encuentra el Diagrama C4 utilizado para brindar de manera general una visión de las funcionalidades del artefacto por cada usuario.

H. Anexo No. 8: Diagrama Arquitectura AWS

A continuación se visualiza el diagrama realizado para diseñar la arquitectura en nube de AWS que tendrá el Grupo Montserrat.

I. Anexo No. 9: Tabla costos - Productos AWS

En el siguiente anexo, se realizó una tabla de costos por medio del software Microsoft Excel, con el fin de enlistar todas las herramientas que se usarán en el desarrollo de la solución y calcular así el valor total del uso de las mismas.

J. Anexo No. 10: Diagramas Base de Datos

En este anexo se ilustran los diagramas de las bases de datos que se implementarán en el artefacto.

K. Anexo No. 11: Actas Reuniones Director

A continuación se mostrarán las actas de las reuniones con el director para dejar constancia de lo pactado en dichas reuniones, con el fin de llevar un seguimiento del avance del desarrollo tecnológico.

L. Anexo No. 12: Actas Reuniones Grupo Montserrat

En este anexo se encontrarán las actas de las reuniones realizadas con el cliente, con el fin de dejar una constancia de lo pactado en las reuniones, a la vez, de poder llevar un seguimiento del avance del proyecto.

M. Anexo No. 13: Configuraciones AWS

A continuación se ilustran las configuraciones necesarias para llevar a cabo la configuración de los servicios de AWS para su correcto levantamiento y funcionamiento.

N. Anexo No. 14: Configuraciones CRUD tablas caso, usuario y credenciales AWS

En este anexo se encontrarán las configuraciones realizadas en AWS sobre el CRUD de las tablas de caso, usuario y credenciales.

O. Anexo No. 15: Maquetación de los mockups en Angular y CRUD tablas tarjeta, documento y comentario

En este anexo se puede visualizar las maquetaciones realizadas a partir de los

mockups, además, del consumo del API de Casos en el módulo de casos en el Frontend.

P. Anexo No. 16: CRUD tabla tarjeta, comentarios y documentos

En este anexo se puede visualizar la creación del CRUD para la tabla tarjeta, comentarios y documentos en AWS. A la vez, se visualiza el consumo de estos CRUD en el módulo de seguimiento desde el Frontend.