Aplicación móvil para el seguimiento emocional de la comunidad de la Universidad Distrital Francisco José de caldas denominada OurHive – Emotional Diary

Daniel García Romero

Yohan Nicolás Olmos Pérez

Universidad Distrital Francisco José de Caldas

Facultad Tecnológica

Tecnología en Sistematización de Datos

Bogotá DC

2024

Aplicación móvil para el seguimiento emocional de la comunidad de la Universidad Distrital Francisco José de caldas denominada OurHive – Emotional Diary

Daniel García Romero - 20202578089

Yohan Nicolás Olmos Pérez - 20211578055

Tutor:
M.Sc. Jorge Eduardo Hernández Rodríguez

Universidad Distrital Francisco José de Caldas
Facultad tecnológica
Tecnología en Sistematización de Datos
Bogotá DC
2024

Tabla de Contenido

Resumen 8

Abstract 9

Introducción 10

1. Planteamiento del problema 11

1.1. Descripción del problema 11

1.2. Formulación del problema 13

2. Objetivos 13

2.1. Objetivo General 13

2.2. Objetivos Específicos 13

3. Justificación 13

4. Marco Teórico 15

4.1. Marco Referencial 15

4.1.1. Inteligencia Emocional 15

4.1.2. Emociones 15

4.1.3. Autoconsciencia 15

4.1.4. Autogestión 15

4.1.5. Empatía 15

4.1.6. Habilidades Sociales 16

4.1.7. Alfabetización Emocional 16

4.2. Marco Conceptual 16

4.2.1. PostgreSQL 16

4.2.2. Servidor web: 16

4.2.3. REST API: 16

4.2.4. JavaScript 17

4.2.5. Node.js 17

4.2.6. TypeScript 17

4.2.7. Dart 17

4.2.8. Flutter 17

4.2.9. NestJS 17
4.2.10. Algoritmo SHA-256 18
4.2.11. JWT (JSON Web Token) 18
4.2.12. Disponibilidad 18
4.2.13. Autenticación 18
4.2.14. Go 18
4.3. Antecedentes 19
4.3.1. Mood Tracker 19
4.3.2. Diario de emociones 19
4.3.3. Herramientas Tecnológicas 19
4.3.3.1. MoodPress - Mood Diary Tracker 19
4.3.3.2. Daylio Jornal – Mood Tracker 21
4.3.3.3. DailyBean: Simplest: Jornal 21
4.3.3.4. How We Feel 22
4.3.4. Comparativa 23
5. Marco Metodológico 24
5.1. SCRUMBAN (SCRUM-KANBAN) 24
6. Alcances 25
6.1 Descripción de los Módulos 26
6.1.1. Perfil 26
6.1.2. Autenticación y Autorización 26
6.1.3. Registro Diario Emocional 26
6.1.4. Registro Emociones Trascendentales 27
6.1.5. Historial Emocional 27
6.1.6. Información y Contenido Recomendado 27
6.1.7. Permisos a Usuarios 27
6.1.8. Notificaciones 28
6.2. Diseño de interfaces y Descripción 28
7. Delimitaciones 48
7.1. Delimitación Temporal 48

7.2. Delimitación Geográfica 49
7.3. Delimitación Técnica 49
8. Factibilidad 50
8.1. Factibilidad Económica 50
8.2. Factibilidad Técnica 53
8.3. Factibilidad Legal 54
9. Cronograma 55
Requerimientos Funcionales y Casos de uso 61
Diagramas de secuencia de alto nivel 82
Ver perfil 83
Autenticación Local 83
Registro 84
Deshabilitar Cuenta 85
Ver Historial 85
Visualizar Líneas de Atención 86
Visualizar My Space 87
Visualizar imágenes para My Space 87
Visualizar imágenes para My Space 88
Crear imagen a través de IA.. 88
Crear imagen a través de IA. 89
Crear Líneas de Atención 90
Crear Registro Diario 91
Crear Registro Transcendental 92
Crear Registro Transcendental 92
Diagrama de Arquitectura de Alto Nivel 93
Diagrama de Componentes: 94
Introducción guía de instalación. 94
Frameworks y estándares 95
Descripción del Modelo de Persistencia 95
Requerimientos de interface con otros sistemas 96

Scripts de instalación y ejecución 96
Resultados: 100
Conclusiones 101
10. Referencias 104

Tabla de Imágenes

Imagen 1 Ejemplo Formato JSON 16
Imagen 2 Imagen Proceso de encriptación. 18
Imagen 3 Imagen herramienta tecnológica – MoodPress 20
Imagen 4 Imagen herramienta tecnológica – Mood Tracker. 21
Imagen 5 Imagen herramienta tecnológica – DailyBean. 22
Imagen 6 Imagen herramienta tecnológica – How we feel. 23
Imagen 7 Pantalla Inicio de Sesión. 29
Imagen 8 Pantallas de Registro. 31
Imagen 9 Pantalla Perfil de Usuario 31
Imagen 10 Pantallas de Registro Emocional. 34
Imagen 11 Pantalla Registro de emoción primaria. 36
Imagen 12 Pantalla de Registro Emoción Secundaria. 38
Imagen 13 Pantalla de Formulario Diario Emocional. 39
Imagen 14 Pantalla de Formulario Registro Emoción Trascendental. 40
Imagen 15 Pantalla del Historial Emocional. 43
Imagen 16 Pantalla Detalle Registro Diario. 44
Imagen 17 Pantalla Registro Trascendental. 44
Imagen 18 Pantalla de Información. 45
Imagen 19 Pantalla de My Space 47
Imagen 20 Pantallas de Red de apoyo. 48
Imagen 21 Imagen diagrama de clases – Modulo de autenticación (NestJS) 56
Imagen 22 Imagen diagrama de clases – Modulo de emociones (NestJS) 57
Imagen 23 Imagen diagrama de clases – Modulo de registro de emociones (NestJS) 58
Imagen 24 Imagen diagrama de clases – Modulo de registro de emociones (NestJS) 59
Imagen 25 Imagen diagrama de clases – Modulo Backend Administrativo (Go). . 60
Imagen 26 Imagen diagrama de secuencia – Ver perfil. 83
Imagen 27 Imagen diagrama de secuencia – Autenticación Local 83
Imagen 28 Imagen diagrama de secuencia – Registro. 84
Imagen 29 Imagen diagrama de secuencia – Deshabilitar cuenta. 85

Imagen 30 Imagen diagrama de secuencia – Ver historial..... 85
Imagen 31 Imagen diagrama de secuencia – Visualizar Líneas de Atención. 86
Imagen 32 Imagen diagrama de secuencia – Visualizar My Space..... 87
Imagen 33 Imagen diagrama de secuencia – Validar img My Space. 87
Imagen 34 Imagen diagrama de secuencia – Visualizar contenido recomendación. 88
Imagen 35 Imagen diagrama de secuencia – Imagen con IA 88
Imagen 36 Imagen diagrama de secuencia – Subir imágenes para My Space..... 89
Imagen 37 Imagen diagrama de secuencia – Crear Líneas de atención. 90
Imagen 38 Imagen diagrama de secuencia – Crear registro diario. 91
Imagen 39 Imagen diagrama de secuencia – Crear registro transcendental. 92
Imagen 40 Imagen diagrama de arquitectura de Alto Nivel..... 93
Imagen 41 Imagen diagrama de Componentes..... 94

Resumen

La salud mental se torna vital para la calidad de vida, y la inteligencia emocional surge como una herramienta esencial para su gestión, el documento busca impulsar la alfabetización emocional y el seguimiento de emociones en respuesta a la creciente preocupación por la salud mental, especialmente después de la pandemia del COVID.

Destacando la trascendencia de la inteligencia emocional, particularmente entre los jóvenes, para comprender, gestionar y emplear las emociones de manera efectiva, este documento se concentra en el desarrollo de una aplicación móvil para la Universidad Distrital Francisco José de Caldas. La preocupación por la salud mental, agravada durante la pandemia, resalta la necesidad de impulsar la alfabetización emocional y el seguimiento de las emociones.

Basado en lo mencionado antes, el documento plantea la pregunta fundamental: ¿Cómo abordar la gestión de emociones a través de una aplicación móvil? Los objetivos se centran en la promoción de la inteligencia emocional y la comprensión de las emociones.

Abstract

Mental health becomes vital for quality of life, and emotional intelligence emerges as an essential tool for its management. This document aims to promote emotional literacy and the tracking of emotions in response to the growing concern for mental health, particularly following the COVID pandemic.

Highlighting the significance of emotional intelligence, especially among young individuals, in comprehending, managing, and effectively employing emotions, this document focuses on developing a mobile application for the Universidad Distrital Francisco José de Caldas. The concern for mental health, exacerbated during the pandemic, underscores the need to promote emotional literacy and emotional tracking.

Building upon the above, the document poses the fundamental question: How to address emotion management through a mobile application? The objectives are centered around promoting emotional intelligence and understanding emotions.

Introducción

La salud mental es un componente esencial del bienestar emocional, psicológico y social de las personas. En una sociedad donde la salud mental se ha vuelto un aspecto crítico de la calidad de vida, es fundamental abordar los desafíos emocionales que enfrentamos.

Un elemento crucial en la promoción del bienestar emocional y la gestión de la salud mental es la inteligencia emocional. La inteligencia emocional se ha convertido en una habilidad indispensable para reconocer, comprender y gestionar nuestras propias emociones, así como las de los demás. Su aplicación se extiende a cómo nos relacionamos con los demás, cómo enfrentamos situaciones estresantes y cómo tomamos decisiones importantes en nuestra vida cotidiana. Nos encontramos en una época en la que los celulares y las aplicaciones móviles se han convertido en un componente de nuestra vida diaria, un componente que puede ayudar a mejorar la relación que tenemos con nosotros mismos y nuestras emociones.

El presente documento tiene como objetivo el abordar estos desafíos, proponiendo el desarrollo de una aplicación móvil que busca permitir a la comunidad de la Universidad Distrital Francisco José de Caldas llevar un seguimiento de sus emociones, con el fin de fomentar una alfabetización de su inteligencia emocional.

1. Planteamiento del problema

1.1. Descripción del problema

En un estudio de resiliencia y riesgos de salud mental durante la pandemia en 2020 [27], evaluaron el riesgo de problemas de ansiedad y depresión, evidenciando que el 30.1% de los entrevistados tenían riesgo de padecerlas, esto siendo una cifra tres veces mayor que la medición revelada en 2015. La salud mental, que se refiere al bienestar emocional, psicológico y social de una persona, es un componente fundamental de su calidad de vida. Implica la capacidad de afrontar el estrés, mantener relaciones saludables y tomar decisiones informadas. El aumento de casos de suicidio en Colombia durante el primer trimestre del año 2021, con 709 casos registrados [34], destaca la importancia de contar con las herramientas necesarias para abordar la salud mental.

Hay un elemento vital de la salud mental y es la inteligencia emocional, un concepto acuñado por Daniel Goleman, un renombrado psicólogo y escritor en el campo de la psicología y el comportamiento humano. La inteligencia emocional es esencial para reconocer, comprender y gestionar tanto nuestras propias emociones como las de los demás [35]. Se trata de utilizar esta comprensión para construir relaciones sólidas y saludables, tomar decisiones informadas y resolver conflictos de manera efectiva. En la vida cotidiana, la inteligencia emocional influye en cómo nos relacionamos con las personas, cómo manejamos situaciones estresantes y cómo tomamos decisiones importantes. Esta se convierte en un pilar fundamental para mantener una buena salud mental y para afrontar eficazmente situaciones emocionales adversas, como el aumento de los casos de suicidio.

Sin embargo, la importancia de la inteligencia emocional adquiere un matiz aún más crítico cuando observamos la salud mental de los jóvenes, en un estudio realizado por UNICEF [8], 27% de los jóvenes reportó sentir ansiedad y 15% depresión durante la pandemia. Es que los jóvenes adquieran habilidades que les permitan comprender, utilizar y gestionar sus propios sentimientos de manera efectiva. Y es que esta emerge como un factor fundamental en la gestión de la salud mental. La capacidad de comprender, utilizar y gestionar las emociones de manera efectiva es esencial, especialmente para los jóvenes que se encuentran en una etapa crucial de desarrollo personal y académico. Es vital que las personas puedan identificar y comprender sus patrones emocionales. El seguimiento de sus sentimientos es esencial para abordar los desencadenantes que afectan su bienestar emocional, aunque estos desencadenantes pueden varias ampliamente de una persona a otra, en términos generales, incluye factores como el estrés, cambios en su vida, relaciones interpersonales, salud física, traumas pasados, etc. La falta de apoyo externo ya sea de amigos, familiares o conocidos, también es un desafío. Brindar este apoyo es fundamental para el bienestar emocional de los pacientes.

Además, en un estudio exploratorio de la inteligencia emocional y el abuso doméstico, se encontró que las puntuaciones totales y de subescala del Perfil de Inteligencia Emocional (EQ-i) para ambas muestras se correlacionan negativa y significativamente con las puntuaciones en la Escala de Agresión en Parejas (PAS), lo que sugiere que los déficits en varios componentes de la inteligencia emocional están relacionados con un aumento en la propensión a ser abusivo [38], es por eso que la conciencia emocional, la gestión de las emociones, la empatía y la gestión de las relaciones son habilidades clave que pueden marcar la diferencia en la vida de las personas que enfrentan desafíos de salud mental en Colombia. En la vida cotidiana, todos se enfrentan emociones desbordantes en algún momento. Estas emociones pueden ser de todo tipo, desde la ira y la tristeza, hasta el estrés y el nerviosismo. La clave para lidiar con estas emociones radica en:

- Reconocimiento de emociones: El primer paso para afrontar emociones displacenteras es reconocerlas. A pesar de sentirse abrumados, es importante reconocer la emoción que se está sintiendo.
- Autoconciencia: Comprender por qué se siente de cierta manera es fundamental. Esto permite explorar las razones detrás de una emoción. Por ejemplo, si alguien está estresado en el trabajo, la autoconciencia le ayuda a identificar si se debe a una carga excesiva de trabajo o una relación laboral conflictiva.
- Autocontrol: En lugar de actuar de manera impulsiva, las personas deben pausar y decidir de manera racional el cómo responder frente a la emoción negativa. Esto es especialmente útil cuando se trata de emociones intensas como la ira, donde el autocontrol puede evitar conflictos innecesarios.
- Habilidades Sociales: Las habilidades sociales incluyen la capacidad de manejar relaciones interpersonales de manera efectiva. Estas habilidades involucran trabajar en equipo, colaborar, resolver conflictos, influenciar de manera positiva, comunicarse de manera asertiva y efectiva, entre otros. Las habilidades sociales son esenciales para establecer relaciones saludables y exitosas en la vida personal y profesional. Ayudan a construir un entorno en el que las personas se sientan valoradas y comprendidas.
- Resolución de problemas: En lugar de permitir que las emociones desbordantes sean paralizantes, se debe abordar los problemas subyacentes. Si se está nervioso debido a la carga de trabajo, se debe buscar soluciones, como hablar con un jefe o desarrollar mejores habilidades de gestión del tiempo.

La proporción de herramientas que permitan a estudiantes, jóvenes, etc. enfrentar sus emociones de manera saludable y consciente, se convierte en una necesidad latente en nuestra sociedad. No obstante, es importante subrayar que ninguna

herramienta individual por sí sola será suficiente para abordar este complejo problema.

1.2. Formulación del problema

¿Cómo se puede desarrollar una aplicación móvil que proporcione herramientas para gestionar la inteligencia emocional, brindando a los usuarios un sistema de registro de emociones?

2. Objetivos

2.1. Objetivo General

Desarrollar una aplicación móvil que permita a la comunidad de la Universidad Distrital Francisco José de Caldas llevar un seguimiento de sus emociones con el propósito de fomentar una alfabetización de su inteligencia emocional.

2.2. Objetivos Específicos

- Especificar los requerimientos funcionales y no funcionales, realizando sesiones con el personal de psicología con el fin de generar los criterios de aceptación.
- Crear un módulo de registro emocional, el cual permita al usuario ingresar y distinguir entre emociones primarias y secundarias.
- Implementar un módulo de gestión de líneas de atención que permita a los usuarios conocer las distintas líneas de atención psicológicas que existen en Colombia.
- Crear un módulo de "red de apoyo" cuyo fin es permitir que el usuario pueda compartir su historial emocional con otros usuarios.

3. Justificación

El dominio de habilidades como:

- **Reconocimiento de emociones:** El primer paso para afrontar emociones displacenteras es reconocerlas. A pesar de sentirse abrumados, es importante reconocer la emoción que se está sintiendo.
- **Autoconciencia:** Comprender por qué se siente de cierta manera es fundamental. Esto permite explorar las razones detrás de una emoción. Por ejemplo, si alguien está estresado en el trabajo, la autoconciencia le ayuda a identificar si se debe a una carga excesiva de trabajo o una relación laboral conflictiva.

- **Autocontrol:** En lugar de actuar de manera impulsiva, las personas deben pausar y decidir de manera racional el cómo responder frente a la emoción negativa. Esto es especialmente útil cuando se trata de emociones intensas como la ira, donde el autocontrol puede evitar conflictos innecesarios.
- **Habilidades Sociales:** Las habilidades sociales incluyen la capacidad de manejar relaciones interpersonales de manera efectiva. Estas habilidades involucran trabajar en equipo, colaborar, resolver conflictos, influenciar de manera positiva, comunicarse de manera asertiva y efectiva, entre otros. Las habilidades sociales son esenciales para establecer relaciones saludables y exitosas en la vida personal y profesional. Ayudan a construir un entorno en el que las personas se sientan valoradas y comprendidas.
- **Resolución de problemas:** En lugar de permitir que las emociones desbordantes sean paralizantes, se debe abordar los problemas subyacentes. Si se está nervioso debido a la carga de trabajo, se debe buscar soluciones, como hablar con un jefe o desarrollar mejores habilidades de gestión del tiempo.

Puede permitir a los jóvenes mejorar su desempeño en las actividades cotidianas. Por ejemplo, un estudiante con un alto nivel de Inteligencia Emocional es capaz de comprender las emociones de sus compañeros y otros estudiantes, manejar el estrés de manera efectiva, expresar sus emociones de manera asertiva frente a los demás y contribuir al establecimiento de relaciones estables en su vida. Además, puede brindar apoyo emocional, ya sea alentando o tranquilizando a las personas con las que interactúa. En la BUAP (Benemérita Universidad Autónoma de Puebla), se aplicó un cuestionario de inteligencia emocional a una muestra de 118 estudiantes de diferentes carreras. Los resultados mostraron que existe una correlación positiva y significativa entre la inteligencia emocional y el rendimiento académico, especialmente en las dimensiones de autoconocimiento, autocontrol y empatía, concluyendo que la inteligencia emocional es un factor importante para el éxito académico y profesional de los estudiantes universitarios. El artículo sugiere que se implementen programas de desarrollo de la inteligencia emocional en las instituciones educativas, para mejorar las habilidades socioemocionales de los estudiantes y su adaptación al entorno académico [11].

Existen varias razones para considerar la incorporación de un diario de emociones en la vida cotidiana. Llevar un diario de emociones puede ayudar a mejorar la salud mental al reducir los niveles de estrés, ansiedad y depresión. Además, fomenta la autorreflexión al permitir a las personas conocerse mejor, identificar patrones emocionales y comprender cómo las experiencias pasadas afectan las emociones actuales. Fortalece la toma de decisiones al promover decisiones más informadas y saludables. Además, al fomentar la aceptación de todas las emociones, incluso las negativas, el diario de emociones es un paso crucial hacia la autoaceptación [36].

Por lo tanto, es fundamental considerar la incorporación de un diario de emociones en la vida de los jóvenes para mejorar su bienestar emocional y mental.

4. Marco Teórico

4.1. Marco Referencial

4.1.1. Inteligencia Emocional

La inteligencia emocional, según Daniel Goleman, se trata de la capacidad de reconocer, comprender y gestionar las emociones propias y ajenas, así como de utilizar esta comprensión para construir relaciones sólidas y saludables con los demás. Estas habilidades son fundamentales tanto en la vida personal como en la profesional, ya que impactan en la toma de decisiones, la resolución de conflictos y la calidad de las relaciones humanas en general [14].

4.1.2. Emociones

Las emociones se refieren a las repuestas fisiológicas que puede llegar a experimentar una persona en consecuencia de estímulos externos, lo que permite familiarizarse con una situación relacionada con una persona, objeto o lugar. Estas respuestas son de corta duración, pero intensas en comparación con los sentimientos. En cuanto a los sentimientos, estos son las consecuencias de las emociones, son más duraderos y pueden ser expresados verbalmente [14].

4.1.3. Autoconsciencia

Es una habilidad que implica la comprensión y conciencia sobre las emociones propias, incluyendo la capacidad de identificar lo que se siente, por qué se siente de esa manera y cómo esas emocionas afectan las acciones y decisiones personales [14].

4.1.4. Autogestión

Se refiere a la capacidad de gestionar y regular las emociones propias de una manera efectiva. Incluye la habilidad de mantener un enfoque positivo frente a los desafíos, perseguir las distintas metas personales con determinación y controlar las emociones no placenteras para evitar que interfieran en la vida cotidiana [14].

4.1.5. Empatía

Consiste en ser capaz de comprender las emociones y perspectivas de los demás. Implica leer las señales emocionales de las personas, tanto las expresadas de manera verbal, como las que se comunican a través del lenguaje corporal o no

verbal, además de tener la capacidad de ponerse en el lugar de los demás para comprender sus puntos de vista y sentimientos [14].

4.1.6. Habilidades Sociales

Hacen parte integral de la inteligencia emocional, son habilidades que implican la capacidad de manejar relaciones interpersonales de manera efectiva y significativa, tales habilidades como trabajar en equipo, colaborar, revolver conflictos, influenciar a los demás de manera positiva y comunicarse de manera asertiva y efectiva [14].

4.1.7. Alfabetización Emocional

Es la capacidad que tiene una persona para comprender, expresar y manejar sus propias emociones y las emociones de los demás de manera efectiva [14].

4.2. Marco Conceptual

4.2.1. PostgreSQL

Permite almacenar datos e información de los registros del sistema, utilizando lenguaje SQL. [30]

4.2.2. Servidor web:

Es un software el cual permite comunicar distintas aplicaciones a través de los protocolos web como HTTP. [23]

4.2.3. REST API:

Es una Interfaz de Programación de Aplicaciones [33], la cual usa un formato de comunicación popular llamado JSON, y tiene la siguiente estructura:

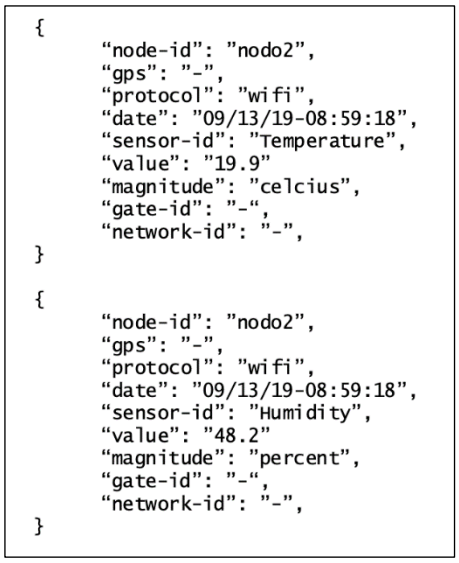


Imagen 1 Ejemplo Formato JSON

Fuente: [23]

En la imagen se muestran dos objetos JSON que representan datos de sensores. Cada objeto contiene información sobre una lectura de sensor específica, con formato de llave-valor.

4.2.4. JavaScript

Es un lenguaje de programación de propósito general, el cual nos permite realizar aplicaciones móviles, para Android y iOS, tal como se muestra en los proyectos de aplicación a la hidráulica [22], determinación de coeficientes [21] y en el proyecto para uso de Newton-Raphson. [20]

4.2.5. Node.js

“Es un entorno de ejecución multiplataforma de código abierto para desarrollar aplicaciones web. Esta librería se ejecuta sobre JavaScript. Node.js es un intérprete de JavaScript del lado del servidor. De igual manera permite usar el mismo lenguaje para el desarrollo del front-end y back-end.” [15]

4.2.6. TypeScript

Es un superconjunto de JavaScript que compila código limpio de JavaScript [26]. Agregando Tipado Estricto, POO, etc.

4.2.7. Dart

Dart es un lenguaje optimizado para el cliente para desarrollar aplicaciones rápidas en cualquier plataforma. Su objetivo es ofrecer el lenguaje de programación más productivo para el desarrollo multiplataforma, junto con una plataforma de ejecución flexible para marcos de aplicaciones. [9]

4.2.8. Flutter

Flutter se construyen utilizando un marco moderno que se inspira en React. La idea central es construir la interfaz de usuario a partir de widgets. Los widgets describen el aspecto que debería tener su vista en función de su configuración y estado actuales. Cuando cambia el estado de un widget, este reconstruye su descripción, que el framework compara con la descripción anterior para determinar los cambios mínimos necesarios en el árbol de renderizado subyacente para pasar de un estado al siguiente. [10]

4.2.9. NestJS

“A progressive Node.js framework for building efficient, reliable and scalable server-side applications.” [29]

4.2.10. Algoritmo SHA-256

“En SHA-256 de funciones hash el mensaje de longitud variable M es dividido en bloques de 512 bits M0, M1, ..., Mn-1. El valor hash de 256 bits Vn se da la siguiente manera que es la fórmula para encriptación de algoritmo SHA-256” [40].

$$V_0 = IV;\ V_{s+1} = \mathtt{compress}(V_s, M_s) = E_{M_s}(V_s) + V_s\ for\ 0 \leq s < n,$$

Imagen 2 Imagen Proceso de encriptación.

Fuente: [40]

- V_0 es el valor inicial del hash.
- IV es el vector inicial predefinido en SHA-256.
- V_{s+1} es el estado del hash después de procesar el bloque s+1.
- V_s es el estado del hash actual (después de procesar el bloque s).
- M_s representa el bloque de mensaje actual que se está procesando.
- compress() es la función de compresión de SHA-256.
- $EM_s()$ es la función de expansión de mensaje para el bloque s.

4.2.11. JWT (JSON Web Token)

“Los tokens web JSON son un método abierto y estándar de la industria RFC 7519 para representar reclamaciones de forma segura entre dos partes.” [19]

4.2.12. Disponibilidad

“Se encarga de asegurar que el acceso que se tenga a la información alojada en un sistema solamente lo puedan visualizar los usuarios autorizados, evita se pueda leer, copiar, o modificar la información a menos que se tenga autorización.” [5]

4.2.13. Autenticación

“Es la forma en la que un sistema compruebe la identidad de quien pretende ingresar, verificando la autenticación para así mismo conceder el permiso de acceder a la información solicitada, son las medidas que se deben tomar como protección a la información que se pretende acceder.” [5]

4.2.14. Go

“Go es un lenguaje de programación open source que permite crear software de manera simple, fiable y eficiente.” [42]

4.3. Antecedentes

4.3.1. Mood Tracker

Un mood tracker es una herramienta que se utiliza para registrar el estado de ánimo del usuario de la agenda. Se trata de una técnica de psicología para mejorar la salud mental y que ayuda a encontrar o identificar patrones sobre cómo varía un estado de ánimo y cómo afecta a la persona. Un mood tracker es importante para:

- Identificar los factores que provocan cambios de humor, tanto externos como internos, para así desarrollar técnicas que ayuden a lidiar con ellos.
- Averiguar cómo factores como el sueño, la dieta o las actividades diarias afectan en el día a día de un individuo.
- Detectar patrones y comprender mejor los cambios de humor.
- Mejorar la capacidad de tomar decisiones.
- Alertar sobre niveles crecientes de ansiedad y depresión, señalando el hecho de que se podría necesitar ayuda y asistencia profesional [37].

4.3.2. Diario de emociones

Potencialmente, el diario se compone por unos principios recurrentes a lo personal, cuyo funcionamiento requiere del uso constante para construir mutuamente resultados específicos de cada una de las piezas. Esta herramienta va acorde al planteamiento de transformación, proceso que permite visualizar cómo existen estados actuales que se encaminan y evolucionan a medida que el tiempo transcurre. Tal como lo menciona Diana Mendoza “El diario es la retrospectiva de algo y la introspectiva de sí mismo, es una forma de perderse para poder encontrarse”.

Con base en la pieza editorial Nárrame, se estima que el ejercicio de escritura, en función de un diario, cumplen con parámetros correspondientes a la etapa temprana de una situación presente del conflicto [15].

4.3.3. Herramientas Tecnológicas

4.3.3.1. MoodPress - Mood Diary Tracker

Diario personal de tus sentimientos y actividades.
Únete a tu Moodpress no sólo para registrar tu estado de ánimo, sino también para guardar tus secretos, fotos importantes y cualquier registro de tu vida sin importar cuándo y dónde. También es una misión de humor realmente impresionante para analizar su diario emocional grabado.

- Proteja su privacidad y mantener la confidencialidad - Reconocimiento de huellas dactilares y la función de contraseña de apoyo.

- Configura y organiza tu diario personal - Fechas, estados de ánimo, fotos, etiquetas de actividad, temas dinámicos, etc.
- Crea una actividad con tu propia frase - Busca y añade actividades personalizadas utilizadas con el icono seleccionado.
- Función de gestión del estado de ánimo mediante calendario y gráfico.
- Sonido curativo para ayudarle a calmarse - Sé genuinamente consciente de tus sentimientos internos.
- Copia de seguridad y restauración de la agenda - Copia de seguridad/restaura secretos en tu cuenta privada de Google Drive.
- Configura la alarma de la agenda y recíbela a la hora deseada.
- Puedes exportar documentos PDF.
- Comparte rápidamente tu calendario mensual de estados de ánimo con el tema y los emoticonos que quieras.
- Ver y utilizar directamente en la pantalla a través de un widget conveniente.

El diario se guarda siempre en tu teléfono. MoodPress valora tu privacidad y no la almacena ni recopila por separado. [1]



Imagen 3 Imagen herramienta tecnológica – MoodPress

Fuente: [1]

4.3.3.2. Daylio Jornal – Mood Tracker

“Bullet Jornal de autocuidado con objetivos, diario del estado de ánimo y seguimiento de la felicidad. Lleve un diario y capture su día sin ni una sola palabra.” [7]



Imagen 4 Imagen herramienta tecnológica – Mood Tracker.

Fuente: [7]

4.3.3.3. DailyBean: Simplest: Jornal

Es una aplicación de diario simple para aquellos que desean registrar su vida diaria fácilmente. ¡Registre su día con solo unas pocas pestañas!
DailyBean proporciona estas funciones.

- Calendario mensual que le permite ver el flujo de su estado de ánimo. Si hace click en el bean, puede verificar el registro que dejó en el día de inmediato.
- Toque los frijoles de estado de ánimo y los iconos de actividad para un registro simple. Elijamos tu estado de ánimo para el día y resumamos el día con íconos de colores. Puede agregar una imagen y una línea de notas.
- Bloques de categorías que le permiten seleccionar solo las categorías que desee. Los bloques se pueden agregar o eliminar cuando lo desee, y las categorías se actualizarán continuamente.

- Estadísticas que analizan el estado de ánimo y la actividad semanal / mensual. Observe el flujo de su estado de ánimo a través de las estadísticas y vea qué actividades afectan su estado de ánimo. También puede comprobar el número de registros de iconos de forma semanal y mensual. [31]

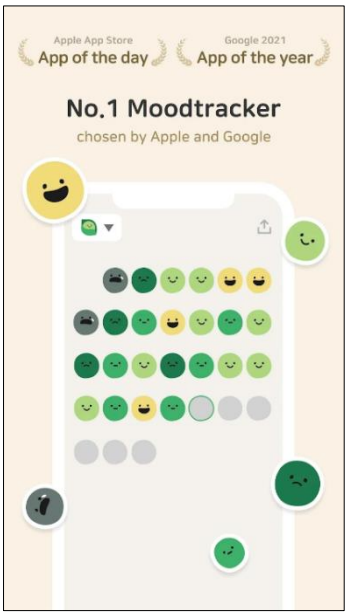


Imagen 5 Imagen herramienta tecnológica – DailyBean.

Fuente: [39]

4.3.3.4. How We Feel

“Es un diario gratuito para tu bienestar creado por científicos, diseñadores, ingenieros y psicólogos. Con el tiempo, aprenderás palabras precisas para describir cómo te sientes, detectar tendencias y patrones, y practicar estrategias sencillas para regular tus emociones de forma saludable.” [15]

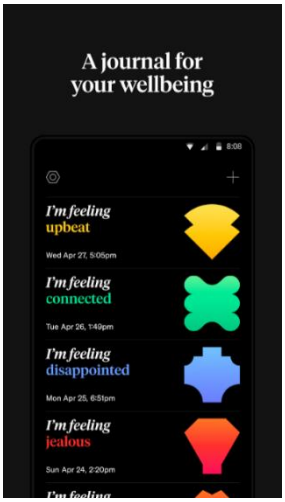


Imagen 6 Imagen herramienta tecnológica – How we feel.

Fuente: [17]

4.3.4. Comparativa

La gran diferencia que existe entre los Diarios Emocionales y los Mood Trackers, consiste en que, por un lado, los Mood Trackers buscan registrar el estado de ánimo e identificar cambios de humor, a través de factores internos como externos, los diarios emocionales están más orientados a generar un espacio personal en donde podamos desarrollar una retrospectiva e introspectiva, la cual finalmente nos ayude a "encontrarnos".

Realizamos una tabla comparativa entre las soluciones tecnológicas previamente descritas, la cual detalla factores relevantes para hallar la mejor solución para la Universidad Distrital Francisco José de Caldas.

*Tabla **1**. **Comparativa entre soluciones tecnológicas.***

Nombre	Soporte a español	Identificación de Emociones	Separación Emociones	Información de Emociones	Contenido Audiovisual Sugerido	Líneas de Atención	Usabilidad (1-5)
Moodpress - Mood Diary Tracker	X						1
Daylio Jornal -	X	X					1

Mood Tracker							
DailyBean: Simplest Jornal	X						3
How We Feel		X		X	X		4
Our Hive - Emotional Diary	X	X	X	X	X	X	2

Fuente: Autores

5. Marco Metodológico

5.1. SCRUMBAN (SCRUM-KANBAN)

Para la metodología de este proyecto, estaremos utilizando Scrumban, que se situó como la quinta metodología más utilizada en el período de 2015 a 2020. Durante ese tiempo, Scrum ocupó el primer lugar como la metodología más extendida, seguida de cerca por Kanban, que se ubicó en el tercer lugar [2]. Scrumban proporciona un enfoque altamente eficiente y adaptable para la gestión de proyectos, combinando elementos clave de ambas metodologías, Scrum y Kanban. Aquí hay más elementos clave de ScrumBan, incluyendo algunos conceptos específicos de Scrum:

- Definición de Tareas Claras: En ScrumBan, todas las tareas se definen de manera clara antes de asignarlas al flujo de trabajo Kanban. Esto evita malentendidos y asegura que el equipo tenga una comprensión precisa de lo que se espera de ellos.
- Limitación del Trabajo en Proceso (WIP): Una característica fundamental de Kanban es limitar el trabajo en proceso. Esto significa que se asigna un límite máximo de tareas en cada etapa del flujo de trabajo. Esto evita la sobrecarga y ayuda a mantener el enfoque en las tareas prioritarias.
- Priorización Dinámica: Tanto Scrum como Kanban permiten la reevaluación y la reorganización de las prioridades de manera continua. Esto es esencial para adaptarse a las cambiantes necesidades y desafíos del proyecto.
- Colaboración y Autonomía del Equipo: ScrumBan fomenta la colaboración entre los miembros del equipo. A la vez, les otorga cierta autonomía para gestionar sus tareas y responsabilidades dentro de los límites establecidos.
- Medición y Mejora Continua: ScrumBan se basa en la recopilación de datos y métricas para evaluar el rendimiento del equipo y del proyecto en general. Esto permite una mejora continua en la eficiencia y la calidad.

- Retroalimentación y Aprendizaje: La retroalimentación es fundamental en ScrumBan. Se alienta a los miembros del equipo a aprender de cada sprint o iteración, y a aplicar esos aprendizajes a futuros trabajos.
- Visibilidad del Proceso: Un aspecto central de Kanban es la visualización del proceso de trabajo. Esto proporciona a todos los interesados una visión clara y en tiempo real de cómo se están desarrollando las tareas.
- Scrum Master: El Scrum Master es un rol clave en Scrum. Esta persona actúa como un facilitador y un defensor del proceso Scrum. Se asegura de que se sigan las prácticas y reglas de Scrum y ayuda al equipo a ser más eficiente.
- Product Backlog: El Product Backlog es una lista de todas las funcionalidades, características, y mejoras que se desean en el producto. Es una lista priorizada que guía el trabajo del equipo.
- Sprint Planning: Antes de cada sprint, el equipo realiza una reunión de Sprint Planning. En esta reunión, se seleccionan elementos del Product Backlog para incluir en el Sprint y se establece el objetivo del Sprint.
- Sprints: Los sprints son iteraciones de trabajo que generalmente duran de 1 a 4 semanas. Durante un Sprint, el equipo trabaja para completar las tareas seleccionadas en el Sprint Planning y entregar un incremento del producto.
- Daily Scrum (Reunión Diaria del Scrum): Estas son reuniones diarias cortas en las que el equipo comparte lo que ha logrado desde la última reunión, identifica obstáculos y se sincroniza para el día.
- Sprint Review: Al final de cada Sprint, el equipo lleva a cabo una reunión de revisión para mostrar los resultados del Sprint y recibir retroalimentación de los stakeholders.

Dado que somos un equipo pequeño, ScrumBan nos permite visualizar de manera constante las actividades del equipo. Facilitando la adición continua de funcionalidades y permite a nuestro Product Owner tener una visión clara de las tareas en curso. Además, seríamos altamente reactivos a los cambios, lo que mejora nuestra capacidad para adaptarnos a medida que avanzamos en el proyecto.

6. Alcances

La idea principal de este proyecto consiste en crear una aplicación móvil la cual brinde herramientas, para realizar un seguimiento de las emociones, por lo cual es necesario hacer una clara separación en el registro de emociones, donde separemos los registros entre emociones diarias y emociones trascendentales, las ultimas hacen referencia a todas las emociones que denotan un gran valor o una complicación en su gestión de parte de los usuarios. Mientras que el registro diario busca que el usuario haga un breve resumen de su día e identifique que emoción fue la que sintió alrededor del día.

A su vez es importante que durante la etapa de registrar una emoción el usuario pueda ver una interfaz amigable la cual realice una separación entre las emociones primarias y secundarias elegibles dentro de la aplicación.

Puesto que el proyecto planteado busca apoyar a la comunidad de la Universidad Distrital Francisco José de Caldas, será útil tener en cuenta el idioma usado dentro de la aplicación el cual tendrá de preferencia español y deberá ser principalmente una aplicación móvil, puesto que, para la integración del software con otras áreas de conocimiento, es útil centralizar y facilitar el uso a nuestros usuarios [3].

6.1 Descripción de los Módulos

6.1.1. Perfil

En este módulo se le permitirá al usuario visualizar y editar su información personal como datos realmente relevantes podríamos tener:

- Nombre de Usuario.
- Nombre Completo.
- Contraseña (Para el cual se debe hacer uso de SHA-256, puesto que podemos brindar un cierto nivel de seguridad a las contraseñas [5].
- Correo.

6.1.2. Autenticación y Autorización

Es necesario manejar Autenticación y Autorización en nuestra aplicación, puesto que de esta manera se gestionarán los permisos y los roles que se tienen dentro de la misma.

Para la parte de Autenticación es necesario en cada funcionalidad de la aplicación validar el hecho de que el usuario este Autenticado y cuente con un JWT Asignado desde el Backend Centralizado.

Para la parte de Autorización es relevante manejar este campo a través de validaciones en el Backend y uso del JWT previamente comentado, esto con el fin de verificar y dar o quitar permisos a otros usuarios de poder ver el historial de otros.

6.1.3. Registro Diario Emocional

Este módulo hace parte del núcleo de nuestra solución tecnológica, ya que brindar una herramienta que le permita a nuestros usuarios generar el hábito de constantemente hablar y registrar las emociones que se viven de manera diaria. Lo cual brindará conciencia emocional. Con el desarrollo de una interfaz las cual separé las emociones primarias de las secundarias (derivadas).

6.1.4. Registro Emociones Trascendentales

Por otra parte, el presente módulo también hace parte fundamental para el cumplimiento de los objetivos del proyecto, puesto que, para diseñar los registros de emociones, que permitan a los usuarios llevar un registro de su estado emocional y conocer información importante acerca de las emociones, pero el módulo se diferencia del diario emocional, ya que está enfocado en los desbordes emocionales. Esto se refiere a situaciones en las que las emociones de una persona pueden sobrepasar su capacidad de gestión emocional, y esto puede ocurrir varias veces durante el día. En contraste con un el módulo de “Registro Diario Emocional”, que generalmente se utiliza para registrar emociones de manera más general o a lo largo del tiempo, este módulo se enfoca en capturar esos momentos intensos o trascendentales en los que las emociones son más abrumadoras.

Para el desarrollo de este módulo es importante capturar información del momento que produjo este desborde emocional; tal como:

- Lugar.
- Miembros cercanos (Familiares, Amistades, Relaciones Románticas, etc.).
- Actividad que se realizaba.

6.1.5. Historial Emocional

El módulo busca desarrollar un sistema de historial en el cual se pueda ver los registros realizados por los usuarios dentro de la aplicación, para el cual sería útil contar con sistemas de filtrado, para que el usuario pueda tener una mayor facilidad de visualizar las emociones vividas con anterioridad. Permitiéndole al usuario realizar una retrospectiva emocional, análisis y tomar decisiones.
A su vez el usuario podrá abrir un detalle del registro, brindándole detalles acerca de las emociones registradas.

6.1.6. Información y Contenido Recomendado

Será importante contar con un módulo orientado a la información y contenido, puesto para manejar el proceso del desarrollo de la Inteligencia Emocional, es vital contar con números de contacto que apoyen los momentos de desbordes emocionales, haciendo más fácil la gestión emocional; Tener enlaces de interés hacia Podcast, videos, imágenes, etc. con el fin de generar una cultura psicológica y de salud metal en nuestros usuarios.

6.1.7. Permisos a Usuarios

El módulo deberá permitir al usuario gestionar los permisos que tendrán otros usuarios para ver información del historial de emociones vividas por el usuario, esto se realiza para brindar una herramienta enfocada en el punto de “habilidades sociales”, ya que es importante tener una red de apoyo (Amigos, Familiares y

personas con las que podríamos apoyarnos emocionalmente). Pero es importante darles el control absoluto a los usuarios, para que los mismos puedan definir a que personas les brinda este acceso y que siempre tengan la posibilidad de deshabilitarle este acceso.

6.1.8. Notificaciones

Por último, pero como parte de los módulos más importantes de la aplicación, es necesario definir concretamente el módulo. El cual consiste en definir unas notificaciones que le sean útiles para el uso de la aplicación de parte de nuestros usuarios. Por lo cual será importante:

- Notificaciones para impulsar al usuario a llevar el diario emocional.
- Notificar a los usuarios, si tienen el acceso a el historial de otro usuario.
- Notificar acerca de nuevo contenido audiovisual disponible dentro de la aplicación.

6.2. Diseño de interfaces y Descripción

Log In

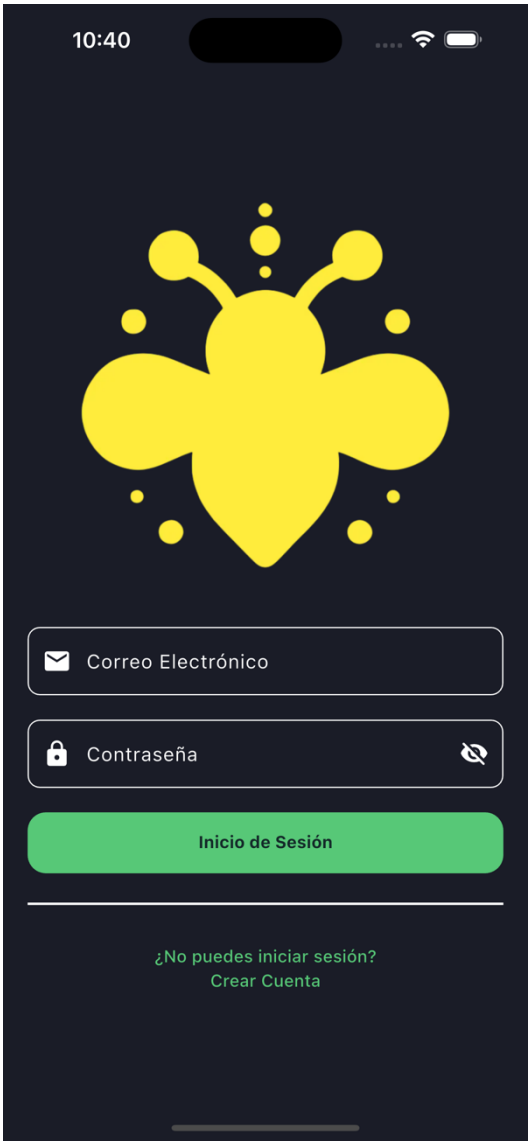
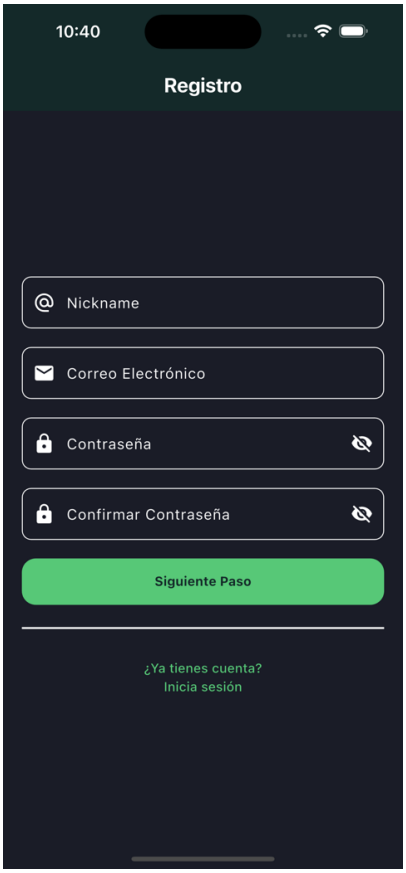


Imagen 7 Pantalla Inicio de Sesión.

Fuente: Autores

La pantalla de log In, será la encargada de permitir a nuestros usuarios ingresar a la aplicación con el fin de hacer uso de todas las herramientas de los módulos previamente descritos.

Registro



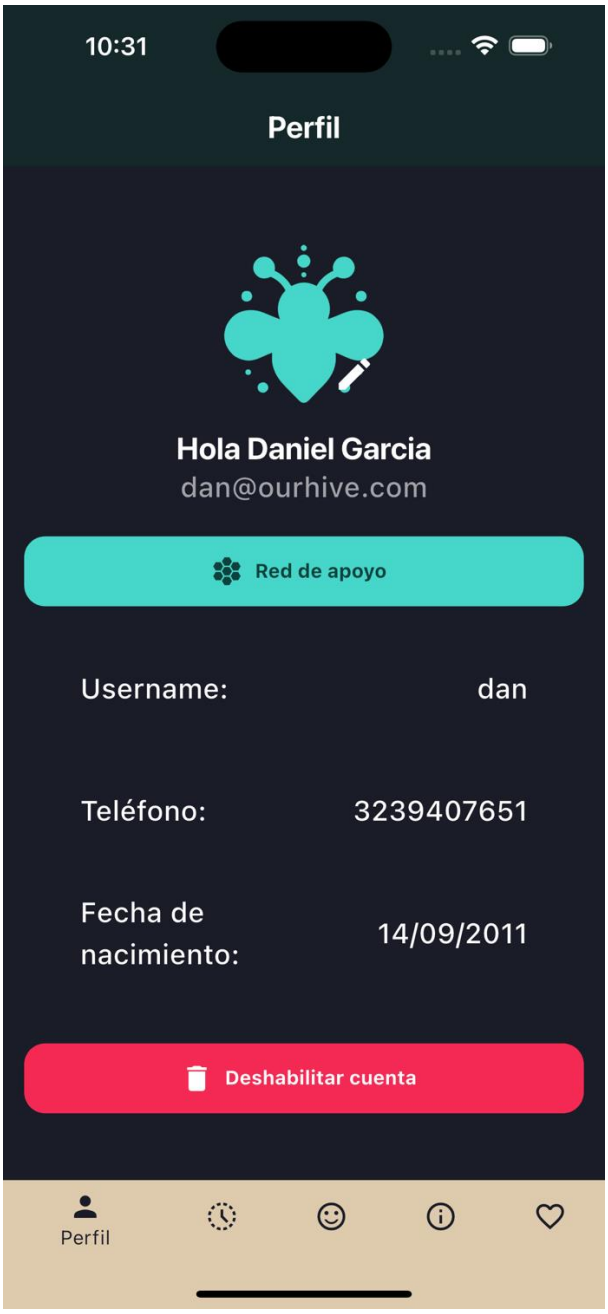


A su vez es importante contar con una pantalla encargada de obtener la información relevante de nuestros usuarios, para la creación de la cuenta.

Imagen 8 Pantallas de Registro.

Fuente: Autores

Perfil de Usuario



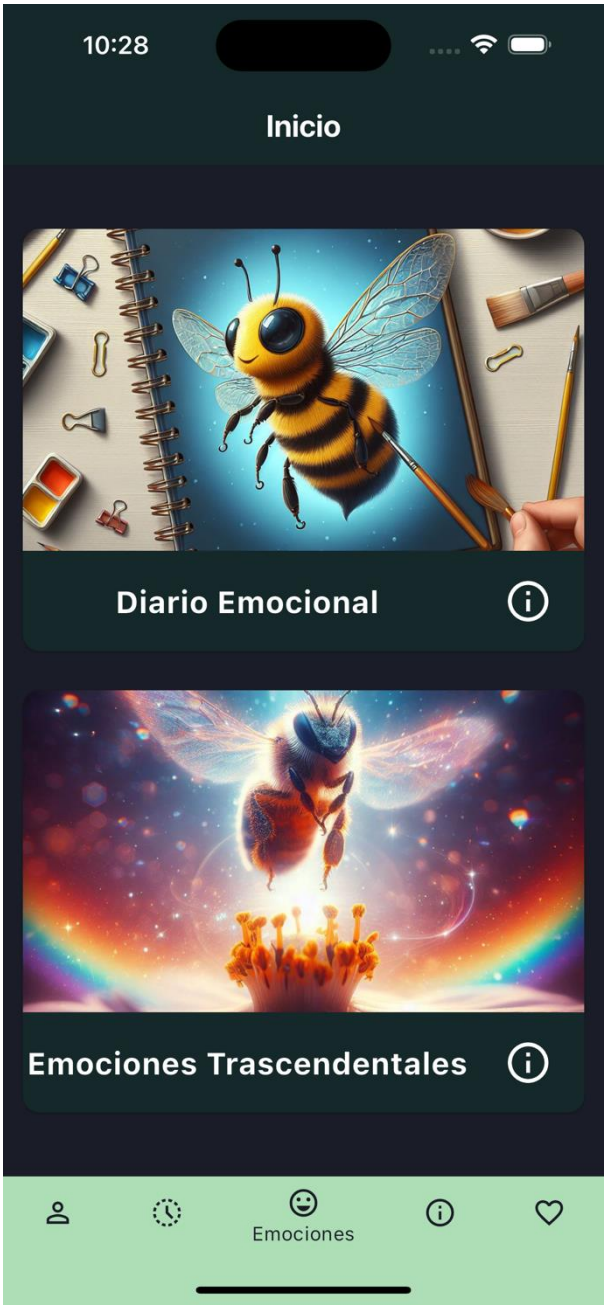
De los primeros módulos, tenemos el denominado perfil de usuario el cual está encargado de permitir al usuario 3 acciones principales:

1. Visualizar la información del usuario.
- Editar los campos permitidos.
- Deshabilitar la cuenta.

Imagen 9 Pantalla Perfil de Usuario

Fuente: Autores

Registro Emocional



Esta pantalla hace referencia a la pantalla principal de la aplicación por lo cual cuenta con la posibilidad de elegir entre:

1. Crear un registro de emoción trascendental.
2. Crear un registro de diario emocional.

A su vez cuenta con pequeños botones los cuales mostraran información referente a en que consiste cada uno de los tipos de registros.

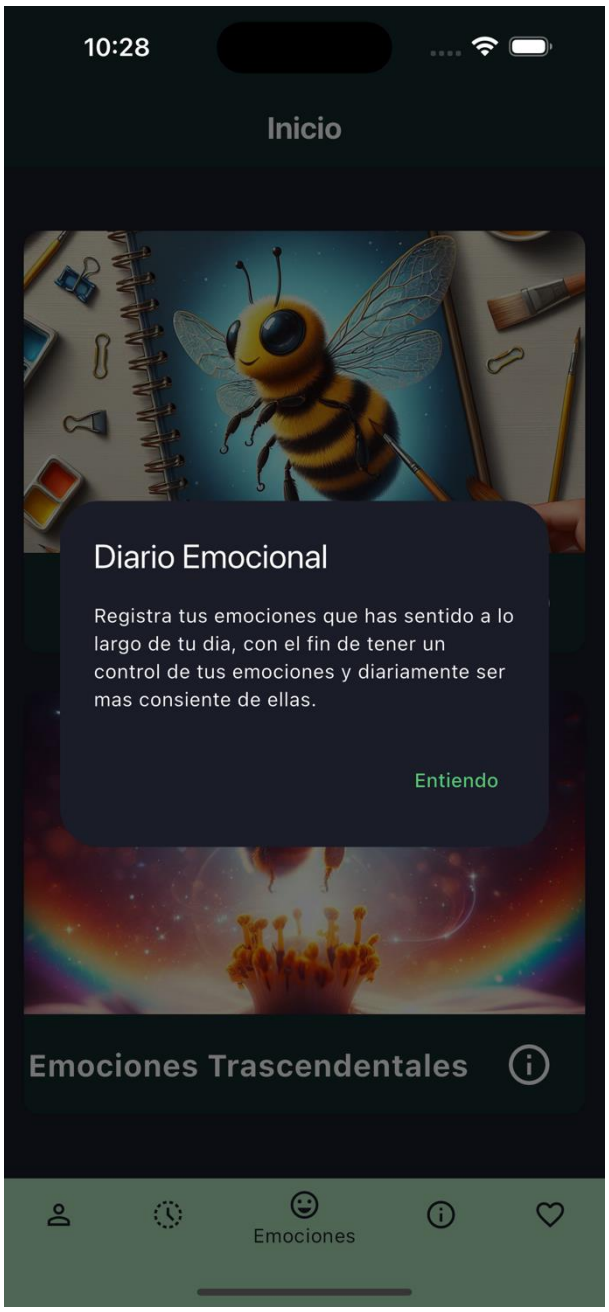

10:28
Inicio
Diario Emocional
Registra tus emociones que has sentido a lo largo de tu dia, con el fin de tener un control de tus emociones y diariamente ser mas consiente de ellas.
Entiendo
Emociones Trascendentales
Emociones



Imagen 10 Pantallas de Registro Emocional.

Fuente: Autores

Selector de Emoción Primaria



Esta pantalla es la que permitirá al usuario seleccionar entre las distintas emociones primarias, siendo a su vez la primera parte de los registros de:

1. Diario Emocional.
2. Emoción Trascendental.

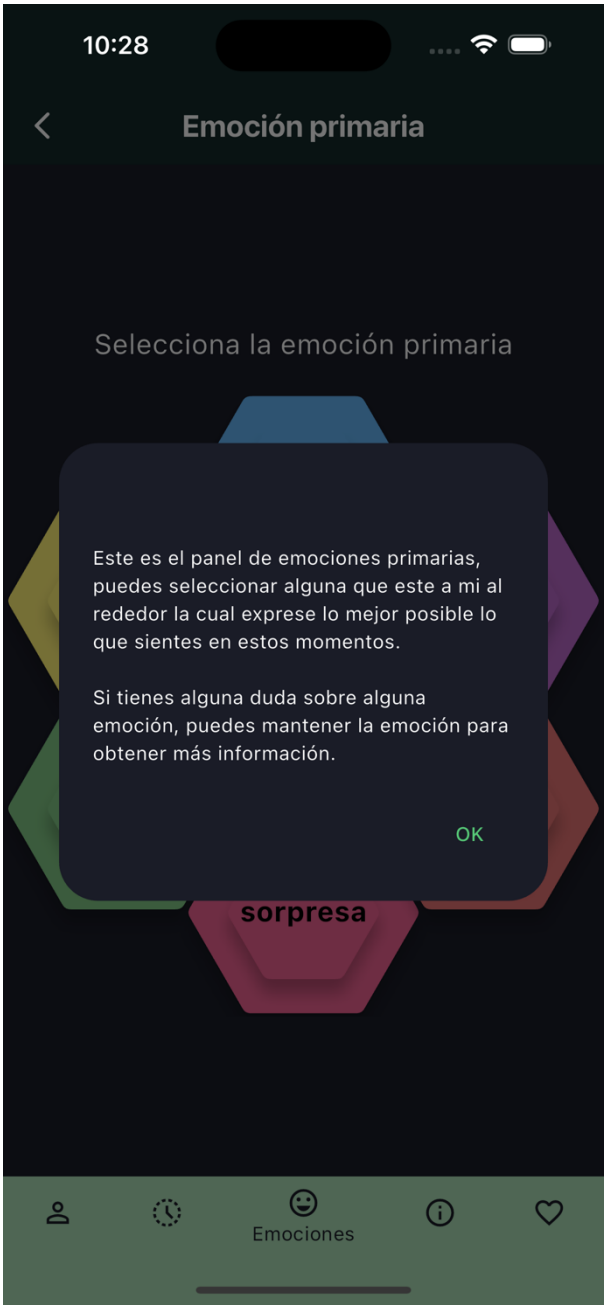


Imagen 11 Pantalla Registro de emoción primaria.

Fuente: Autores

Selector Emoción Secundaria



Este módulo hace referencia a la segunda parte fundamental de ambos registros, esta consiste en un listado de N emociones las cuales harán referencia a las emociones secundarias, derivadas de la emoción primaria previamente seleccionada.
A su vez se le dará al usuario información la cual brinde una descripción de la emoción seleccionada.



Imagen 12 Pantalla de Registro Emoción Secundaria.

Fuente: Autores

Formulario Diario Emocional

Esta pantalla que hará referencia a la última sección del registro diario. Puesto que este tipo de registro busca que el usuario haga una retrospectiva de las emociones y sucesos que pasaron durante

su día, se le brindara la posibilidad de expresarse libremente.
Finalmente se guardará este registro, como registro del día concreto.



Imagen 13 Pantalla de Formulario Diario Emocional.

Fuente: Autores

Formulario Emoción Trascendental

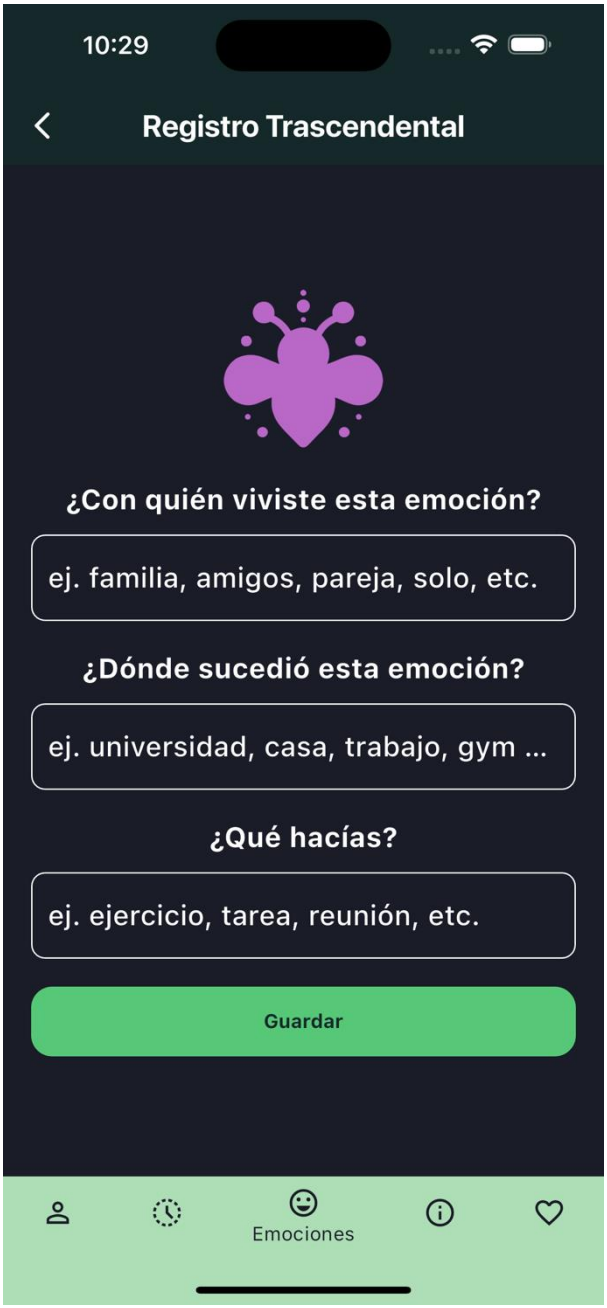


Imagen 14 Pantalla de Formulario Registro Emoción Trascendental.

Fuente: Autores

Esta pantalla que hará referencia a la última sección del registro emoción trascendental.

Puesto que este tipo de registro busca que el usuario sea concreto con la situación que fue trascendental o desbordante, será necesario preguntar por algunos ítems concretos como:

- ¿Con quién viviste la emoción?
- ¿Dónde sucedió?
- ¿Que estabas haciendo?

Esto con el fin de registrar acciones concretas que actuaron alrededor de esa emoción trascendental o desbordante.

Historial



Este módulo hace parte fundamental para la aplicación ya que un ítem clave para el desarrollo de la inteligencia emocional, costa de tener autoconciencia de nuestras emociones, por lo cual tener un módulo especial enfocado en mostrar un historial de las emociones registradas en la aplicación.

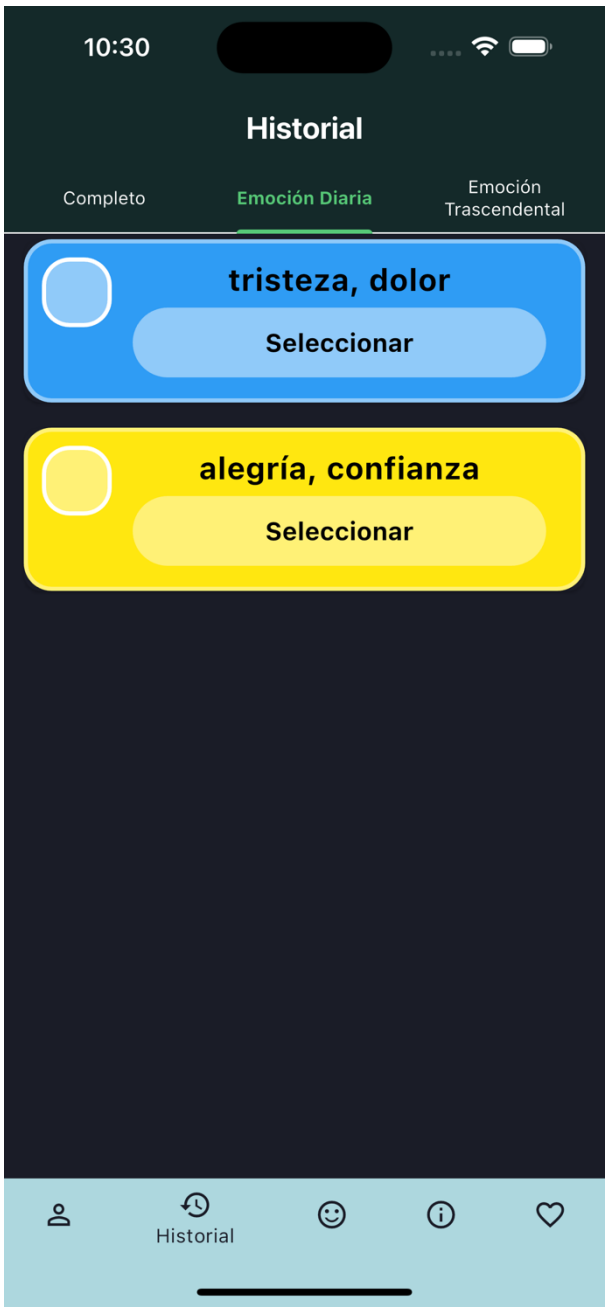

10:30
Historial
Completo
Emoción Diaria
Emoción Trascendental
tristeza, dolor
Seleccionar
alegría, confianza
Seleccionar
Historial



Imagen 15 Pantalla del Historial Emocional.

Fuente: Autores

Detalle de Registro Diario y Registro Trascendental



Imagen 16 Pantalla Detalle Registro Diario.

Fuente: Autores



Imagen 17 Pantalla Registro Trascendental.

Fuente: Autores

Sera una pantalla enfocada en mostrar la información previamente digitalizada de parte de los usuarios con el fin de tener a la mano el detalle de cómo se sintió en determinado momento, tendremos este detalle para:

1. Registro Diario Emocional.
2. Registro Emoción Trascendental.

Información y contenido recomendado

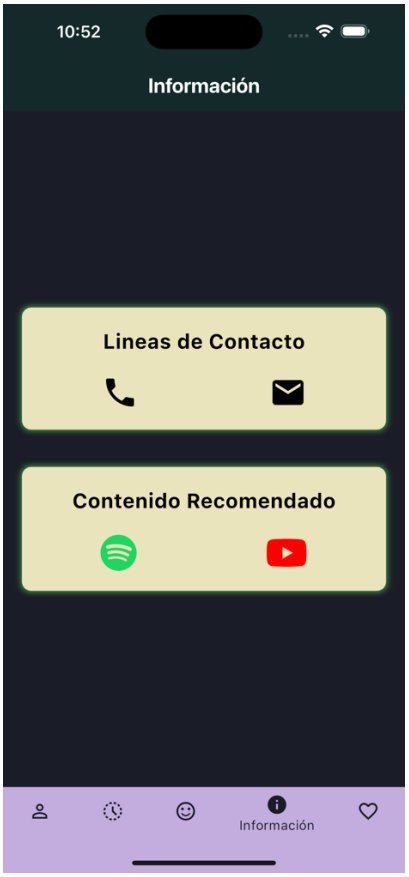


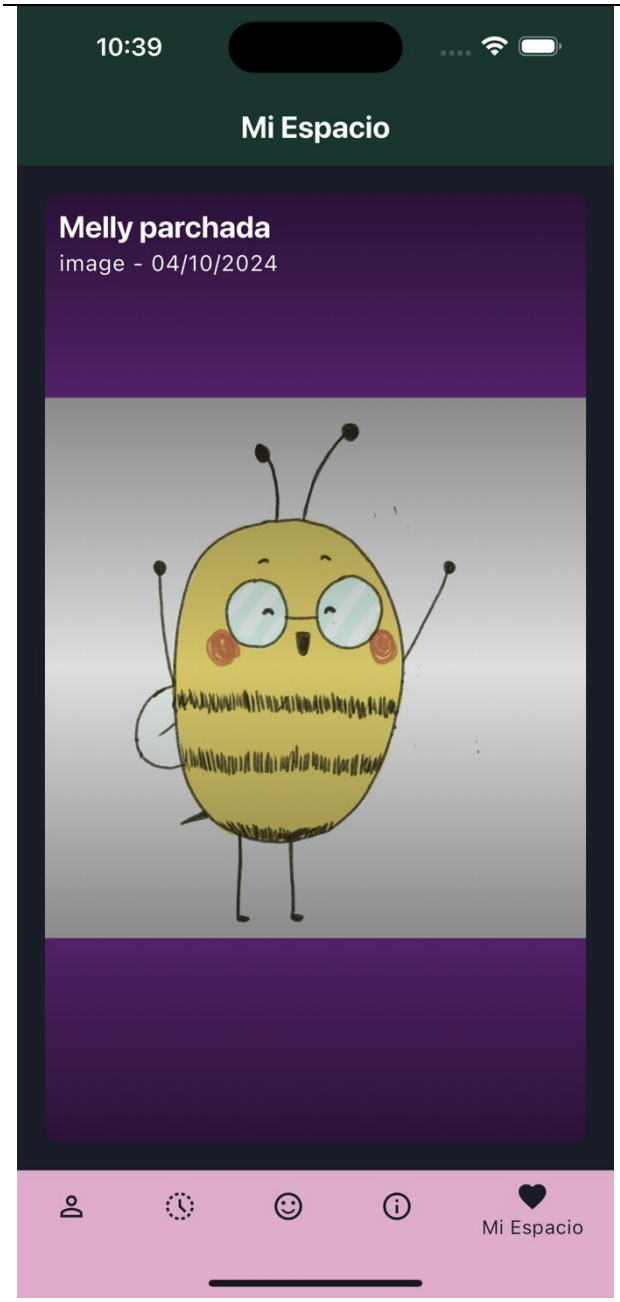
Imagen 18 Pantalla de Información.

Fuente: Autores

Este módulo tendrá como fin dar información útil a los otros usuarios, la información estará separada en 2 ítems importantes:

1. Líneas de contacto, para tener un apoyo en un desborde emocional.
2. Contenido en formato audio y audiovisual de psicología e inteligencia emocional.

Lo cual brindaría un apoyo para la alfabetización emocional de nuestros usuarios.



El objetivo de este módulo es crear un espacio para que el usuario se pueda sentir seguro, con contenido audiovisual, el cual brindé un espacio de relajación para los usuarios.

El contenido estará ubicado en el recuadro verde.



Imagen 19 Pantalla de My Space

Fuente: Autores

Usuarios Permitidos

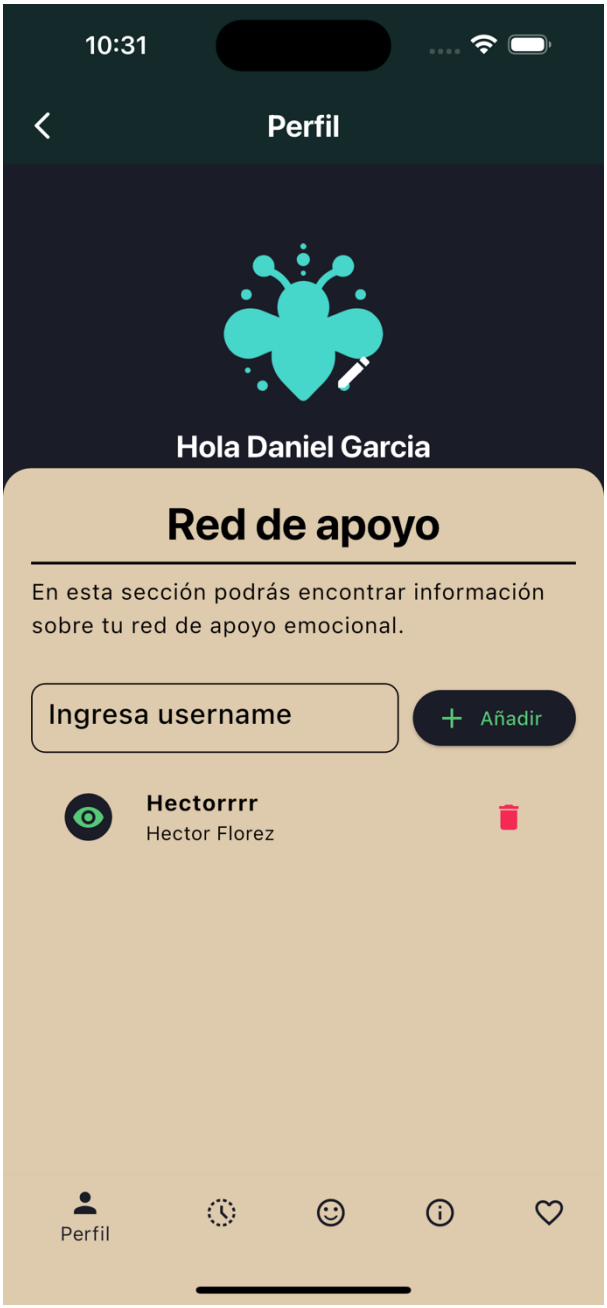


Imagen 20 Pantallas de Red de apoyo.

Fuente: Autores

Este módulo busca mostrar una lista de distintos usuarios los cuales, se les concederá o denegará los permisos de acceder al historial del usuario actual, esto se realiza con el fin de compartir nuestras emociones y sentimientos a otros usuarios de la aplicación.

7. Delimitaciones

7.1. Delimitación Temporal

El proyecto por realizar será desarrollado en un tiempo límite de 6 meses, para posteriormente ser enviado a la oficina asesora de sistemas para su respectiva evaluación.

7.2. Delimitación Geográfica

El proyecto estará orientado a la comunidad de la Universidad Distrital Francisco José de Caldas la cual cuentas con facultades en:

Tabla 2. Delimitación Geográfica.

Nombre	**Ubicación (**Bogotá, Colombia**)**
Facultad de Artes - ASAB	Carrera 13 # 14 – 69
Facultad de Ciencias de la Salud	Calle 52 Sur # 93D – 97
Facultad de Ciencias Matemáticas y Naturales	Carrera 4 # 26 D - 31
Facultad de Ciencias y Educación	Carrera 3 # 26A - 40
Facultad de Ingeniería	Carrera 7 # 40B - 53
Facultad del Medio Ambiente y Recursos Naturales	Carrera 5 Este # 15-82
Facultad Tecnológica	Calle 68D Bis A Sur # 49F - 70

Fuente: Universidad Distrital [2]

7.3. Delimitación Técnica

La aplicación está orientada en las principales plataformas de dispositivos móviles (IOS y Android).

Tabla 3. Delimitación Técnica.

Tecnología	**Descripción**
Android	Sistema Operativo móvil
iOS	Sistema Operativo móvil
PostgreSQL	Motor de Base de Datos
Visual Studio Code	Editor de texto
NestJS	Framework para el desarrollo de aplicaciones del lado del servidor
Flutter	SDK para el desarrollo de aplicaciones multiplataforma

Fuente: Autores

8. Factibilidad

8.1. Factibilidad Económica

El proyecto por realizar requiere unos mínimos recursos para poder ser llevado a cabo. Dentro de estos se debe contar con el equipo de trabajo, software, hardware, acceso a Internet y posibles gastos de papelería.

A continuación, se presentarán algunas tablas con la clasificación de la factibilidad económica, organizadas de la siguiente manera:

- Herramientas de hardware y software.
- Recursos Humanos.
- Otros Recursos.

Tabla 4. Factibilidad económica recursos técnicos.

TIPO	**DESCRIPCIÓN**	**VALOR UNITARIO**	**CANTIDAD**	**TOTAL**
Equipo MacBook Air	Para el desarrollo es necesario contar por lo menos con 1 equipo de cómputo el cual permita trabajar con simuladores de Android/iPhone.	$ 4’700.000	1	$ 4’700.000
Equipo de computo	Para el desarrollo es necesario contar con 1 equipo que permita trabajar de manera ágil.	$ 2’000.000	1	$ 2’000.000
TypeScript (Software libre)	Lenguaje de programación con el cual se desarrollará principalmente la aplicación.	$ 0	2	$ 0

PostgreSQL (Software libre)	Sistema de Administración de Base de Datos.	$ 0	1	$ 0
Visual Studio Code (Software libre)	Nos ayudara a gestionar el código fuente.	$ 0	2	$ 0
Flutter (Software libre)	Tecnología con la que podremos desarrollar las aplicaciones móviles.	$ 0	1	$ 0
Draw.io (Software Libre)	Nos permitirá desarrollar los diagramas necesarios para la visualización de la arquitectura de la aplicación.	$ 0	2	$ 0
Licencia de office	Nos permite realizar y juntar cada fase de la documentación.	$ 360.000	1	$ 360.000
GitHub	Es necesario contar con una cuenta en GitHub, para desarrollar la aplicación de manera colaborativa.	$ 0	2	$ 0
Total, recursos de software y hardware				**$ 7’ 060.000**

Fuente: Autores

Tabla 5. Factibilidad económica recursos humano y capacitación.

TIPO	**DESCRIPCIÓN**	**VALOR**	**CANTIDAD**	**TOTAL**
Tutor	Para la revisión periódica del desarrollo del proyecto y asesoría de este.	$ 40.000	200 horas	$ 8.000.000

Desarrolladores	Dos programadores con la responsabilidad de desarrollar el aplicativo, planeación, documentación y el diseño.	$ 1’500.000 mensual x desarrollador	6 meses	$ 18’000.000
Total, recursos humanos				**$ 26’ 000.000**

Fuente: Autores

Tabla 6. Factibilidad económica otros recursos.

TIPO	**DESCRIPCIÓN**	**VALOR**	**CANTIDAD**	**TOTAL**
Costo de impresión y papeleo.	Es posible requerir algunos costos de papeleo, principalmente enfocado a la parte de documentación.	$ 5.000	6 meses	$ 30.000
Costo de Internet	Es necesario contar con un buen acceso a internet para el trabajo colaborativo y desarrollo de la aplicación.	$ 100.000	6 meses	$ 600.000
Membresía de Desarrollador para Apple.	Para la publicación en la App Store de Apple, es necesario tener una membresía que se debe renovar de manera anual.	99 USD (432.000 COP aproximadamente).	1 x año	$ 432.000
Membresía de Desarrollador para Android.	Para la publicación en tiendas Play Store, es necesario una	25 USD (109.000 COP	1	$ 109.000

	membresía que permite esta acción.	aproximadamente).		
Total, otros recursos				**$ 1’171.000**

Fuente: Autores

*Tabla **7**. Factibilidad económica total.*

RECURSO	**VALOR**
Total, recursos de software y hardware	$ 7’ 060.000
Total, recursos humanos	$ 26’ 000.000
Total, otros recursos	$ 630.000
Costos Imprevistos 10%	$ 3’369.000
Total, costos	**$ 37’059.000**

Fuente: Autores

8.2. Factibilidad Técnica

Para el desarrollo del proyecto se hará uso de las siguientes herramientas, gestores, entornos y lenguajes, con el objetivo de desarrollar de manera ágil y robusta.

Tabla 8. Descripción Tecnologías.

Tecnología - Recurso	**Nombre**
SDK de desarrollo	Flutter
Frameworks de desarrollo	NestJS

Lenguaje de Programación	TypeScript
Base de Datos	PostgreSQL
Gestor de Base de Datos	Beekeeper Studio Community
Control de versiones	Git
Plataforma de alojamiento de código	GitHub
IDE	Visual Studio Code
Herramienta para la Metodología	GitHub Projects

Fuente: Autores

8.3. Factibilidad Legal

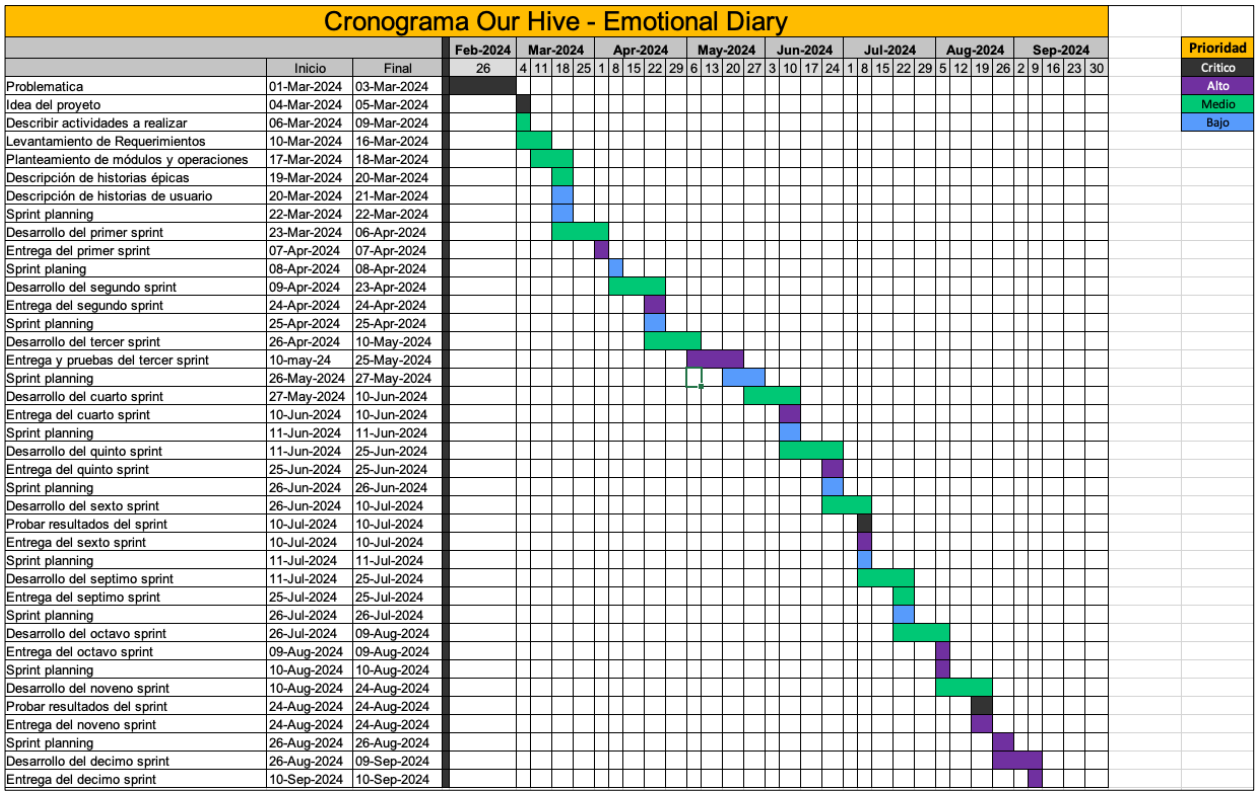
Se conoce que todo software o herramienta está ligada a licencias legales sobre su uso y distribución de copias, que también estaría sujeto a derechos de autor y otras normas. Lo anterior depende de si el software usado es de acceso libre o debe se pagado para su uso.

Las tecnologías mencionadas en la tabla 8, son las van a ser usadas para el desarrollo del sistema y todas son de acceso libre y gratuito, aunque cada una con sus propias especificaciones que serán descritas a continuación:

- Para las licencias de NestJS [28] el cual es el framework principal para el desarrollo backend cuentan con licencias MIT, la cuales nos permiten el libre uso de la tecnología.
- Flutter [41], el SDK que usaremos para el desarrollo frontend, cuentan con licencia BSD 2-Clause License [10], las cuales nos permiten el libre uso de la tecnología.
- La licencia Apache 2.0, nos permite hacer uso libre del lenguaje de programación TypeScript [24].
- PostgreSQL [31] cuenta con una licencia de Open Source, la cual nos permite el uso, modificación y distribución del software.

- Beekeeper Studio Community [4], cuenta con una licencia GPL (Licencia Publica General), la cual nos permiten el uso libre y gratuito del software.
- Git [13], cuenta con una licencia GPL (Licencia Publica General), la cual permite al software su libre uso, distribución, etc.
- GitHub [12], cuenta con una licencia la cual permite el uso de la plataforma, para el alojamiento de código.
- La licencia de Visual Studio Code [25], detallada por Microsoft se establece el uso libre, su distribución, etc.
- GitHub [6], cuenta con un plan gratuito de uso, perfecto para el desarrollo y organización de este proyecto.

9. Cronograma



	Inicio	Final
Problematica	01-Mar-2024	03-Mar-2024
Idea del proyeto	04-Mar-2024	05-Mar-2024
Describir actividades a realizar	06-Mar-2024	09-Mar-2024
Levantamiento de Requerimientos	10-Mar-2024	16-Mar-2024
Planteamiento de módulos y operaciones	17-Mar-2024	18-Mar-2024
Descripción de historias épicas	19-Mar-2024	20-Mar-2024
Descripción de historias de usuario	20-Mar-2024	21-Mar-2024
Sprint planning	22-Mar-2024	22-Mar-2024
Desarrollo del primer sprint	23-Mar-2024	06-Apr-2024
Entrega del primer sprint	07-Apr-2024	07-Apr-2024
Sprint planing	08-Apr-2024	08-Apr-2024
Desarrollo del segundo sprint	09-Apr-2024	23-Apr-2024
Entrega del segundo sprint	24-Apr-2024	24-Apr-2024
Sprint planning	25-Apr-2024	25-Apr-2024
Desarrollo del tercer sprint	26-Apr-2024	10-May-2024
Entrega y pruebas del tercer sprint	10-may-24	25-May-2024
Sprint planning	26-May-2024	27-May-2024
Desarrollo del cuarto sprint	27-May-2024	10-Jun-2024
Entrega del cuarto sprint	10-Jun-2024	10-Jun-2024
Sprint planning	11-Jun-2024	11-Jun-2024
Desarrollo del quinto sprint	11-Jun-2024	25-Jun-2024
Entrega del quinto sprint	25-Jun-2024	25-Jun-2024
Sprint planning	26-Jun-2024	26-Jun-2024
Desarrollo del sexto sprint	26-Jun-2024	10-Jul-2024
Probar resultados del sprint	10-Jul-2024	10-Jul-2024
Entrega del sexto sprint	10-Jul-2024	10-Jul-2024
Sprint planning	11-Jul-2024	11-Jul-2024
Desarrollo del septimo sprint	11-Jul-2024	25-Jul-2024
Entrega del septimo sprint	25-Jul-2024	25-Jul-2024
Sprint planning	26-Jul-2024	26-Jul-2024
Desarrollo del octavo sprint	26-Jul-2024	09-Aug-2024
Entrega del octavo sprint	09-Aug-2024	09-Aug-2024
Sprint planning	10-Aug-2024	10-Aug-2024
Desarrollo del noveno sprint	10-Aug-2024	24-Aug-2024
Probar resultados del sprint	24-Aug-2024	24-Aug-2024
Entrega del noveno sprint	24-Aug-2024	24-Aug-2024
Sprint planning	26-Aug-2024	26-Aug-2024
Desarrollo del decimo sprint	26-Aug-2024	09-Sep-2024
Entrega del decimo sprint	10-Sep-2024	10-Sep-2024

*Ilustración 2*1: Imagen cronograma.

Go: Es un lenguaje que fue diseñado por Google pensando en el desarrollo de aplicaciones backend, aprovechando su eficiencia en la gestión de la concurrencia y el uso de recursos, su simplicidad permite crear sistemas rápidamente sin lidiar con una sintaxis compleja, esta simplicidad también da la oportunidad de usar distintos modelos arquitectónicos como arquitectura hexagonal, MVC, etc. Además, es un lenguaje con una biblioteca estándar muy completa, esta elimina la necesidad de dependencias externas para crear proyectos robustos y fáciles de mantener. Al ser un lenguaje diseñado y soportado por Google, Go cuenta con una integración

fluida con Firebase, también desarrollado por Google, lo que garantiza un rendimiento optimizado y una integración sencilla. Además, la documentación oficial es muy completa y bien mantenida, lo que facilita la adopción del lenguaje y acelera la implementación de soluciones basadas en bases de datos NoSQL y almacenamiento de archivos en la nube.

NestJS: Es un framework para Node.js diseñado para el desarrollo de aplicaciones backend, este utiliza TypeScript y tiene un enfoque en la escritura de código mantenible y bien estructurado. Gracias a su arquitectura modular, permite crear sistemas escalables de forma rápida y sencilla. NestJS cuenta con un ORM muy completo, TypeORM, el cual simplifica la interacción con bases de datos relacionales, permitiendo una gestión de datos de forma eficiente sin tener que escribir consultas SQL. La facilidad de uso de este ORM lo convierte en una excelente opción para aplicaciones que requieren acceso frecuente a bases de datos.

Módulo de Autenticación (NestJS)

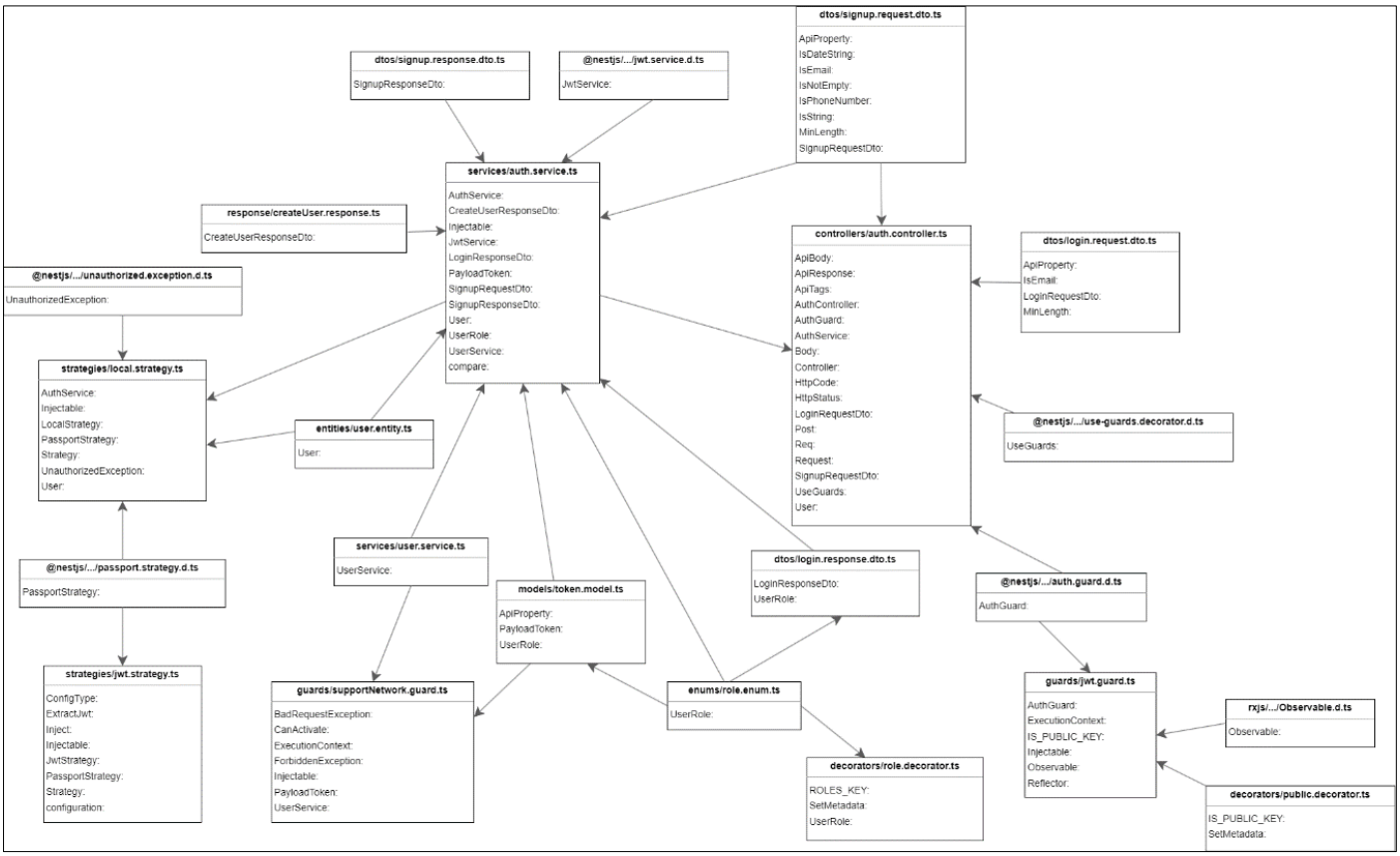


Imagen 21 Imagen diagrama de clases – Modulo de autenticación (NestJS)

Módulo de Emociones (NestJS)

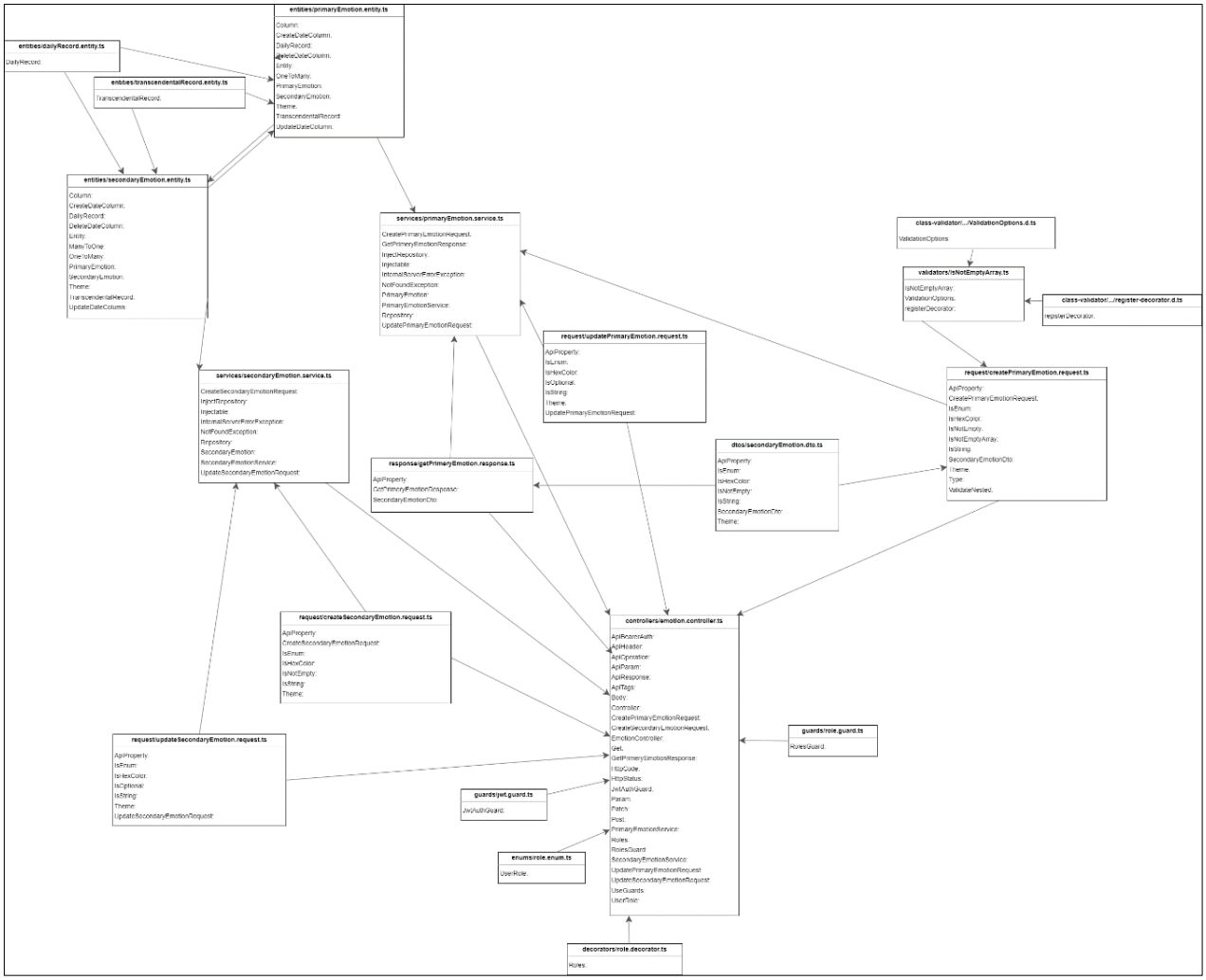


Imagen 22 Imagen diagrama de clases – Modulo de emociones (NestJS)

Módulo de Registro de Emociones (NestJS)

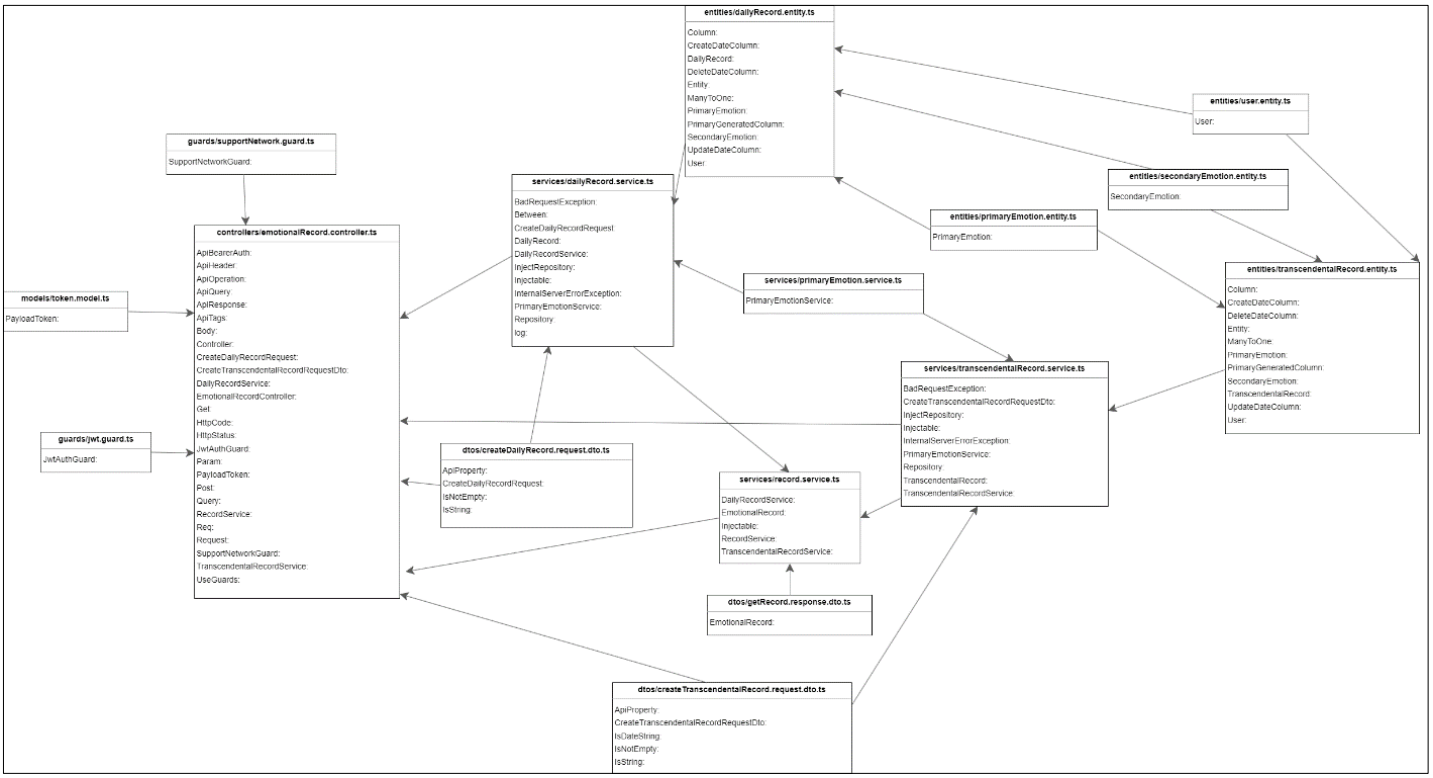


Imagen 23 Imagen diagrama de clases – Modulo de registro de emociones (NestJS)

Módulo de Usuarios (NestJS)

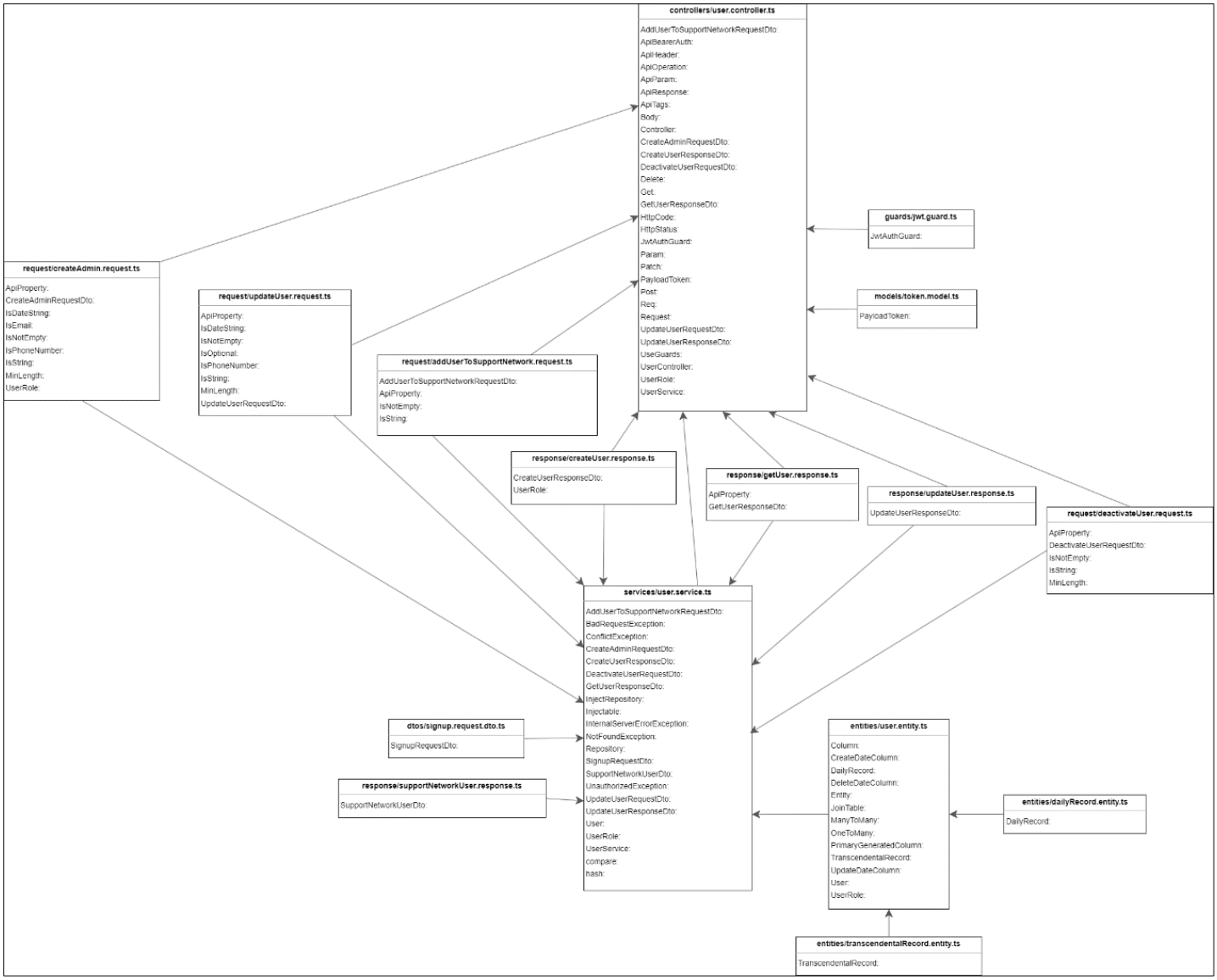


Imagen 24 Imagen diagrama de clases – Modulo de registro de emociones (NestJS)

Módulo de Backend Administrativo (Go)

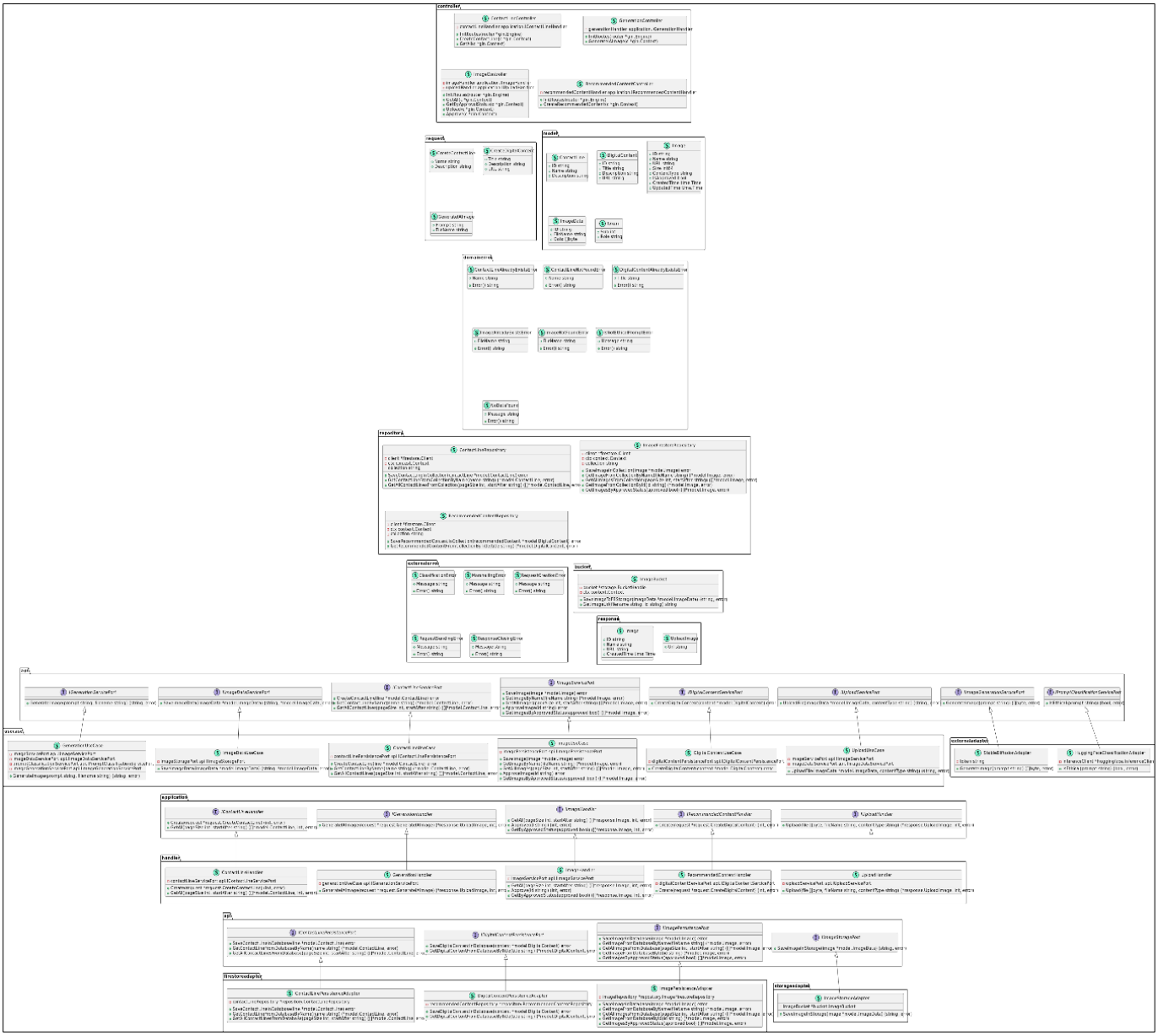


Imagen 25 Imagen diagrama de clases – Modulo Backend Administrativo (Go).

Visualizar diagrama mejor calidad.

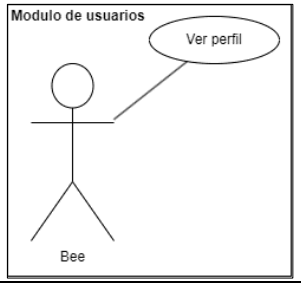
Requerimientos Funcionales y Casos de uso

Identificador	Especificación del requerimiento	Clase	
REQ – 1.1	El sistema debe permitir a los usuarios ver la información detallada de su propio perfil y, en función de la configuración de privacidad del perfil consultado, también visualizar la información de perfil de otros usuarios.	**Interfaz**	**Proceso** x
Restricciones	1. La visualización de información de perfil puede estar limitada por la autenticación por medio del JWT 2. Usuarios no autenticados o usuarios que no han iniciado sesión en el sistema pueden estar limitados y, por ende, no pueden visualizar ningún perfil. 3. Los diferentes roles de usuario pueden tener diferentes niveles de acceso a la información de la aplicación.		

Caso de uso:

Descripción: Este caso de uso describe el proceso mediante el cual un usuario puede visualizar la información de perfil, su propio perfil
Actores: Bee
Precondiciones: 1. El usuario está autenticado en el sistema
Flujo Normal: 1. El usuario accede a la página de perfil 2. El sistema muestra la información de perfil del usuario, incluyendo detalles como nombre, dirección de correo electrónico y otros datos relevantes. 3. Si el usuario está visualizando su propio perfil, tiene la opción de editar la información visible haciendo clic en el botón de edición correspondiente.
Flujo Alternativo: 1. En el paso 2 del flujo normal se podría dar el caso de que no se pueda acceder al api 2. El sistema mostrara un mensaje de error de conexión 3. El usuario es regresado al home de la aplicación. 4. Si se puede conectar al API el sistema mostrara el contenido del registro seleccionado.
Postcondiciones: 1. El usuario ha visualizado la información de perfil.

Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 1.2	El sistema debe garantizar la persistencia de la autenticación para los usuarios, permitiéndoles permanecer autenticados durante un período de tiempo específico después de iniciar sesión. Esta funcionalidad mejora la experiencia del usuario al evitar la necesidad de volver a iniciar sesión repetidamente en un corto período de tiempo.	**Interface**	**Proceso X**
Restricciones	1. El tiempo debe estar limitado para reducir la exposición al riesgo en caso de que un dispositivo o sesión sea comprometido. 2. El token debe tener un secret para que sea seguro. 3. En el token debe ir información como Id del usuario.		

Caso de uso:

Descripción: Este caso de uso describe cómo el sistema garantiza la persistencia de la autenticación para los usuarios, permitiéndoles permanecer autenticados durante un período de tiempo específico después de iniciar sesión. Esta funcionalidad mejora la experiencia del usuario al evitar la necesidad de volver a iniciar sesión repetidamente en un corto período de tiempo.

Actores: Bee

Precondiciones:

1. El usuario está autenticado en el sistema
2. El sistema ha generado un token de autenticación valido para el usuario.

Flujo Normal:

1. El sistema verifica el tiempo restante en el token de autenticación del usuario.
2. Si el tiempo restante es suficiente para mantener al usuario autenticado, el sistema permite al usuario continuar utilizando la aplicación sin necesidad de volver a autenticarse.
3. Si el tiempo restante en el token de autenticación es insuficiente, el sistema requerirá al usuario que vuelva a autenticarse antes de continuar.

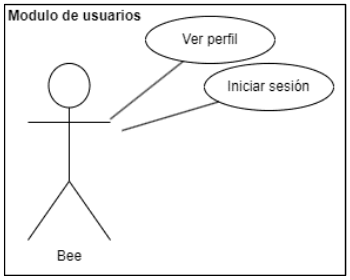
Flujo Alternativo:

1. El usuario sigue autenticado mientras el tiempo en el token de autenticación sea suficiente.

Postcondiciones:

1. El usuario ha visualizado la información de perfil.

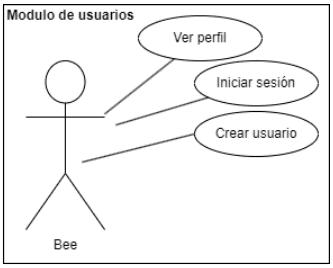
Diagrama del caso de uso:



Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 1.4	El sistema debe proporcionar al usuario la posibilidad de crear su cuenta de usuario. Este proceso de ser intuitivo y seguro que solicite información esencial para la creación de la cuenta.	**Interface**	**Proceso** x
Restricciones	1. La interfaz de usuario debe proporcionar una experiencia fluida y amigable para los usuarios 2. El frontend debe validar que el formato del correo electrónico ingresado sea válido antes de enviar la solicitud. 3. Toda la comunicación entre el frontend y el backend durante el proceso de creación de usuario debe estar cifrada mediante HTTPS		

Caso de uso:

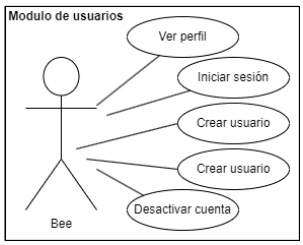
Descripción: Este caso de uso describe cómo un usuario puede utilizar la interfaz de usuario en el frontend para crear una nueva cuenta de usuario. El proceso de creación de usuario debe ser intuitivo y seguro, solicitando información esencial para la creación de la cuenta.
Actores: Bee
Precondiciones: 1. El usuario ha accedido a la interfaz de creación de usuario en el frontend. 2. El usuario proporciona información válida y esencial requerida para la creación de la cuenta.
Flujo Normal: 1. El usuario accede a la interfaz de creación de usuario. 2. El sistema muestra un formulario de registro que solicita información esencial, como nombre, dirección de correo electrónico y contraseña. 3. El usuario completa el formulario ingresando la información requerida. 4. El sistema valida que la dirección de correo electrónico tenga un formato válido antes de enviar la solicitud. 5. El usuario envía la solicitud de registro haciendo clic en el botón correspondiente.
Flujo Alternativo:

1. En el punto 4 si la dirección de correo electrónico no tiene un formato válido, el sistema muestra un mensaje de error al usuario y solicita una dirección de correo electrónico válida.
Postcondiciones: 1. El usuario recibe una confirmación de registro exitoso en la pantalla.
Diagrama del caso de uso: 

Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 1.6	El sistema debe proporcionar una interfaz de usuario que permita la desactivación de una cuenta. Esto incluye mostrar claramente si un usuario ha sido desactivado, la fecha en que se realizó.	**Interface**	**Proceso** x
Restricciones	1. Solo el usuario al que pertenece el perfil tiene acceso a la desactivación de este. 2. Antes de permitir la desactivación, el sistema debe requerir una autenticación adicional del usuario. 3. El usuario debe recibir una confirmación explícita y clara sobre la acción que está a punto de realizar 4. Después de la desactivación, se debe restringir completamente el acceso al perfil del usuario. 5. La desactivación de usuarios debe ser registrada con detalles la fecha y la hora		

Caso de uso:

Descripción: Este caso de uso describe cómo un usuario puede desactivar su cuenta
Actores: Bee
Precondiciones: 1. El actor está autenticado en el sistema
Flujo Normal: 1. El sistema muestra la opción de "Desactivar Usuario" como parte de las acciones disponibles. 2. El usuario selecciona la opción "Desactivar Usuario". 3. El sistema solicita una autenticación adicional al usuario para confirmar la acción. 4. El usuario proporciona la autenticación adicional, como una contraseña o un PIN.

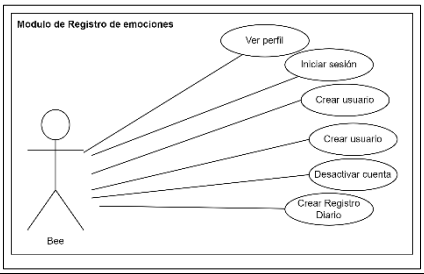
5. El sistema envía la solicitud de desactivación.
Flujo Alternativo: 1. En el punto 4, si el actor no proporciona la autenticación adicional correctamente, el sistema muestra un mensaje de error y la acción de desactivación se cancela.
Postcondiciones: 1. La cuenta del usuario se ha desactivado con éxito.
Diagrama del caso de uso: 

Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 2.1	El sistema debe permitirle al usuario crear un nuevo registro diario por día, en donde este último pueda agregar sus emociones a lo largo del día.	**Interface**	**Proceso X**
Restricciones	1. La interfaz debe ser intuitiva y fácil de usar 2. Se deben verificar los campos rellenados por el usuario para garantizar la integridad de los datos. 3. La interfaz debe adaptarse a diferentes tamaños de pantalla. 4. Se deben manejar de manera adecuada las respuestas de la API, incluyendo errores. 5. Se debe verificar que solo se puede hacer un registro por día.		

Caso de uso:

Descripción: El usuario podrá crear un registro diario en el cual podrá escribir las emociones que tuvo en el día en que se realizó el registro.
Actores: Usuario
Precondiciones: 1. El usuario debe haber iniciado sesión. 2. El usuario debe haber navegado a la interfaz de registro diario. 3. El usuario no tuvo que haber creado un registro diario
Flujo Normal: 1. El usuario inicia sesión

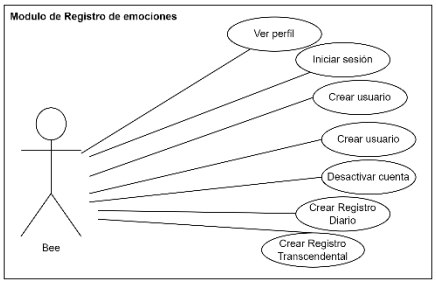
2. El usuario navega hasta la interfaz de registro diario 3. El usuario pulsa el botón de crear un nuevo registro 4. El sistema mostrara al usuario el formulario para la creación del registro 5. El sistema valida los datos ingresados. 6. El usuario confirma la creación del registro
Flujo Alternativo: 1. En el paso 5 del flujo normal, el sistema detecta que los datos ingresados no son válidos o están incompletos. 2. El sistema muestra un mensaje de error indicando los campos que requieren corrección. 3. El usuario corrige los campos con errores y vuelve a intentar la confirmación. 4. El sistema repite la validación de datos. 5. Si los datos son válidos, el sistema procede con la creación del registro.
Postcondiciones: El sistema deberá guardar el registro en la base de datos asociada al usuario.
Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 2.2	El sistema debe permitirle al usuario crear un nuevo registro transcendental, la cual le permitirá registrar las emociones desbordantes que sintió en un momento dado de su día a día.	**Interface**	**Proceso X**
Restricciones	1. La interfaz debe ser intuitiva y fácil de usar. 2. Se debe permitir que los usuarios creen registros ocasionales sin ninguna restricción de tiempo. 3. Se debe garantizar la seguridad de los registros, al momento del envió de la información. 4. Se deben verificar los campos rellenados, para garantizar la integridad de los registros.		

Caso de uso:

Descripción: En este caso de uso el usuario podrá crear un registro transcendental, en donde podrá guardar momentos específicos de su día a día (sin importar el día) donde hubo una emoción desbordante y/o trascendental.
Actores: Bee

Precondiciones: 1. El usuario debe haber iniciado sesión 2. El usuario debe haber navegado a la interfaz de registros.
Flujo Normal: 1. El usuario inicia sesión 2. El usuario navega hasta la interfaz de registros 3. El usuario pulsa el botón de crear un nuevo registro 4. El sistema mostrara al usuario el formulario para la creación del registro 5. El usuario pulsa el botón indicar que el registro será uno transcendental. 6. El sistema valida los datos ingresados. 7. El usuario confirma la creación del registro.
Flujo Alternativo: 1. En el paso 5 del flujo normal, el usuario puede olvidarlo crearlo 2. El sistema muestra un mensaje de error indicando los campos que requieren corrección. 3. El usuario corrige los campos con errores y vuelve a intentar la confirmación. 4. El sistema repite la validación de datos. 5. Si los datos son válidos, el sistema procede con la creación del registro.
Postcondiciones: El sistema deberá guardar el registro en la base de datos asociada al usuario.
Diagrama del caso de uso: 

Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 2.3	El sistema debe permitir al usuario seleccionar las emociones en cualquiera de los dos registros que se pueden hacer por medio de un selector, la cual dará información sobre las emociones tanto primarios como secundarias.	**Interface**	**Proceso** X
Restricciones	1. Se debe garantizar la facilidad de uso, de la ruleta. 2. La información proporcionada por el sistema sobre las emociones debe ser precisa. 3. Se deben categorizar de manera precisa las emociones que se encontraran en la ruleta.		

Caso de uso:

Descripción: En este caso de uso se le permitirá a través de la ruleta de emociones, seleccionar las emociones que sintió en un momento dado, esta ruleta estará disponible a la hora de hacer cualquier registro (diario o transcendental).

Actores: Bee

Precondiciones:

1. El usuario debe haber iniciado sesión
2. El usuario debe haber navegado a la interfaz de registro diario.
3. El usuario debe haber navegado a la interfaz de crear un nuevo registro.

Flujo Normal:

1. El usuario debe haber iniciado sesión
2. El usuario debe haber navegado a la interfaz de registro.
3. El usuario debe haber navegado a la interfaz de crear un nuevo registro.
4. El usuario al seleccionar el campo de “emociones”, se desplegará la ruleta de emociones
5. El usuario elige la emoción a guardar en el registro
6. El usuario rellena todos los campos del registro
7. El usuario confirma la creación de registro.
8. Se crea el registro.

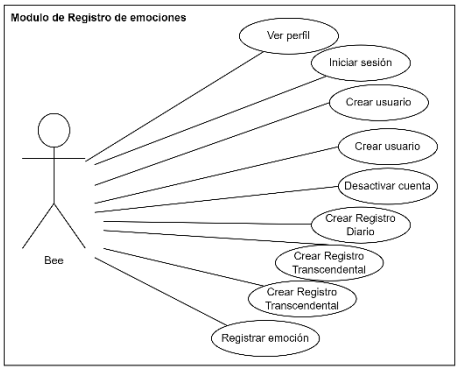
Flujo Alternativo:

6. En el paso 5 del flujo normal, el sistema detecta que no se seleccionó emociones para el registro.
7. El sistema arrojara un error indicando la falta de información en este campo.
8. El usuario corrige el error y vuelve a hacer la confirmación del registro.
9. Se crea el registro.

Postcondiciones:

El sistema deberá guardar el registro (ya sea registro diario o transcendental) en la base de datos asociada al usuario.

Diagrama del caso de uso:

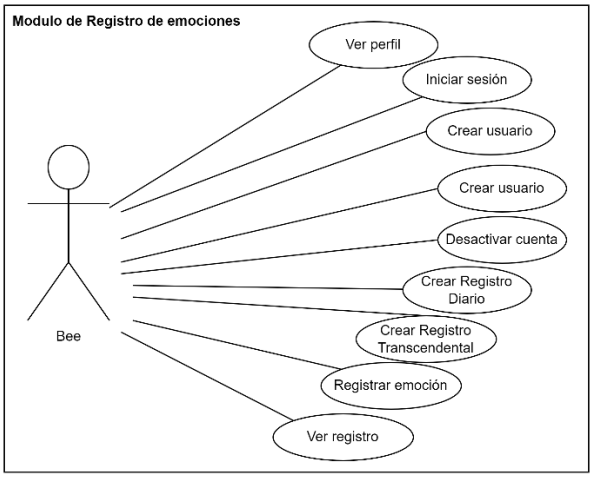


Identificador	Especificación del requerimiento	Clase

REQ – 2.5	El sistema debe proporcionar una vista para visualizar un registro tanto diario como trascendental, según la elección del usuario.	**Interface**	**Proceso** X
Restricciones	1. Se debe garantizar la compatibilidad con diferentes tamaños de pantalla. 2. Se debe verificar que el rendimiento del componente visual para que la experiencia del usuario sea lo más fluida y gratificante posible. 3. La vista debe darle un formato a la visualización de los datos traídos del backend.		

Caso de uso:

Descripción: El sistema deberá garantizar que el usuario podrá abrir el registro diario que haya selección para ver todo el contenido guardado en tal registro.
Actores: Bee
Precondiciones: 1. El usuario debe haber iniciado sesión 2. El usuario debe haber navegado a la interfaz de registro diario. 3. El usuario debe haber seleccionado el registro que quiere ver.
Flujo Normal: 1. El usuario debe haber iniciado sesión 2. El usuario debe haber navegado a la interfaz de registro diario. 3. El usuario debe haber seleccionado el registro que quiere ver de la lista de registros 4. El sistema le muestra al usuario el contenido del registro seleccionado
Flujo Alternativo: 1. En el paso 3 del flujo normal se podría dar el caso de que no se pueda acceder al api 2. El sistema mostrara un mensaje de error de conexión 3. El usuario es regresado a la lista de registros diarios. 4. Si se puede conectar al API el sistema mostrara el contenido del registro seleccionado.
Postcondiciones: 5. El usuario cierra el registro seleccionado para ver los demás o seguir navegando en la aplicación.
Diagrama del caso de uso:



Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 2.7	El sistema debe proporcionar al usuario una forma de ver todos los registros tanto diarios como ocasionales, previamente publicados o añadidos por el usuario, estos serán visualizarán de forma compacta, con información básica del registro como por ejemplo la fecha.	**Interface**	**Proceso X**
Restricciones	1. La visualización del historial debe ser compacta, mostrando información relevante para reconocer el registro como por ejemplo el titulo y la fecha. 2. Los registros deben ser visualizados de forma ordenada según la fecha o el orden de creación. 3. Se debe adaptar a diferentes tamaños de dispositivos. 4. Se debe garantizar que la navegación dentro del historial sea intuitiva para el usuario.		

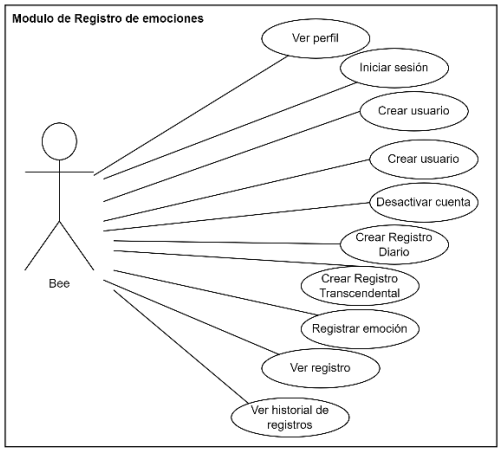
Descripción: El usuario podrá visualizar el historial de todos los registros hechos a lo largo de su interacción con la aplicación.
Actores: Bee
Precondiciones: 1. El usuario debe haber iniciado sesión 2. El usuario debe haber navegado a la interfaz de registro diario.
Flujo Normal: 1. El usuario debe haber iniciado sesión 2. El usuario debe haber navegado a la interfaz de registro diario. 3. El sistema mostrar al usuario los registros hechos previamente
Flujo Alternativo: 1. El paso 3 del flujo normal puede darse la posibilidad en que se no se pueda conectar al API

2. El sistema mostrara un mensaje de error de conexión al usuario
3. El usuario será redirigido a la interfaz principal
4. Si ahora es posible la conexión con el API y el usuario ingresa a la interfaz de diario emocional, el sistema procederá a mostrar los registros previamente hechos.

Postcondiciones:

1. El usuario cierra el historial de registros, para utilizar otros módulos del sistema.

Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 2.9	El sistema debe proporcionar a un usuario la forma de ver los registros de otros usuarios, por lo tanto, se hará un endpoint con el cual, al recibir una petición, traiga todos los registros diarios que ha hecho el usuario.	**Interface**	**Proceso X**
Restricciones	1. Se debe verificar la identidad del usuario. 2. El usuario debe confirmar su consentimiento para poder compartir la información registrada en la aplicación.		

Caso de uso:

Descripción: En este caso de uso se describe la interacción del usuario para su intervención para ver su historial de registros de otro usuario

Actores: Bee

Precondiciones:

1. El cliente recibe el evento para buscar los registros de un usuario (con la información necesaria)

Flujo Normal:

1. El cliente recibe la información del paciente para la búsqueda de sus registros diarios.
2. El cliente contacta con la API, haciendo una petición al endpoint correspondiente con la información necesaria.

3. La API recibe la petición del cliente y retorna la respuesta correspondiente.
4. El cliente recibe la respuesta a la petición y posteriormente es mostrada al usuario.
5. Si la respuesta es satisfactoria el cliente mostrara todos los registros creados por el usuario al profesional.

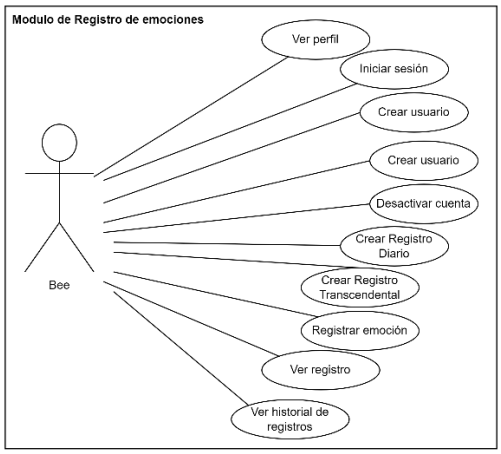
Flujo Alternativo:

1. En el paso 4 puede llegar a suceder que la respuesta de la API sea un error porque no se encontró al usuario.
2. El cliente muestra la respuesta de error y señala el campo erróneo.
3. El cliente rectifica la información y vuelve enviar la información al cliente.
4. El cliente contacta con la API, haciendo una petición al endpoint correspondiente con la información necesaria.
6. La API recibe la petición del cliente y retorna la respuesta correspondiente.
7. El cliente recibe la respuesta a la petición y posteriormente es mostrada al usuario.
8. Si la respuesta es satisfactoria el cliente mostrara todos los registros creados por el usuario.

Postcondiciones:

1. El usuario cierra los registros del usuario una vez se haya acabado la consulta.
2. El cliente deberá regresar al formulario, para buscar de nuevo a un usuario.

Diagrama del caso de uso:

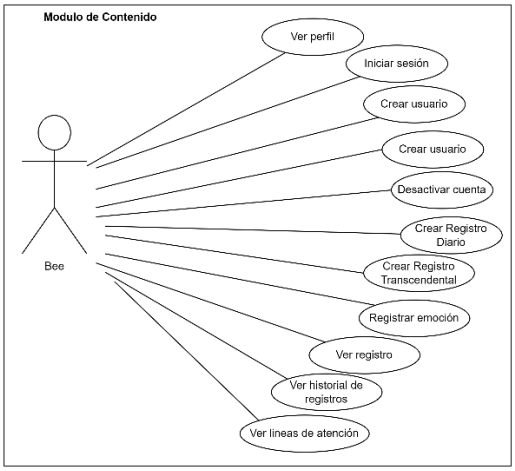


Identificador	Especificación del requerimiento	Clase	
REQ – 3.2	El sistema debe crear un endpoint en el backend que permita a los usuarios consultar las líneas de atención disponibles. Cuando un usuario solicite esta consulta desde el frontend, se enviará una solicitud al backend a través de este nuevo endpoint. El backend deberá proporcionar una lista de las líneas de atención disponibles, incluyendo información relevante como números de contacto de emergencia, acceso rápido a estas líneas y cualquier otra información necesaria.	**Interface**	**Proceso X**

Restricciones	1. El nuevo endpoint debe estar protegido y requerir autenticación por parte del cliente para acceder a la funcionalidad de consulta de líneas de atención. 2. El backend debe responder a las solicitudes de consulta de manera eficiente y dentro de un tiempo razonable. 3. La lista de líneas de atención proporcionada por el endpoint debe estar actualizada y reflejar la información más reciente. 4. La información proporcionada debe ser clara y concisa, facilitando la comprensión y el acceso rápido a las líneas de atención de emergencia.

Caso de uso:

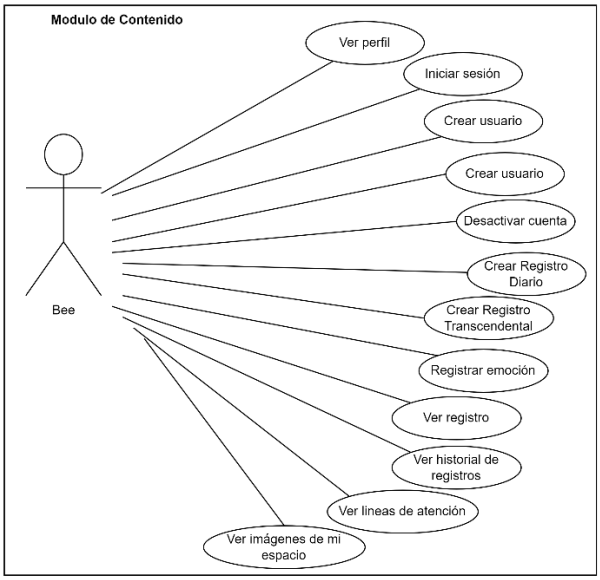
Descripción: Este caso de uso describe cómo un usuario puede consultar las líneas de atención disponibles a través del nuevo endpoint creado en el backend. Cuando un usuario solicita esta consulta desde el frontend, se envía una solicitud al backend a través de este endpoint, y el backend proporciona una lista actualizada de las líneas de atención, incluyendo información relevante como números de contacto de emergencia y cualquier otra información necesaria.
Actores: Bee
Precondiciones: 1. El usuario debe estar autenticado en el sistema. 2. El sistema debe contar con el nuevo endpoint en el backend para la consulta de líneas de atención.
Flujo Normal: 1. El cliente solicita la consulta de líneas de atención desde el frontend. 2. El backend proporciona una lista actualizada de las líneas de atención, incluyendo información como números de contacto de emergencia y cualquier otra información necesaria. 3. El cliente recibe la lista de líneas de atención en el frontend.
Flujo Alternativo: 1. En el punto 3, si el usuario no está autenticado en el sistema, el caso de uso no puede continuar y se muestra un mensaje de error.
Postcondiciones: 1. El cliente ha realizado la petición al endpoint con éxito y ha recibido la información relevante.
Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 3.4	El sistema debe proporcionar un endpoint de API que permita a los usuarios visualizar imágenes de mi espacio. Estas imágenes están destinadas a ayudar a los usuarios a relajarse y reducir el estrés. El endpoint debe proporcionar un acceso seguro y eficiente a estas imágenes para los usuarios autorizados.	**Interface**	**Proceso X**
Restricciones	1. Todas las imágenes proporcionadas por el endpoint deben ser apropiadas para todas las edades y culturas. 2. El endpoint debe enviar el url de las imágenes		

Caso de uso:

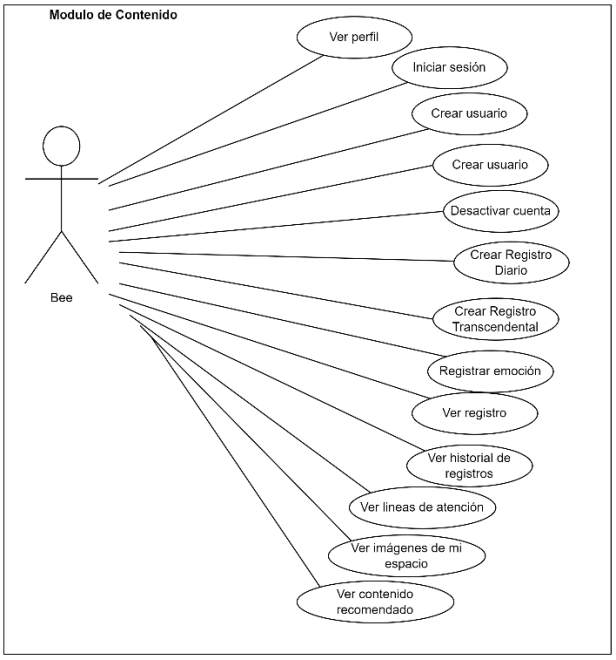
Descripción: Este caso de uso describe cómo un usuario puede visualizar imágenes de mi espacio a través de un endpoint de API proporcionado por el sistema. Estas imágenes están destinadas a ayudar a los usuarios a relajarse y reducir el estrés. El endpoint debe proporcionar un acceso seguro y eficiente a estas imágenes para los usuarios autorizados.
Actores: Bee
Precondiciones: 1. El usuario debe estar autenticado en el sistema. 2. El endpoint de API de visualización de imágenes de relajación está disponible.
Flujo Normal: 1. El usuario envía una solicitud al endpoint de API de visualización de imágenes de relajación. 2. El backend consulta en la base de datos las urls de las imágenes. 3. El sistema procesa la solicitud y envía los links de las imágenes de mi espacio.
Flujo Alternativo: 2. En el punto 1, si el usuario no está autenticado en el sistema, el caso de uso no puede continuar y se muestra un mensaje de error.
Postcondiciones:

1. El cliente ha realizado la petición al endpoint con éxito y ha recibido la información relevante.
Diagrama del caso de uso: 

Identificador	**Especificación del requerimiento**	**Clase**	
REQ – 3.6	El sistema debe proporcionar un endpoint de API que permita a los usuarios ver **Modulo de contenido** recomendado externo como podcasts, vi de divulgación psicologica. Este contenido está destinadas a ayudar a los usuarios a conocer más acerca de la psicología. El endpoint debe proporcionar un acceso seguro y eficiente a este contenido para los usuarios autorizados.	**Interface**	**Proceso X**
Restricciones	1. Todo el contenido proporcionado por el endpoint debe ser apropiadas para todas las edades y culturas. 2. El endpoint debe enviar el url del contenido recomendado		

Caso de uso:

Descripción: Este caso de uso describe cómo un usuario puede visualizar contenido recomendado que es traído desde una API y mostrado en la interfaz de usuario del frontend. El sistema debe permitir que los usuarios naveguen y vean el contenido de manera fácil y eficiente.
Actores: Bee
Precondiciones: 1. El usuario debe estar autenticado en el sistema. 2. La API que proporciona el contenido recomendado debe estar disponible y accesible. 3. La interfaz de usuario del frontend debe estar correctamente integrada con la API.

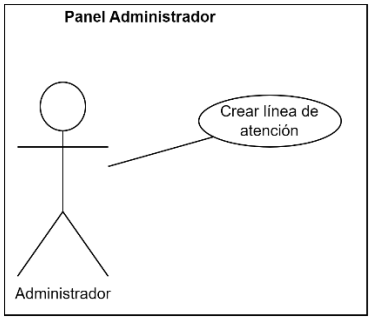
Flujo Normal: 1. El usuario accede a la interfaz de usuario del frontend y navega hacia la sección de contenido recomendado. 2. El cliente (frontend) realiza una solicitud a la API para obtener el contenido recomendado. 3. La API procesa la solicitud y envía el contenido recomendado al cliente. 4. El cliente recibe el contenido y lo muestra en la interfaz de usuario. 5. El usuario navega y visualiza el contenido recomendado de manera fácil y eficiente.
Flujo Alternativo: 1. En el punto 1, si el usuario no está autenticado en el sistema, el caso de uso no puede continuar y se muestra un mensaje de error. 2. En el punto 2, si la API no está disponible o no responde, el caso de uso no puede continuar y se muestra un mensaje de error al usuario.
Postcondiciones: 1. El cliente ha realizado la petición al endpoint con éxito y ha recibido las URLs del contenido recomendado.
Diagrama del caso de uso: 

Identificador	Especificación del requerimiento	Clase	
REQ – 3.8	El sistema debe permitir a los administradores crear las líneas de atención que serán visualizadas por los usuarios en la pantalla de emergencia. Esto incluye definir la información detallada de las líneas de atención, como números de teléfono, correos electrónicos, y otros métodos de contacto,	**Interface**	**Proceso X**

	que serán almacenados en la base de datos y accesibles a través de un endpoint de la API.		
Restricciones	1. El administrador debe poder gestionar la información de las líneas de atención de manera segura y eficiente a través de una interfaz administrativa. 2. La información de las líneas de atención debe ser actualizable y debe reflejarse inmediatamente en la pantalla de emergencia cuando sea modificada. 3. Los cambios realizados por el administrador deben ser sincronizados con la base de datos y estar disponibles para los usuarios a través del endpoint de la API.		

Caso de uso:

Descripción: Este caso de uso describe cómo un administrador del sistema puede crear las líneas de atención que serán mostradas en la pantalla de emergencia para los usuarios. El administrador puede definir detalles como números de teléfono, correos electrónicos y otros métodos de contacto, los cuales se almacenarán y serán accesibles a través de un endpoint de la API.
Actores: Bee
Precondiciones: 1. El administrador debe estar autenticado en el sistema con los permisos adecuados para gestionar las líneas de atención. 2. La interfaz administrativa debe estar disponible y conectada a la base de datos y al endpoint de la API.
Flujo Normal: 1. El administrador accede al sistema y navega a la sección de gestión de líneas de atención. 2. El administrador selecciona la opción para crear una nueva línea de atención o editar una existente. 3. El administrador ingresa o modifica la información detallada, como números de teléfono, correos electrónicos y otros métodos de contacto. 4. El administrador guarda los cambios, y el sistema almacena la información en la base de datos. 5. El sistema sincroniza los datos con el endpoint de la API, haciendo que la información esté disponible para su visualización en la pantalla de emergencia.
Flujo Alternativo: 1. En el punto 4, si hay un problema al guardar la información en la base de datos, el sistema retorna un mensaje de error al cliente y sugiere verificar la conexión o los datos ingresados.
Postcondiciones: 1. Las nuevas líneas de atención o las modificaciones realizadas por el administrador están disponibles para los usuarios a través de la pantalla de emergencia. 2. La información actualizada es accesible mediante el endpoint de la API.
Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 3.10	El sistema debe permitir a los administradores subir imágenes a "Mi Espacio" para que sean visualizadas por los usuarios. Estas imágenes deben ser almacenadas en la base de datos y estar disponibles a través de un endpoint de la API para su visualización en la interfaz de usuario del frontend.	**Interface**	**Proceso X**
Restricciones	1. Las imágenes deben ser cargadas de manera segura y eficiente a través de la interfaz administrativa. 2. Las imágenes subidas deben estar optimizadas para una carga rápida y eficiente cuando sean visualizadas por los usuarios. 3. Las imágenes deben estar disponibles para los usuarios inmediatamente después de ser subidas y almacenadas. 4. El sistema debe garantizar que solo los administradores autorizados puedan subir imágenes a "Mi Espacio".		

Caso de uso:

Descripción: Este caso de uso describe cómo un administrador puede subir imágenes a "Mi Espacio" utilizando la interfaz administrativa. Las imágenes subidas serán almacenadas en la base de datos y estarán disponibles para los usuarios a través de un endpoint de la API.
Actores: Bee
Precondiciones: 1. El administrador debe estar autenticado en el sistema con los permisos adecuados para subir imágenes. 2. La interfaz para subir imágenes debe estar disponible y conectada a la base de datos y al endpoint de la API.
Flujo Normal: 1. El administrador accede al sistema y navega a la sección de "Mi Espacio" en la interfaz administrativa. 2. El administrador selecciona la opción para subir una nueva imagen. 3. El administrador elige una imagen desde su dispositivo y la sube a través de la interfaz. 4. El sistema valida el formato y tamaño de la imagen para asegurar que cumpla con los requisitos. 5. La imagen es almacenada en la base de datos.

6. El sistema sincroniza los datos y pone la imagen a disposición de los usuarios a través del endpoint de la API.
7. La imagen subido está disponible para ser visualizada en la interfaz de usuario del frontend.
8. datos con el endpoint de la API, haciendo que la información esté disponible para su visualización en la pantalla de emergencia.

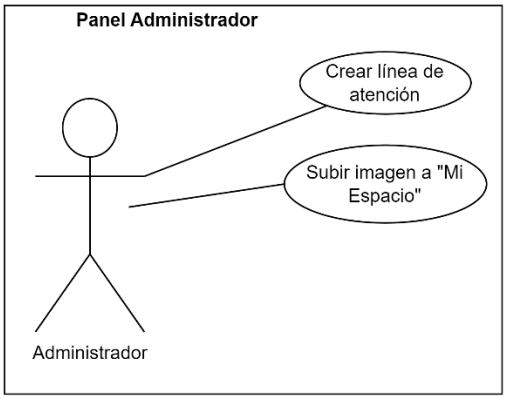
Flujo Alternativo:

1. En el punto 4, si la validación de la imagen falla (por ejemplo, si el formato o tamaño no es compatible), el sistema muestra un mensaje de error y solicita al administrador que cargue una imagen válida.

Postcondiciones:

1. La imagen subida por el administrador está disponible en "Mi Espacio" para que los usuarios la visualicen a través de la interfaz de usuario de.
2. La información actualizada es accesible mediante el endpoint de la API.

Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase	
REQ – 3.12	El sistema debe manejar de forma eficiente la generación de imágenes mediante IA a partir de prompts de texto proporcionados por los administradores. El backend debe encargarse de procesar los prompts, generar las imágenes, almacenarlas en la base de datos y asegurar que estén disponibles para su visualización a través de un endpoint de la API.	**Interface**	**Proceso x**
Restricciones	1. El backend debe validar automáticamente los prompts recibidos, asegurando que cumplan con las políticas de contenido antes de iniciar la generación de imágenes. 2. El proceso de generación de imágenes debe ser optimizado para utilizar los recursos del servidor de manera eficiente, evitando la sobrecarga. 3. Las imágenes generadas deben ser almacenadas en la base de datos de manera segura y rápida, con metadatos que faciliten su recuperación. 4. El sistema debe garantizar la disponibilidad inmediata de las imágenes generadas a través del endpoint de la API para los usuarios autorizados		

Caso de uso:

Descripción: Este caso de uso describe cómo el backend del sistema maneja el proceso de generación de imágenes a partir de prompts de texto proporcionados por los administradores. Involucra la validación del prompt, la interacción con el servicio de IA, el almacenamiento de la imagen generada, y la actualización de la API para permitir su acceso por los usuarios.

Actores: Administrador

Precondiciones:

1. El sistema debe estar correctamente configurado para interactuar con el servicio de IA y la base de datos.
2. El administrador debe haber enviado un prompt a través de la interfaz.

Flujo Normal:

1. El backend recibe el prompt de texto enviado por el administrador desde la interfaz.
2. El backend valida el prompt asegurando que cumpla con las políticas de contenido y formato establecidos.
3. Si el prompt es válido, el backend envía el prompt al servicio de IA para generar la imagen.
4. El servicio de IA procesa el prompt y genera la imagen.
5. El backend recibe la imagen generada y la almacena en la base de datos, junto con los metadatos correspondientes.
6. El backend actualiza el endpoint de la API para que la imagen generada esté disponible para su validación.

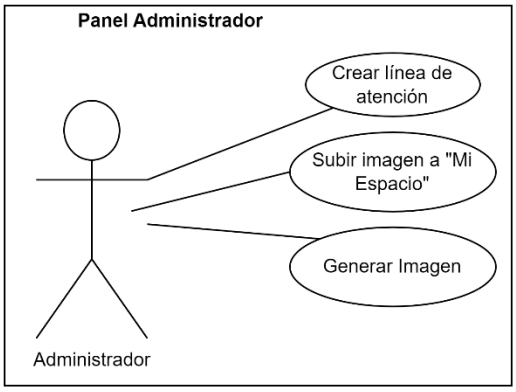
Flujo Alternativo:

1. En el punto 2, si el prompt no cumple con las políticas de contenido, el backend rechaza el proceso y notifica el fallo a la interfaz para que el administrador pueda corregirlo.
2. En el punto 4, si el servicio de IA no puede generar la imagen por alguna razón (por ejemplo, debido a un error en la IA), el backend maneja el error y notifica a la frontend para que se informe al administrador.

Postcondiciones:

1. La imagen generada está almacenada de manera segura en la base de datos y está disponible para los usuarios a través del endpoint de la API.

Diagrama del caso de uso:



Identificador	Especificación del requerimiento	Clase

REQ – 3.14	El sistema debe implementar un proceso que maneje la validación y aceptación de imágenes generadas antes de que estas estén disponibles en la aplicación "Our Hive". Este proceso debe gestionar la revisión de las imágenes por parte de los administradores, asegurando que solo las imágenes aprobadas se publiquen y sean accesibles por los usuarios.	**Interface**	**Proceso x**
Restricciones	1. El backend debe gestionar de manera segura la solicitud de validación de imágenes, incluyendo la carga, almacenamiento, y la actualización de su estado (aprobado o rechazado). 2. El sistema debe registrar automáticamente la decisión de los administradores (aprobación o rechazo) 3. Las imágenes rechazadas deben mantenerse en un estado inaccesible para los usuarios finales y no deben ser mostradas en la aplicación "Our Hive". 4. El backend debe actualizar el estado de las imágenes aprobadas, permitiendo su disponibilidad inmediata en la aplicación.		

Caso de uso:

Descripción: Este caso de uso describe cómo el backend del sistema gestiona el proceso de validación y aceptación de imágenes generadas, desde la revisión por parte de un administrador hasta la actualización del estado de la imagen en la base de datos, asegurando que solo las imágenes aprobadas estén disponibles para los usuarios.
Actores: Administrador
Precondiciones: 1. El sistema debe estar correctamente configurado para recibir las decisiones de validación del administrador. 2. El administrador debe haber enviado su decisión de aprobación o rechazo desde la interfaz de validación de imágenes.
Flujo Normal: 1. El backend recibe la solicitud de validación de una imagen por parte del administrador, incluyendo la decisión tomada (aprobación o rechazo). 2. El backend actualiza el estado de la imagen en la base de datos según la decisión del administrador. 3. Si la imagen es aprobada, el estado se actualiza a "Aprobado", y la imagen se marca como disponible para los usuarios en la aplicación "Our Hive". 4. Si la imagen es rechazada, mantiene la imagen en un estado inaccesible para los usuarios. 5. El backend registra la decisión del administrador junto con la fecha y hora en los registros del sistema. 6. El backend notifica a la interfaz de que la acción se ha completado correctamente.
Flujo Alternativo: 1. En el punto 1, si el backend no puede acceder a la base de datos o actualizar el estado de la imagen debido a un error técnico, el sistema debe enviar una notificación de error al administrador para que se pueda intentar nuevamente o resolver el problema.

Postcondiciones:

1. Las imágenes aprobadas están disponibles para los usuarios en la aplicación "Our Hive".
2. Las imágenes rechazadas permanecen almacenadas en la base de datos pero no son accesibles para los usuarios finales.

Diagrama del caso de uso:

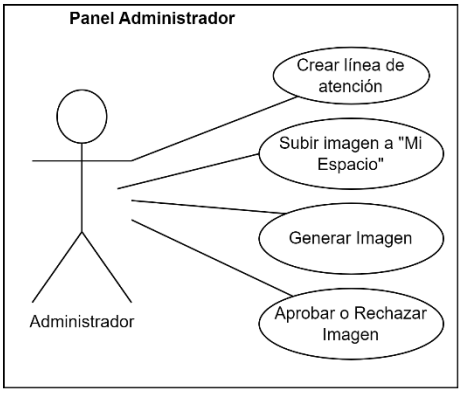


Tabla 10: Casos de uso

Diagramas de secuencia de alto nivel

Un Diagrama de Alto Nivel es una representación visual simplificada que proporciona una visión general de un sistema, proceso o proyecto. Se enfoca en aspectos clave sin entrar en detalles específicos, lo que permite a los stakeholders comprender rápidamente la estructura y el funcionamiento general.[https://creately.com/diagram/example/IoYpffOhu3x/diagrama-de-alto-nivel#:~:text=Un%20Diagrama%20de%20Alto%20Nivel,estructura%20y%20el%20funcionamiento%20general.], por lo cual el diagrama de secuencia fue muy útil en el proceso de desarrollo puesto que los programadores tienen una gran inclinación por diversos campos del desarrollo de software, el tener un diagrama en el cual podíamos evidenciar las funcionalidades y responsabilidades que tendría cada desarrollador, ayudo a tener un ágil y comunicativo proceso de desarrollo.

A continuación, podremos evidenciar los diagramas de secuencia útiles para el desarrollo y cumplimiento de los objetivos del proyecto.

Ver perfil

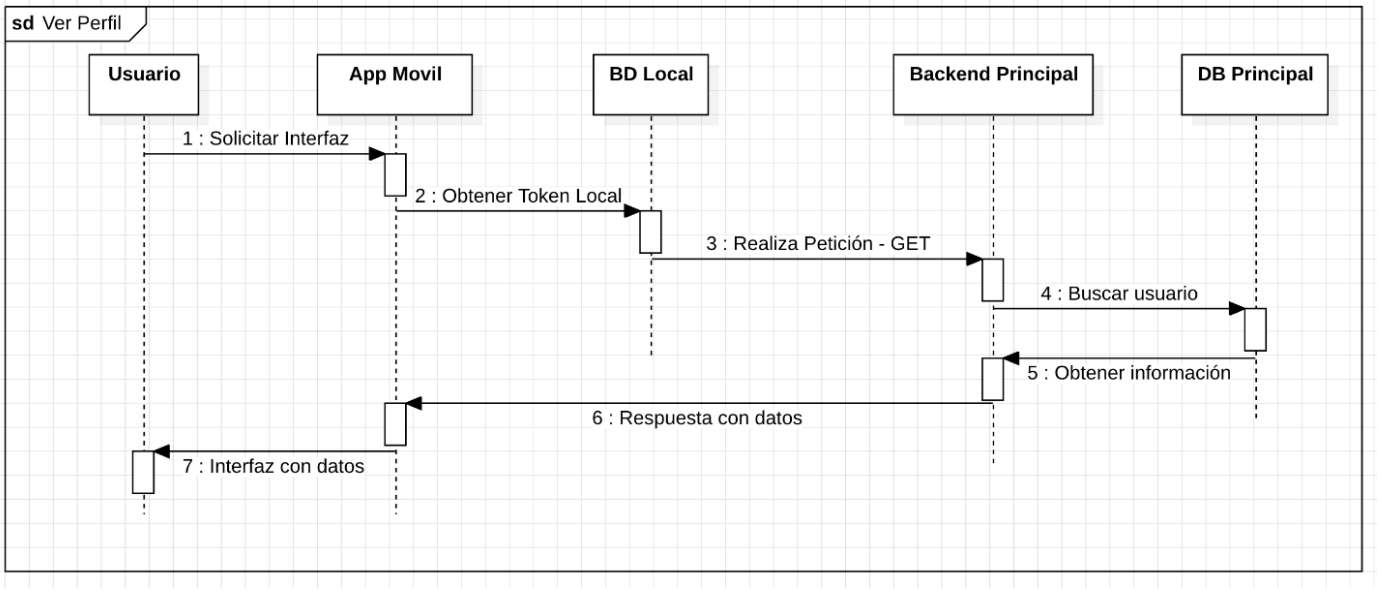


Imagen 26 Imagen diagrama de secuencia – Ver perfil.

Autenticación Local

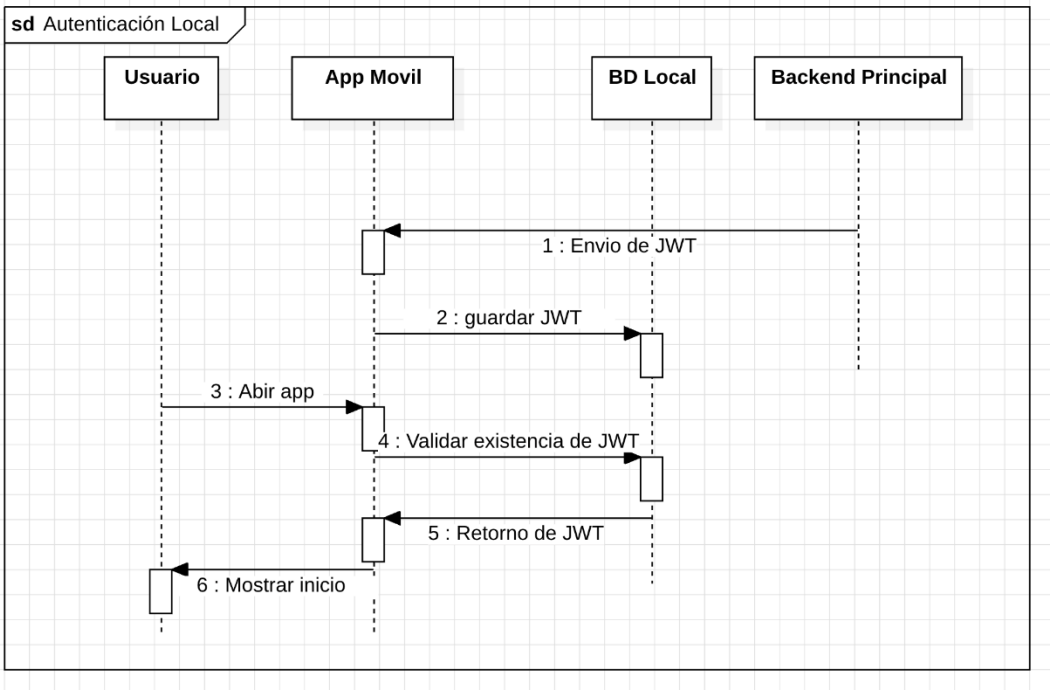


Imagen 27 Imagen diagrama de secuencia – Autenticación Local

.

Registro

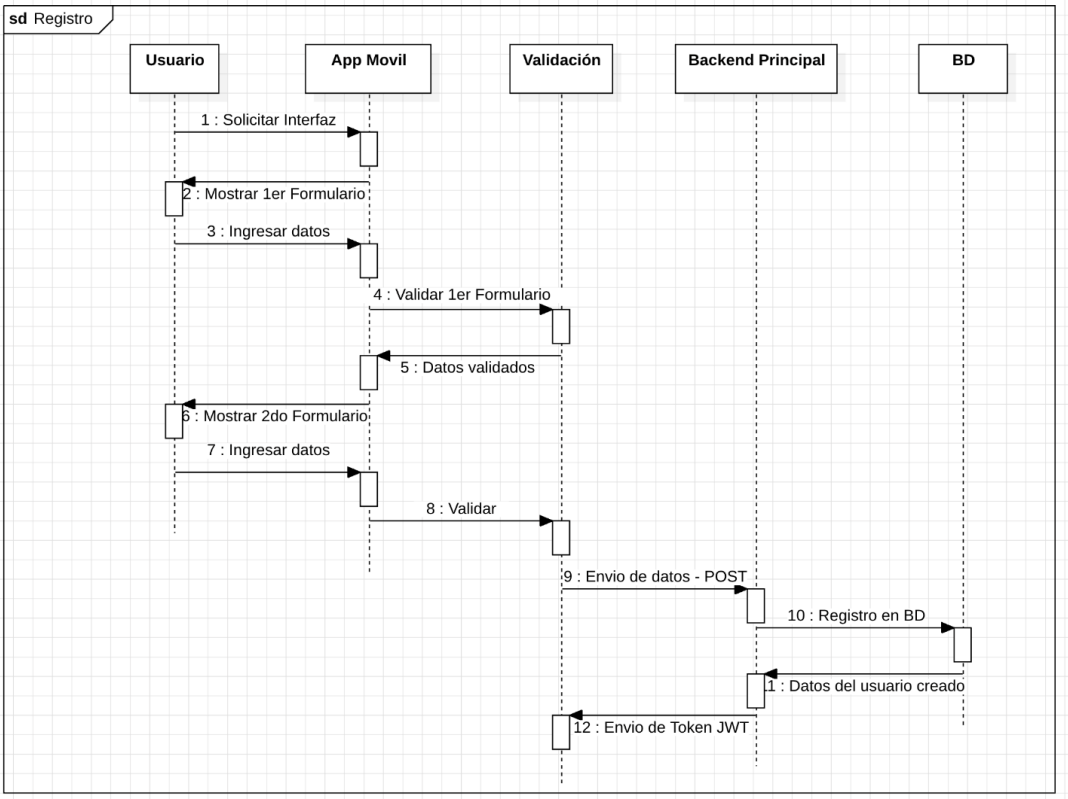


Imagen 28 Imagen diagrama de secuencia – Registro.

Deshabilitar Cuenta

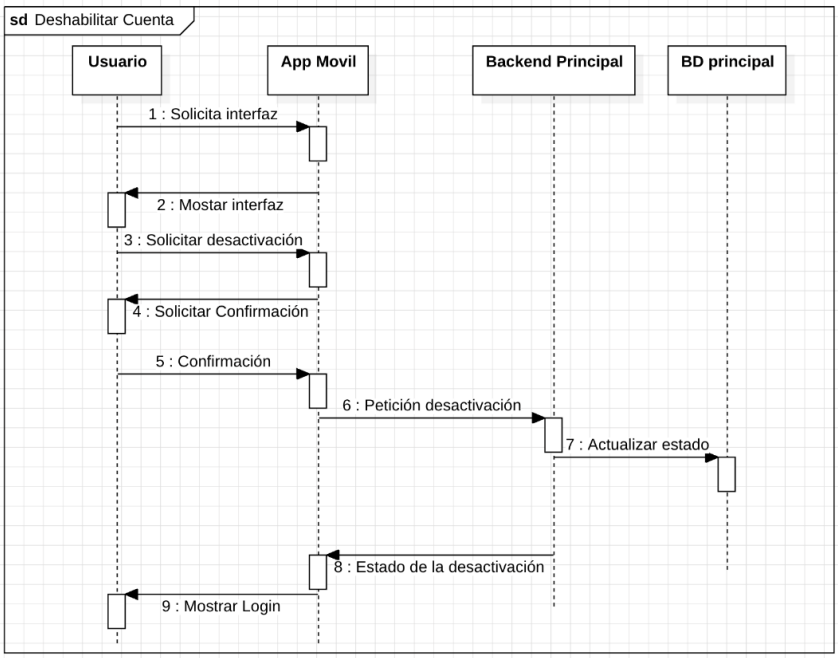


Imagen 29 Imagen diagrama de secuencia – Deshabilitar cuenta.

Ver Historial

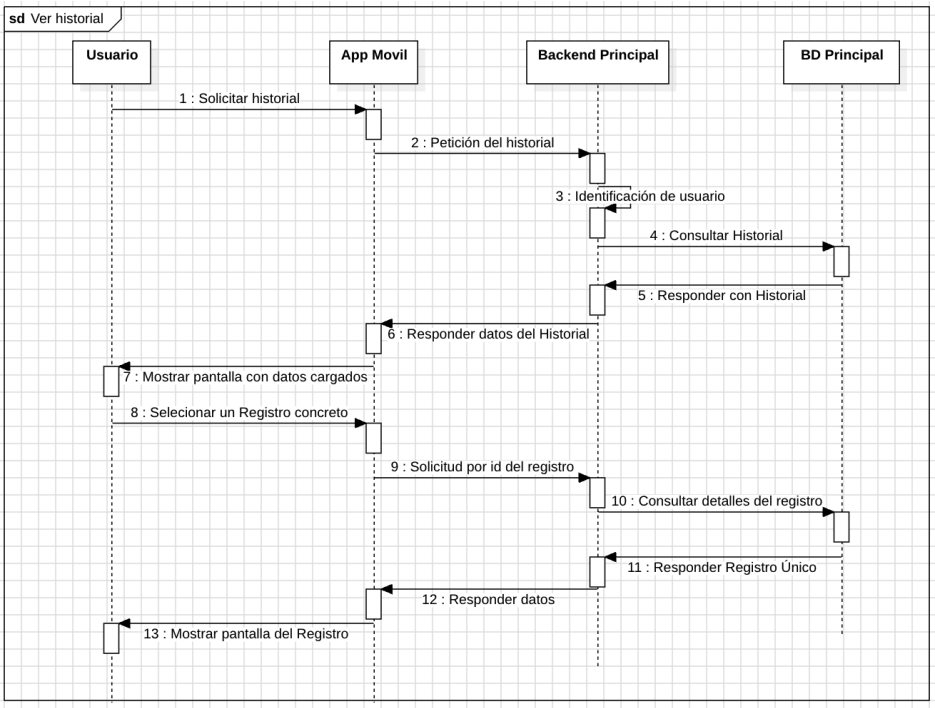


Imagen 30 Imagen diagrama de secuencia – Ver historial.

Visualizar Líneas de Atención

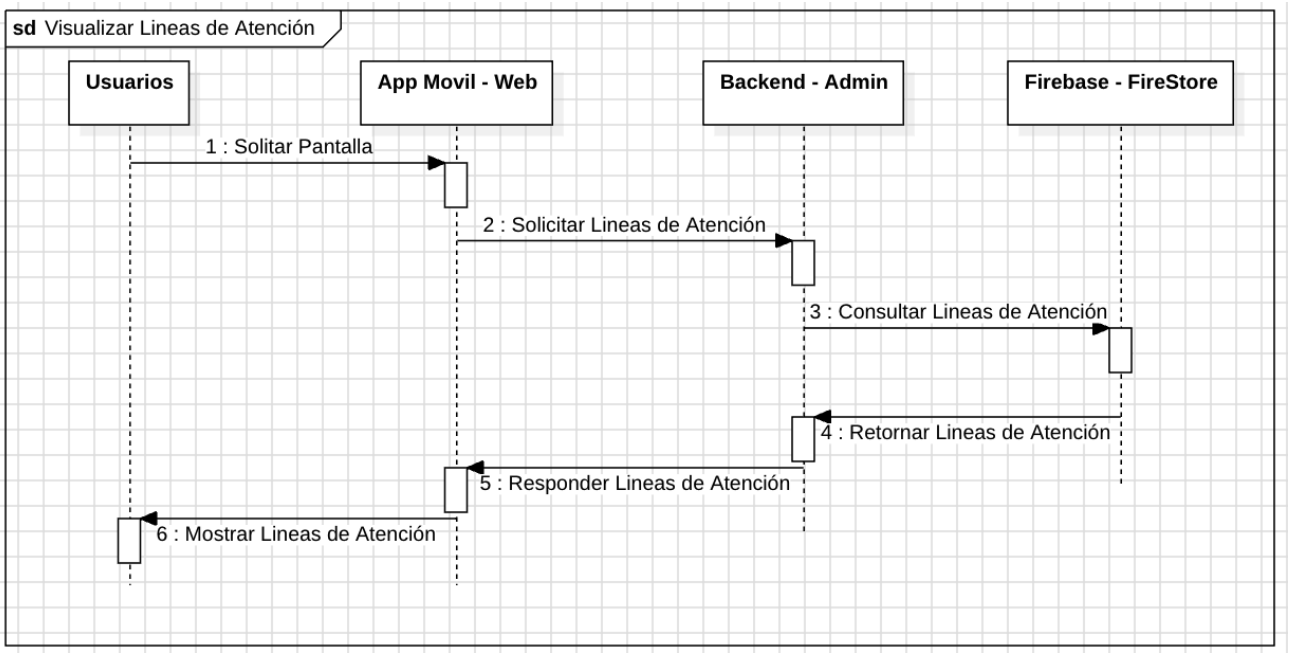


Imagen 31 Imagen diagrama de secuencia – Visualizar Líneas de Atención.

Visualizar My Space

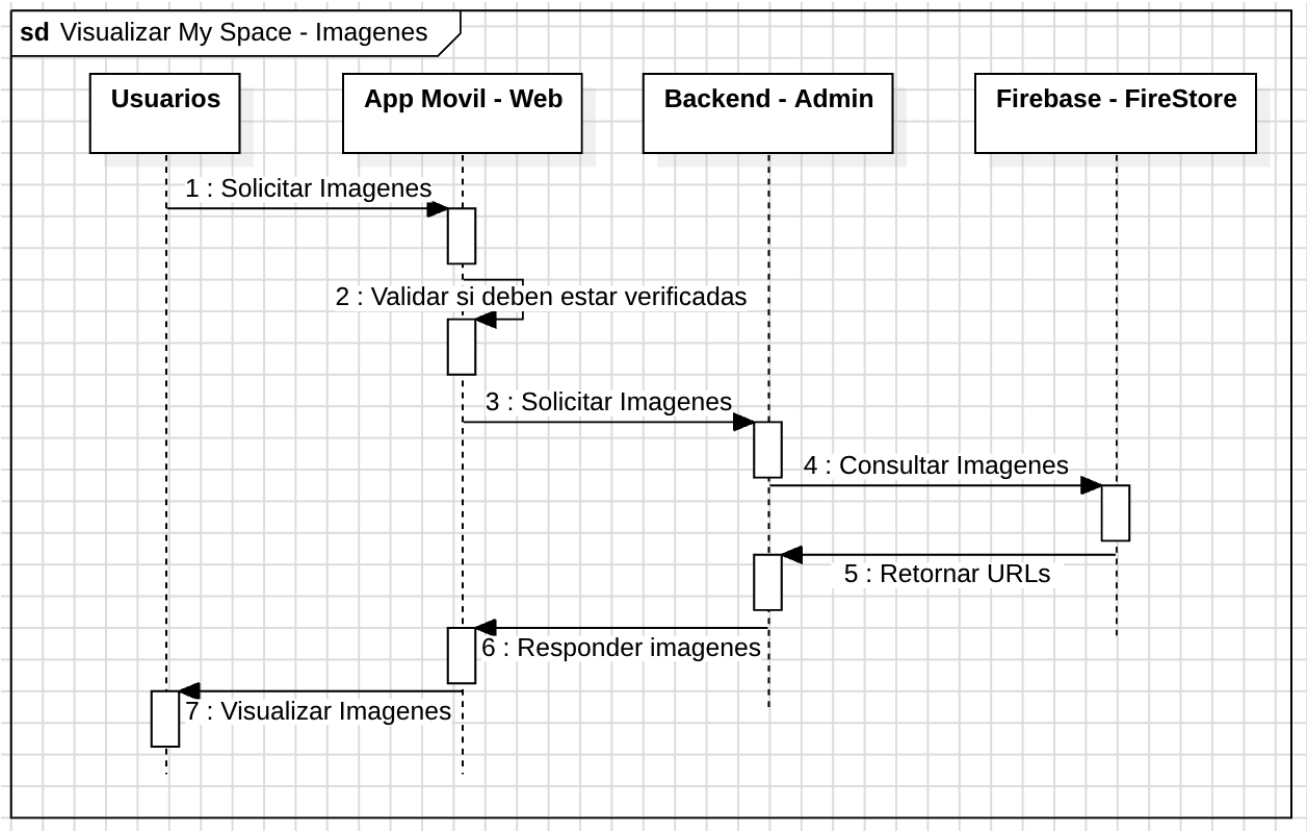


Imagen 32 Imagen diagrama de secuencia – Visualizar My Space.

Visualizar imágenes para My Space

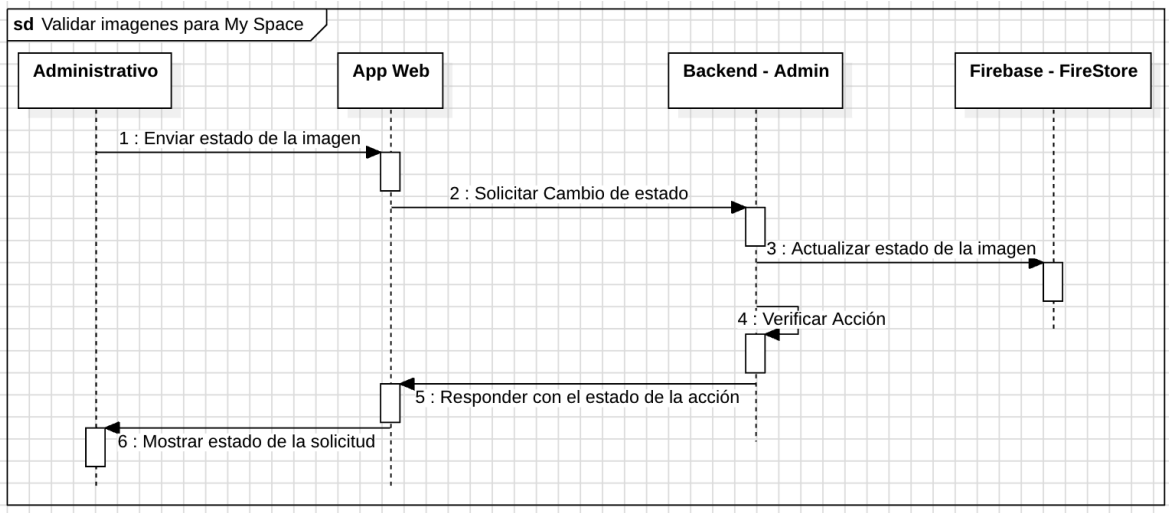


Imagen 33 Imagen diagrama de secuencia – Validar img My Space.

Visualizar imágenes para My Space

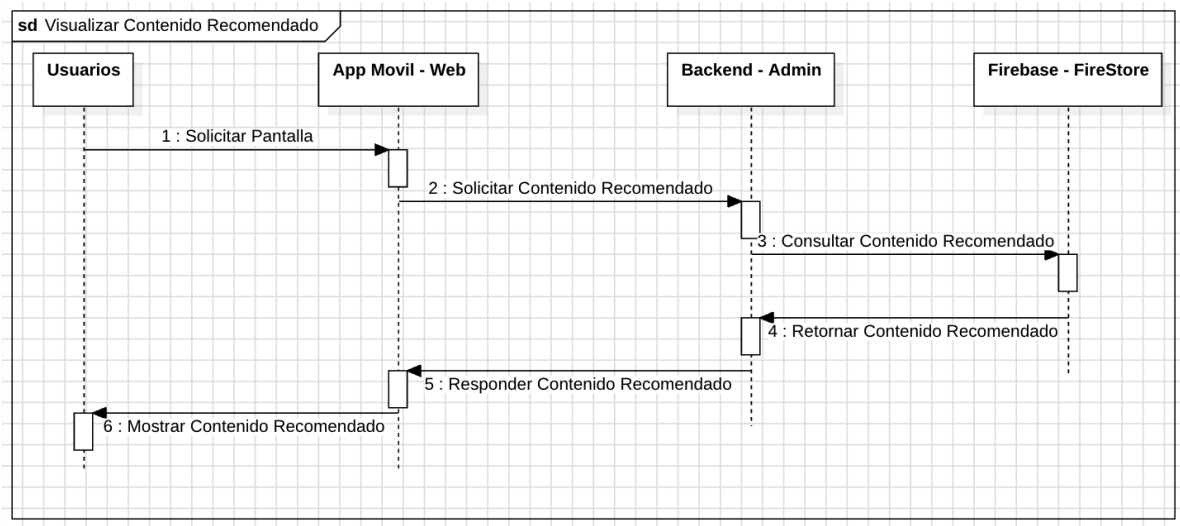


Imagen 34 Imagen diagrama de secuencia – Visualizar contenido recomendación.

Crear imagen a través de IA..

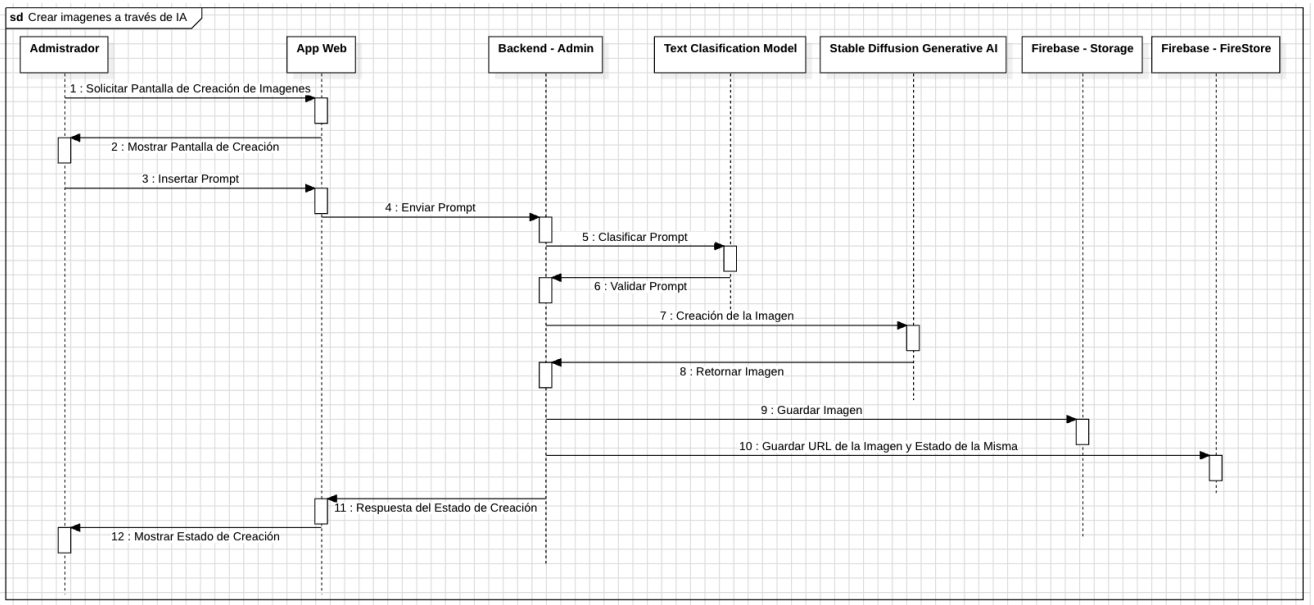


Imagen 35 Imagen diagrama de secuencia – Imagen con IA

Crear imagen a través de IA.

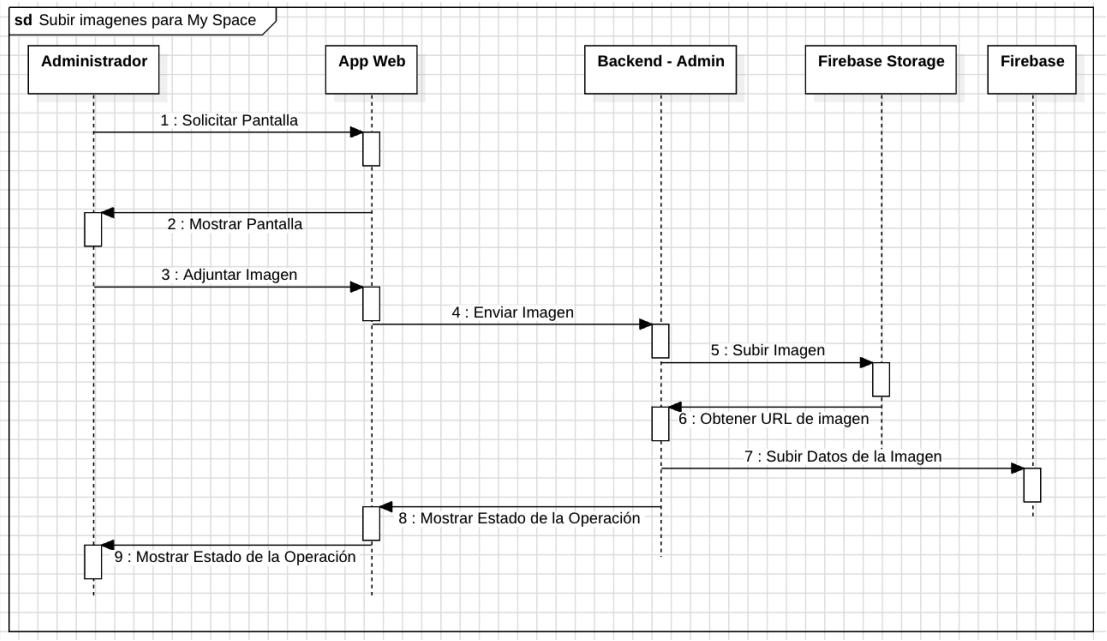


Imagen 36 Imagen diagrama de secuencia – Subir imágenes para My Space.

Crear Líneas de Atención

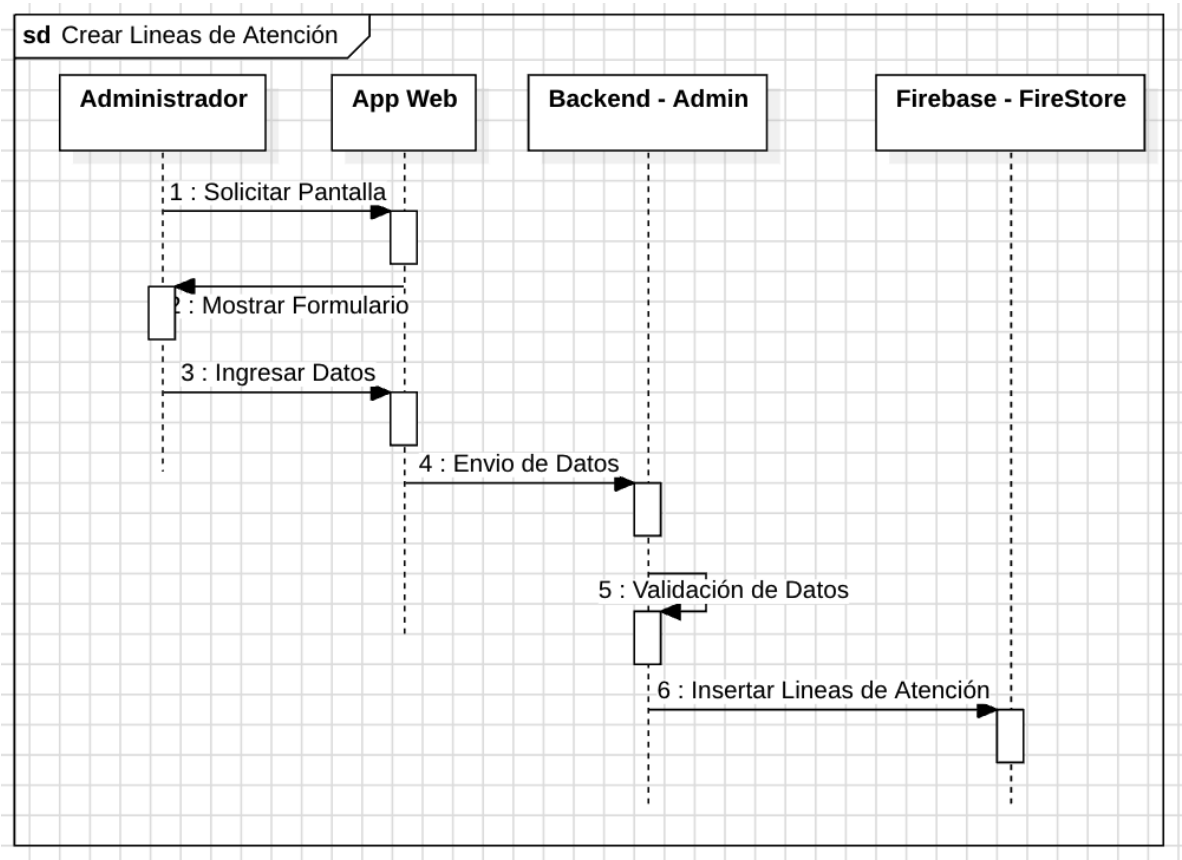


Imagen 37 Imagen diagrama de secuencia – Crear Líneas de atención.

Crear Registro Diario

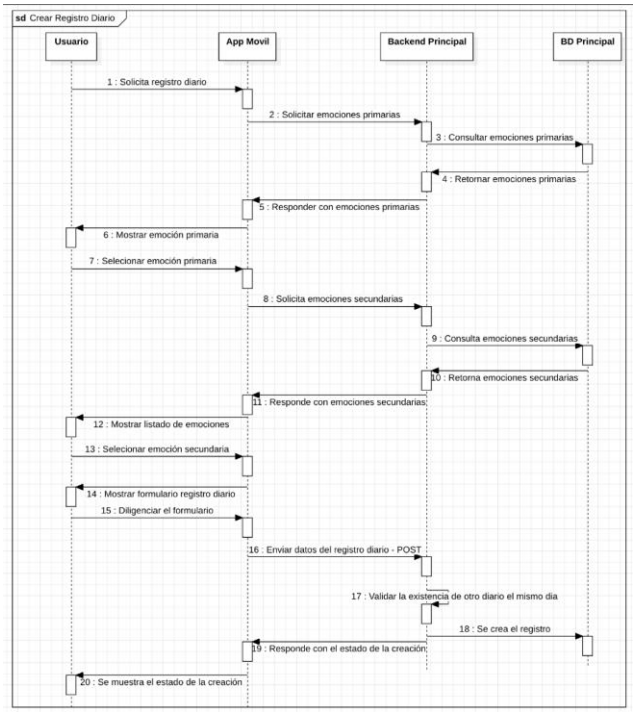


Imagen 38 Imagen diagrama de secuencia – Crear registro diario.

Crear Registro Transcendental

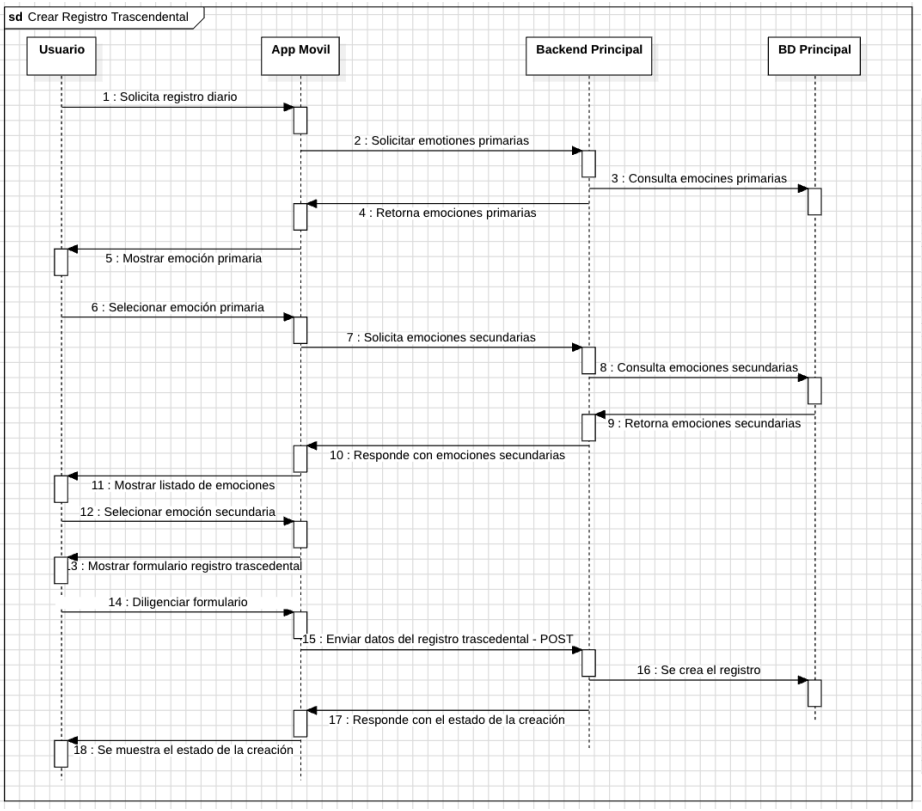


Imagen 39 Imagen diagrama de secuencia – Crear registro transcendental.

Crear Registro Transcendental

Ilustración 40: Imagen diagrama de secuencia – Crear registro transcendental.

Diagrama de Arquitectura de Alto Nivel

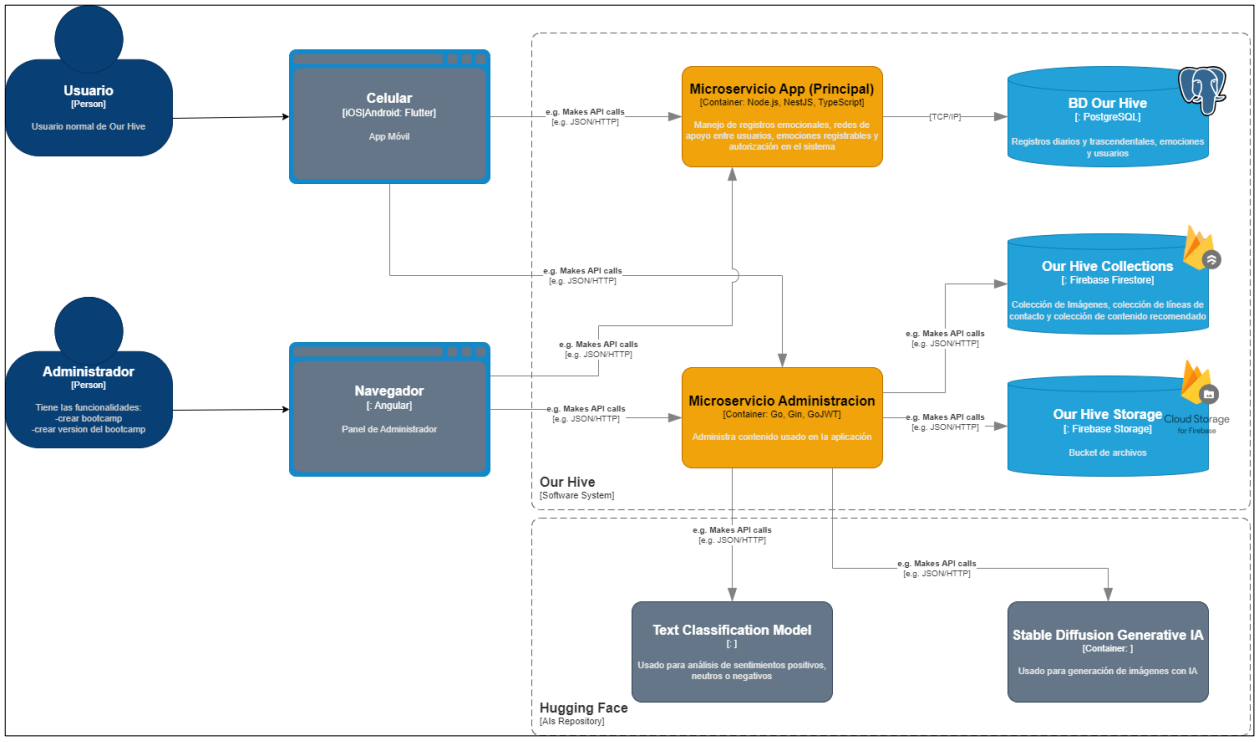


Imagen 40 Imagen diagrama de arquitectura de Alto Nivel.

Diagrama de Componentes:

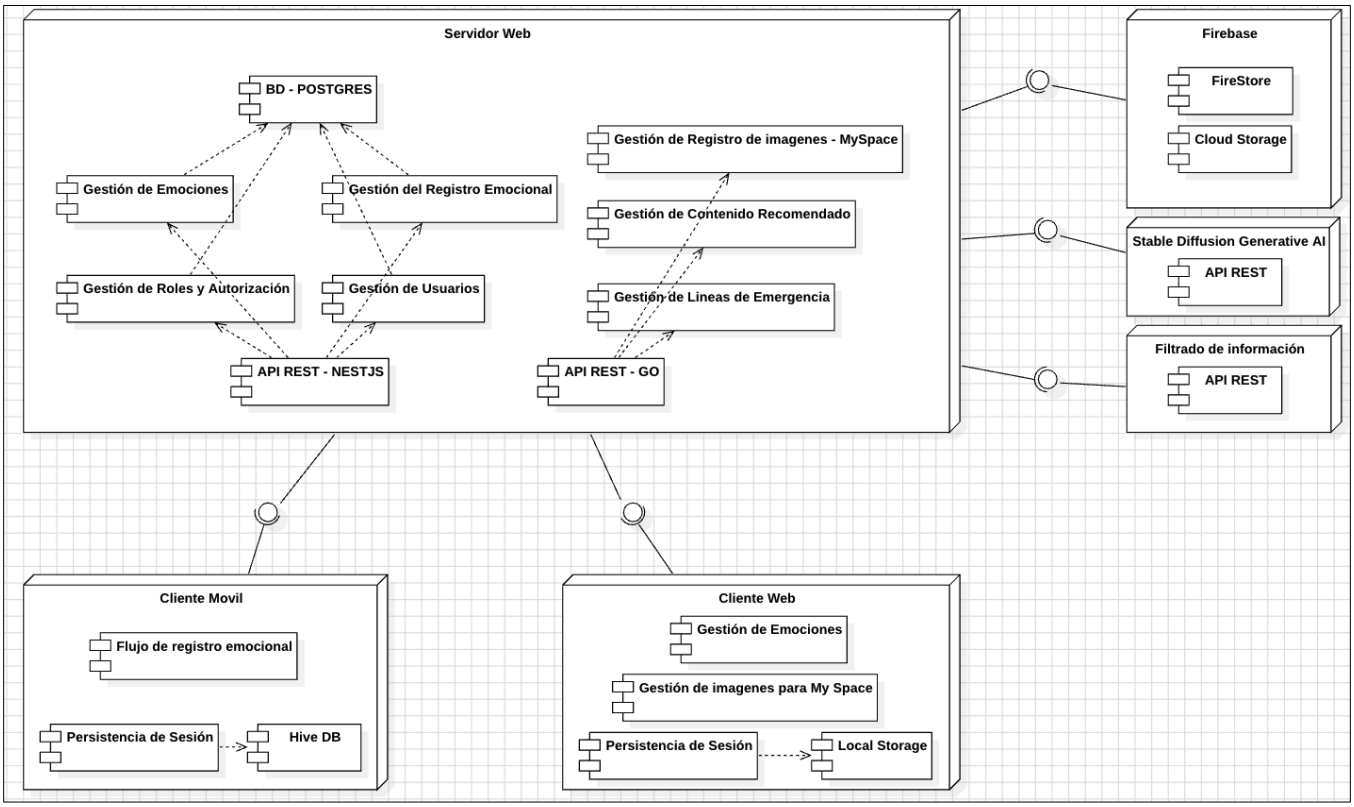


Imagen 41 Imagen diagrama de Componentes.

Requerimientos no funcionales (Lista detallada)

- Alta disponibilidad de los servidores para que la aplicación pueda ser usada todo el tiempo o la gran mayoría del tiempo
- La interacción entre el usuario y la aplicación deben ser lo más rápida posible.
- Los datos de los usuarios deben estar bien protegidos, para evitar la posible fuga de información personal.
- La autenticación del usuario debe ser segura.
- La aplicación debe tener una interfaz entendible y responsive
- La aplicación debe responder de manera eficiente incluso en dispositivos móviles con recursos limitados.
- La aplicación debe establecer la confianza con el usuario a través de políticas claras de privacidad.

Introducción guía de instalación.

Este documento proporciona una guía detallada para la instalación de los dos microservicios/APIs que son utilizadas en Our Hive. Se describen los requisitos

previos, los pasos de instalación y las configuraciones necesarias para garantizar un despliegue exitoso. Este manual está dirigido a administradores de sistemas y desarrolladores responsables de la configuración y mantenimientos de las APIs

Prerrequisitos de instalación del sistema en el servidor

- **Sistema Operativo:**
  - Windows 8.1 o superior.
  - Windows Server 2012 R2 o superior.
  - Distribución Linux con kernel 3.10 o superior.
- **Base de datos:** PostgreSQL 13 (opcionalmente disponible mediante Docker Compose para la API de la aplicación móvil).
- **Servidor de Aplicaciones:**
  - Node.js 12.x o superior (Se recomienda la última versión LTS)
  - Go 1.22.0
- **Docker:** Docker y Docker Compose instalados (necesarios si se utiliza la configuración de base de datos PostgreSQL mediante Docker Compose
- **Conexión a Firebase:** Se requiere acceso a Internet y permisos de salida en el firewall para conectarse a los servicios de Firebase.

Frameworks y estándares

- **Microservicio para la Aplicación Móvil (NestJS):**
  - Framework: NestJS ^10.0.0
  - Lenguaje: TypeScript ^5.1.3
  - Entorno de Ejecución: Node.js 12.x o superior
  - Base de Datos: PostgreSQL 13
  - ORM: TypeORM ^0.3.19
  - Autenticación: JWT con @nestjs/jwt y passport-jwt
  - Documentación API: Swagger ^7.2.0
- **Microservicio para el Panel de Administración (Go):**
  - Lenguaje: Go 1.22.0
  - Framework: Gin v1.9.1
  - Conexión a Firebase: Firebase SDK para Go v3.13.0
  - Conexión a Google Cloud: Google Cloud SDK para Go v0.112.1
  - Formato de Datos: JSON con github.com/goccy/go-json v0.10.2
  - Autenticación JWT: github.com/golang-jwt/jwt/v5 v5.2.1
  - Documentación API: Swagger con github.com/swaggo/gin-swagger v1.6.0 y github.com/swaggo/swag v1.16.3

Descripción del Modelo de Persistencia

- **API Principal (NestJS):**

- **Persistencia de Datos con PostgreSQL:** Utiliza PostgreSQL como base de datos relacional, gestionada a través de TypeORM. TypeORM mapea las entidades del modelo de datos a tablas en la base de datos, lo que asegura la consistencia y la integridad referencial de los datos. La estructura de la base de datos está diseñada para manejar relaciones complejas entre entidades y garantizar un acceso rápido y fiable a los datos. TypeORM facilita la creación, actualización y consulta de datos mediante un enfoque orientado a objetos, optimizando el desarrollo y el mantenimiento de la API.

- **Panel de Administración (Go)**:
  - **Persistencia de Datos con Firestore:** Utiliza Firestore para almacenar datos estructurados en documentos y colecciones. Esto permite una gestión flexible y escalable de la información, ideal para datos dinámicos.
  - **Persistencia de Archivos con Firebase Storage:** Utiliza Firebase Storage para almacenar archivos como imágenes y documentos. Los archivos se almacenan en la nube y están vinculados con los documentos en Firestore, facilitando el acceso y la gestión de los recursos multimedia.

Requerimientos de interface con otros sistemas

Sistemas Externos:

- Conexión a Firebase para base de datos NoSQL (Firestore) y almacenamiento en la nube (Firebase Storage).
- Hugging Face:
  - API de Generación de Imágenes: Permite la generación de imágenes basadas en prompts proporcionados.
  - API de Clasificación de Texto: Revisa la ética de los prompts utilizados en la generación de imágenes para asegurar que sean apropiado

Scripts de instalación y ejecución

API Principal:

1. Descargar código fuente.
2. Diríjase a la carpeta del proyecto:

```
$ cd Our-Hive-Backend
```

3. Configuración de variables de entorno
   a. Copiar el archivo de ejemplo de configuración: En la raíz del proyecto, localice el archivo **`.env.example`**
   b. Cree un nuevo archivo en la raíz del proyecto con el nombre **`.env `**

c. Abra el archivo **`.env.example`** y copie su contenido
d. Pegue el contenido copiado en el archivo **`.env `** que acaba de crear
e. Edite el archivo **`.env `** para completar las variables de entorno con los valores adecuados. Asegúrese de proporcionar los valores adecuados.

> **Nota**: En caso de utilizar Docker, los valores dados Al archivo **`.env`** serán utilizados en el script **`docker-compose.yml`** para la ejecución de la base de datos de PostgreSQL

- **PORT:** Especifica el puerto en el que el servidor de la aplicación escuchará las solicitudes entrantes. Por ejemplo, si se establece PORT=3000, la aplicación se ejecutará en http://localhost:3000.
- **DB_PORT**: El puerto en el que el servicio de base de datos PostgreSQL está escuchando. El puerto por defecto para PostgreSQL es 5432, pero puede ser diferente si se ha configurado de otra manera.
- **DB_HOST:** La dirección del servidor de base de datos PostgreSQL. Esto puede ser una dirección IP, un nombre de host, o localhost si la base de datos está en el mismo servidor que la aplicación.
- **DB_USERNAME:** El nombre de usuario que la aplicación utilizará para conectarse a la base de datos PostgreSQL.
- **DB_PASSWORD:** La contraseña asociada con el DB_USERNAME para autenticar la conexión a la base de datos PostgreSQL.
- **DB_DATABASE:** El nombre de la base de datos específica a la que la aplicación se conectará dentro del servidor PostgreSQL.
- **JWT_SECRET:** Una cadena secreta utilizada para firmar y verificar los tokens JWT (JSON Web Tokens). El secreto debe ser seguro y no debe ser compartido. Se utiliza para asegurar que los tokens no puedan ser alterados sin ser detectados.

4. Estando en la raíz del proyecto, instale las dependencias de este

```
$ npm install
```

5. Ejecute el contenedor de PostgreSQL (Opcional)

```
$ docker-compose up -d
```

6. Ejecute el proyecto según sus necesidades

```
# Ejecutar en modo desarrollo

$ npm run start

# Ejecutar en modo watch

$ npm run start:dev

#Ejecutar en modo producción

$ npm run start:prod
```

API Panel de Administración:

1. Descargar código fuente.
2. Diríjase a la carpeta del proyecto

```
$ cd Our-Hive-Admin-Panel-API
```

3. Configuración de variables de entorno
   a. Copiar el archivo de ejemplo de configuración: En la raíz del proyecto, localice el archivo **`.env.example`**
   b. Cree un nuevo archivo en la raíz del proyecto con el nombre **`.env `**
   c. Abra el archivo **`.env.example`** y copie su contenido
   d. Pegue el contenido copiado en el archivo **`.env `** que acaba de crear
   e. Edite el archivo **`.env `** para completar las variables de entorno con los valores adecuados. Asegúrese de proporcionar los valores adecuados.

Nota exploratoria de variables de entorno:

- **HUGGING_FACE_TOKEN**: Token de acceso para la API de Hugging Face.
- **FB_STORAGE_BUCKET**: Nombre del bucket de almacenamiento en Firebase.
- **SECRET**: Clave secreta utilizada para la autenticación y seguridad en los servicios.
- **FB_ADMIN_SDK_PATH**: Ruta al archivo de configuración del SDK de administración de Firebase (por ejemplo, /home/nico/Documents/our-hive/Our-Hive-Admin-Panel-API/internal/infrastructure/adminsdk.json).
- **FB_ADMIN_SDK_PATH**: Ruta al archivo de configuración del SDK de administración de Firebase

4. Estando en la raíz del proyecto, instale las dependencias de este

```
$ go mod tidy
```

5. Ejecute el proyecto según sus necesidades

```
# Ejecutar en modo desarrollo
$ go run cmd/web/main.go
# Compilar y ejecutar para producción
$ go build cmd/web/main.go && ./main
```

Prerrequisitos:

Para probar las aplicaciones Frontend, es necesario tener en el computador instalado:

- NodeJS: v20.17.0
- Flutter: v3.24.4
- Dart: v3.5.4
- Android SDK: v35.0.0
- Xcode: v16.1

Para la instalación de la aplicación Web.

Verificar en el path, las urls para conectarse a cada Backend:

- src/environments/environment.development.ts

Y en la terminal utilizar el siguiente los siguientes comandos:

Para instalar las dependencias del proyecto:

- npm install

Para iniciar la aplicación en modo desarrollo:

- npm run start:dev

Para la instalación de la aplicación Móvil.

Realizar una copia del contenido del archivo

- .env.example en el archivo .env

Verificar la ruta y las variables de entorno:

API_URL=

GO_API_URL=

Posteriormente a esto, se puede utilizar el siguiente comando:

Para instalar las dependencias del proyecto:

- flutter pub get

Para iniciar la aplicación en modo desarrollo:

- flutter run

Resultados:

- El 23 de agosto de 2023, se llevó a cabo una sesión clave con la Psicóloga de la facultad, Leidy Viviana Montañez Pinzón, para especificar los requerimientos funcionales y no funcionales, elementos esenciales para el desarrollo del proyecto. Durante esta reunión, se abordaron a fondo las problemáticas relacionadas con el manejo y la gestión emocional dentro de la facultad. Esta conversación no solo permitió identificar y definir los módulos críticos para la aplicación, sino también establecer un enfoque psicológico adecuado hacia la gestión emocional. Además, se exploraron posibles apoyos desde el bienestar institucional, así como autores y referencias valiosas para la investigación y posterior implementación del proyecto.
  Adjunto relatoría de la reunión realizada con la Psicóloga.

- Se ha desarrollado el módulo de registro emocional, la herramienta está diseñada para permitir a los usuarios ingresar y diferenciar entre emociones primarias y secundarias de manera precisa y sencilla, ayudándoles a gestionar sus emociones de forma más efectiva, todo esto a través de una interfaz intuitiva y agradable para los usuarios. A través del reconocimiento de estas emociones, los usuarios desarrollarán habilidades para comprender cómo influyen en su día a día y aprenderán a identificar diversas emociones secundarias, que suelen ser menos conocidas que las emociones primarias.

- Con el objetivo de mejorar la situación emocional de los usuarios de Our Hive, se implementó un módulo de gestión de líneas de atención. Este módulo permite a los usuarios acceder de manera ágil y sencilla a diversas líneas de atención psicológica disponibles en Colombia, facilitando un acceso inmediato a estos servicios. La implementación de este módulo busca asegurar que los usuarios puedan conectarse rápidamente con profesionales de la salud mental, superando barreras como el desconocimiento o la falta

de información que, en muchos casos, impiden a las personas acceder a estos recursos esenciales. Adicionalmente, la implementación del módulo “Red de Apoyo” permite a los usuarios compartir su historial emocional con otros miembros de la comunidad, creando así una red de apoyo que fomenta el acompañamiento mutuo y el intercambio de experiencias. Este módulo facilita a los usuarios sentirse respaldados y conectados, promoviendo un entorno de comprensión y apoyo emocional, fortaleciendo el sentido de comunidad y ofreciendo a cada usuario la oportunidad de aprender de las vivencias de los demás, lo que puede contribuir significativamente a su bienestar emocional.

Conclusiones

- Lograr identificar las situaciones que nos rodean a la hora de sentir una emoción, nos puede ayudar a sobre llevar los sucesos que del día a día de los seres humanos.

- Tener un fácil acceso a la información emocional, tal como descripciones, listas de emociones y relación entre emociones principales y secundarias, nos ayuda a reconocer, identificar y comprender las emociones de mejor manera.





- El impacto tecnológico que ha vivido la sociedad hace necesario que personas con el conocimiento técnico y de software, cree herramientas que impacten significativamente en la humanidad en diversos ambientes, como en este caso impactamos en el desarrollo de la gestión emocional.

- El desarrollo de software multiplataforma nos ayuda a brindar soluciones para múltiples usuarios y romper las barreras sociales, a su vez logramos ampliar significativamente la red de apoyo emocional, permitiendo visualizar el historial emocional de otros usuarios.

- La integración de servicios, como diferentes bases de datos, almacenamiento de archivos, Inteligencia Artificial, por medio de las API REST, nos permite manejar estándares y desarrollar productos de software más completos, escalables y mantenibles.

- Trabajar de la mano con personas con conocimientos en el ámbito de la psicología, nos permite realizar un levamiento de requerimientos adecuado

para presentar una solución tecnológica con los criterios de aceptación validados.

10. Referencias

[1] App Mood Tracker. Moodpress - mood diary tracker - apps on google play. Android Apps on Google Play [página web]. [Consultado el 12, octubre, 2023]. Disponible en Internet: https://play.google.com/store/apps/details?id=com.selfcare.diary.mood.tracker.moodpress&hl=en_US.

[2] ARIAS BATTLE, Yeraldin y ROA RODRÍGUEZ, Guillermo. Modelo en gerencia de proyectos para pymes de consultoría informática en Bogotá, basado en marcos ágiles de trabajo. En: Tecnura [en línea]. 25, septiembre, 2022. vol. 26, no. 74 [consultado el 13, noviembre, 2023], p. 49-69. Disponible en Internet: https://doi.org/10.14483/22487638.17094. ISSN 2248-7638.

[3] ARIZA ARIZA, Maria Daniela; CARDONA DÍAZ, Nayibe y TARAZONA BERMÚDEZ, Giovanny Mauricio. Application proposal for Colombian agriculture. En: Revista vínculos [en línea]. 30, junio, 2019. vol. 16, no. 1 [consultado el 13, noviembre, 2023], p. 17-28. Disponible en Internet: https://doi.org/10.14483/2322939x.15089. ISSN 2322-939X.

[4] BEEKEPER STUDIO. Licencia GPL de Beekeper publicada en GitHub. GitHub: Let's build from here · GitHub [página web]. [Consultado el 1, noviembre, 2023]. Disponible en Internet: <https://github.com/beekeeper-studio/beekeeper-studio/blob/master/LICENSE.md>.

[5] CAMARGO, Diego Fernando; TOVAR, Edwar David y NIÑO, Julian David. Proceso de cifrado con algoritmo idea y ofuscamiento de código en servicios web. Sistema de revistas científicas - Universidad Distrital Francisco José de Caldas [página web]. (29, junio, 2022). [Consultado el 1, noviembre, 2023]. Disponible en Internet: https://revistas.udistrital.edu.co/index.php/vinculos/article/view/16855/18579.

[6] GITHUB. Pricing · Plans for every developer. GitHub [página web]. [Consultado el 16, abril, 2024]. Disponible en Internet: <https://github.com/pricing>.

[7] DAYLIO. Daylio - journal, diary and mood tracker. Daylio [página web]. [Consultado el 10, noviembre, 2023]. Disponible en Internet: https://daylio.net/.

[8] UNICEF. EL IMPACTO del COVID-19 en la salud mental de adolescentes y jóvenes. UNICEF [página web]. [Consultado el 14, noviembre, 2023]. Disponible en Internet: https://www.unicef.org/lac/el-impacto-del-covid-19-en-la-salud-mental-de-adolescentes-y-jóvenes.

[9] GOOGLE. Dart overview. Dart programming language | Dart [página web]. [Consultado el 12, abril, 2024]. Disponible en Internet: https://dart.dev/overview.

[10] FLUTTER/LICENSE AT master · flutter/flutter [Equipo de Google]. GitHub [página web]. [Consultado el 12, abril, 2024]. Disponible en Internet: https://github.com/flutter/flutter/blob/master/LICENSE.

[11] FRAGOSO LUZURIAGA, Rocio. Emotional Intelligence in university classrooms: teaching practices that promote its development. En: Zona Próxima [en línea]. 24, enero, 2022. no. 36 [consultado el 14, noviembre, 2023], p. 49-75. Disponible en Internet: https://doi.org/10.14482/zp.36.152.4. ISSN 2145-9444.

[12] GITHUB. "Políticas de uso aceptable de GitHub · Documentación de GitHub." GitHub Docs [página web]. [Consultado el 7, noviembre, 2023]. Disponible en Internet: https://docs.github.com/es/site-policy/acceptable-use-policies/github-acceptable-use-policies#1-compliance-with-laws-and-regulations.

[13] GIT. Licencia GPL de Git publicada en GitHub. GitHub: Let's build from here · GitHub [página web]. [Consultado el 1, noviembre, 2023]. Disponible en Internet: <https://github.com/git/git/blob/master/COPYING>.

[14] GOLEMAN, Daniel; BENNIS, Warren y CHERNISS, Cary. Emotionally intelligent workplace: how to select for, measure, and improve emotional intelligence in individuals, groups, and organizations. [s.l.]: Wiley & Sons, Limited, John, 2011. 384 p. ISBN 9781118308790.

[15] GONZÁLEZ DURÁN, Daniel Ricardo. Des(cubro) - diario emocional, miradas internas. Repositorio Institucional Séneca [página web]. (2, mayo, 2016). [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://repositorio.uniandes.edu.co/entities/publication/bef0b7fa-e625-4c49-936c-52e77cdfaa99.

[16] HIDALGO SUAREZ, Carlos Giovanny; BUCHELI-GUERRERO, Víctor Andrés y ORDÓÑEZ-ERASO, Hugo Armando. "Artificial intelligence and computer-supported collaborative learning in programming: a systematic mapping study." En: Tecnura [en línea]. 1, enero, 2023. vol. 27, no. 75 [consultado el 14, noviembre, 2023], p. 175-206. Disponible en Internet: https://doi.org/10.14483/22487638.19637. ISSN 2248-7638.

[17] HOW WE FEEL ORGANIZATION. How we feel app. How We Feel [página web]. [Consultado el 7, noviembre, 2023]. Disponible en Internet: https://howwefeel.org/.

[18] JORGE EDUARDO, Hernandez Rodriguez y KAREN GISELLE, Daza Alvarado. "Desarrollo e Implementación de un prototipo para la estimación hídrica, RIWA -

hdl:11349/24325." Repositorio Institucional Universidad Distrital - RIUD [página web]. (13, septiembre, 2019). [Consultado el 3, noviembre, 2023]. Disponible en Internet: https://repository.udistrital.edu.co/handle/11349/24325.

[19] JWT. Jwt.io. JSON Web Tokens - jwt.io [página web]. [Consultado el 10, noviembre, 2023]. Disponible en Internet: https://jwt.io/.

[20] LADINO MORENO, Édgar Orlando; GARCÍA-UBAQUE, César Augusto y GARCÍA VACA, María Camila. "Critical flow in open channels: numerical solution using the newton-raphson method for android 4.0 application." En: Tecnura [en línea]. 1, enero, 2020. vol. 24, no. 63 [consultado el 14, noviembre, 2023], p. 99-114. Disponible en Internet: https://doi.org/10.14483/22487638.16196. ISSN 2248-7638.

[21] LADINO MORENO, Edgar Orlando; GARCÍA UBAQUE, César Augusto y GARCÍA VACA, María Camila. "Darcy-Weisbach resistance coefficient determination using newton-raphson approach for android 4.0." En: Tecnura [en línea]. 1, abril, 2019. vol. 23, no. 60 [consultado el 14, noviembre, 2023], p. 52-58. Disponible en Internet: https://doi.org/10.14483/22487638.14929. ISSN 2248-7638.

[22] LADINO MORENO, Edgar Orlando; GARCÍA UBAQUE, César Augusto y PINEDA JAIMES, Jorge Arturo. "Desarrollo de una aplicación móvil para el aprendizaje interactivo en problemas de ingeniería civil: aplicación a la hidráulica de canales abiertos." En: Tecnura [en línea]. 1, enero, 2021. vol. 25, no. 67 [consultado el 14, noviembre, 2023], p. 53-70. Disponible en Internet: https://doi.org/10.14483/22487638.17820. ISSN 2248-7638.

[23] MACIAS, José, et al. "Diseño e implementación de un Gateway IoT multiprotocolo." Sistema de revistas científicas - Universidad Distrital Francisco José de Caldas [página web]. (30, junio, 2021). [Consultado el 1, noviembre, 2023]. Disponible en Internet: https://revistas.udistrital.edu.co/index.php/vinculos/article/view/20188/19184.

[24] MICROSOFT. Licencia Apache 2.0 de TypeScript publicada en Github. GitHub: Let's build from here · GitHub [página web]. [Consultado el 2, noviembre, 2023]. Disponible en Internet: https://github.com/microsoft/TypeScript/blob/main/LICENSE.txt.

[25] MICROSOFT. "Licencia MIT de Visual Studio Code publicada en GitHub." GitHub: Let's build from here · GitHub [página web]. [Consultado el 7, noviembre, 2023]. Disponible en Internet: https://github.com/microsoft/vscode/blob/main/LICENSE.txt.

[26] MICROSOFT. GitHub - microsoft/typescript: typescript is a superset of javascript that compiles to clean javascript output. GitHub [página web]. [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://github.com/microsoft/TypeScript#readme.

[27] MINISTERIO DE SALUD DE COLOMBIA. Reconocer la dignidad humana: clave en abordaje de la salud mental. Ministerio de Salud [página web]. (26, julio, 2022). [Consultado el 7, abril, 2023]. Disponible en Internet: https://www.minsalud.gov.co/Paginas/Reconocer-la-dignidad-humana-clave-en-abordaje-de-la-salud-mental.aspx.

[28] NESTJS. "Licencia MIT de Nest publicada en GitHub." GitHub: Let's build from here · GitHub [página web]. [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://github.com/nestjs/nest/blob/master/LICENSE.

[29] NESTJS. NestJS - A progressive Node.js framework. NestJS - A progressive Node.js framework [página web]. [Consultado el 15, noviembre, 2023]. Disponible en Internet: https://nestjs.com/.

[30] PAVA, Roberto; PEREZ CASTILLO, José Nelson y NIÑO VASQUEZ, Luis Fernando. "Perspectiva para el uso del modelo P6 de atención en salud bajo un escenario soportado en IoT y blockchain." En: Tecnura [en línea]. 1, enero, 2021. vol. 25, no. 67 [consultado el 14, noviembre, 2023], p. 112-130. Disponible en Internet: https://doi.org/10.14483/22487638.16159. ISSN 2248-7638.

[31] POSTGRESQL. PostgreSQL: license. PostgreSQL: The world's most advanced open source database [página web]. [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://www.postgresql.org/about/licence/.

[32] RAMÍREZ, Helio Henry; CARO, Nancy Nubia y CONTRERAS, Juan David. "Aplicación móvil para medir la frecuencia cardíaca: fiabilidad, barreras y oportunidades." Sistema de revistas científicas - Universidad Distrital Francisco José de Caldas [página web]. (28, diciembre, 2021). [Consultado el 2, noviembre, 2023]. Disponible en Internet: https://revistas.udistrital.edu.co/index.php/vinculos/article/view/16388/18577.

[33] REDHAT. REST API. Red Hat [página web]. (7, noviembre, 2023). [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://www.redhat.com/es/topics/api/what-is-a-rest-api.

[34] SALUD MENTAL en Colombia: retos, cifras y más datos de interés con MinSalud [Anónimo]. Caracol Radio [página web]. [Consultado el 14, noviembre, 2023]. Disponible en Internet:

https://caracol.com.co/programa/2022/07/26/6am_hoy_por_hoy/1658846303_927005.html.

[35] SAMAYOA MIRANDA, María de Lourdes y MÁRQUEZ MUNGUÍA, Yvette. "¿Qué es la inteligencia emocional?" En: SAVIA Revista de investigación e intervención social [en línea]. 25, noviembre, 2013. p. 10-15. [Consultado el 10, octubre, 2023]. Disponible en Internet: https://doi.org/10.36794/savia.vi11.46. ISSN 2448-6604.

[36] TERAPIFY. *Diario de las emociones: ¿qué es y para qué sirve? - Terapify.* Terapify [página web]. (24, octubre, 2023). [Consultado el 2, noviembre, 2023]. Disponible en Internet: https://www.terapify.com/blog/diario-de-las-emociones/.

[37] VIDAL SAPIÑA, MARÍA JOSÉ. "Diseño y maquetación de un producto editorial basado en la herramienta mood tracker de ayuda a la mejora del desarrollo personal." *RiuNet repositorio UPV* [página web]. (26, septiembre, 2022). [Consultado el 9, noviembre, 2023]. Disponible en Internet: https://riunet.upv.es/handle/10251/186552.

[38] WINTERS, Jason; CLIFT, Robert J. W. y DUTTON, Donald G. "An exploratory study of emotional intelligence and domestic abuse." En: *Journal of Family Violence* [en línea]. Octubre, 2004. vol. 19, no. 5 [consultado el 23, octubre, 2023], p. 255-267. Disponible en Internet: https://doi.org/10.1023/b:jofv.0000042076.21723.f3. ISSN 0885-7482.

[39] 블루시그넘(BLUESIGNUM CORP.). *DailyBean: simplest journal - apps on google play.* Android Apps on Google Play [página web]. [Consultado el 11, noviembre, 2023]. Disponible en Internet: https://play.google.com/store/apps/details?id=com.bluesignum.bluediary&hl=en_US&pli=1.

[40] ARÉVALO RODRÍGUEZ, Anderson Smith; HURTADO GÓMEZ, Diana Marcela y GALINDO SIERRA, Gilber Jhon. Algoritmo internacional de cifrado de datos (IDEA) que utiliza la variante de cifrado SHA-256. Sistema de revistas científicas - Universidad Distrital Francisco José de Caldas [página web]. (29, diciembre, 2022). [Consultado el 6, noviembre, 2023]. Disponible en Internet: <https://revistas.udistrital.edu.co/index.php/vinculos/article/view/20826/19136>.

[41] GOOGLE. Flutter - Build apps for any screen. Flutter - Build apps for any screen [página web]. [Consultado el 16, abril, 2024]. Disponible en Internet: <https://flutter.dev/>.

[42] GOOGLE. GitHub - golang/go: The Go programming language. GitHub [página web]. [Consultado el 13, agosto, 2024]. Disponible en Internet: <https://github.com/golang/go>.