

MEMORIA ESCRITA DEL PROYECTO

CFGS Desarrollo de Aplicaciones Multiplataforma

FirstYear



Autor: Raquel Melero Somoza

Tutor: Gotzon Valcarcel

Fecha de entrega: 05/12/2024

Convocatoria: 1s2425

Documentos del proyecto:

<https://drive.google.com/drive/folders/1d4nLwrZUM8zEtDxs1duXITG65Wm4VMnl?usp=shari>



Índice

1. Introducción	2
1.1. Motivación	2
1.2. Abstract	3
1.3. Objetivos propuestos	4
2. Estado del Arte	5
2.1. Análisis de las aplicaciones más relevantes	5
2.2. Análisis de tendencias y vacíos en el mercado	8
3. Metodología utilizada	10
3.1. Metodología kanban	10
3.2. Ciclo de vida iterativo incremental	10
4. Tecnologías y herramientas utilizadas en el proyecto	12
5. Planificación, diagnóstico y contexto laboral	15
5.1. Planificación	15
5.2. Diagnóstico	17
6. Análisis del proyecto	19
6.1. Requisitos funcionales y no funcionales	19
6.2. Diagrama entidad-relación	21
6.3. Casos de uso	22
6.4. Diagrama de clases	27
7. Diseño del proyecto	28
7.1. Diseño previo a la codificación	28
7.2. Codificación de la interfaz	30
7.3. Base de datos	32
7.4. Codificación de la vista y eliminación de datos	33
7.5. Mejoras y puesta a punto	34
7.6. Ficheros y carpetas	34
8. Despliegue y pruebas	35
8.1. Herramientas utilizadas	35
8.2. Pruebas de caja negra	35
9. Conclusiones	40
10. Vías futuras	41
11. Bibliografía/Webgrafía	43
12. Anexos	45
12.1. Manual de instalación	45
12.2. Manual de usuario	47
12.3. Imágenes	51

1. Introducción

En este apartado vamos a mostrar una visión general sobre el proyecto del que trata este documento. En él podemos ver la motivación que nos ha llevado a crearlo, un breve abstracto y los objetivos de la aplicación.

En este documento hemos plasmado el análisis, diseño y desarrollo de una aplicación móvil sencilla, intuitiva y con una interfaz visual amigable.

Para poder poner en contexto este documento, este se basa en la memoria o documentación de la creación de **FirstYear**, una aplicación móvil desarrollada para el sistema operativo Android. Esta aplicación se ha diseñado para facilitar a las familias una herramienta útil los primeros años de vida de sus bebés. Esto se debe a que les permite registrar los hábitos y rutinas de los más pequeños, además de ofrecer una visión sobre esos datos. Proporcionando una herramienta de registro y consulta de datos que facilita su día a día.

La aplicación creada se usa de forma fácil e intuitiva, en la que predominan los iconos con dibujos explícitos y sencillos, selectores de fecha y hora automáticos o la visión de los datos registrados de forma intuitiva.

Debido al tiempo que se da para poder realizar el desarrollo y documentación del proyecto, hay funcionalidades que se llevarán a cabo en un futuro, actualizando la aplicación y mejorándola para que sea más funcional, atractiva y útil para las familias.

1.1. Motivación

La motivación para crear **FirstYear** surge de la necesidad que tienen las familias de organizar y registrar el cuidado diario de sus bebés de forma fácil y digital.

El primer año de vida de un bebé, además de ser muy intenso para su familia, que se enfrenta a muchos cambios, está lleno de eventos importantes. A primera vista no todos parecen tan importantes, pero lo son.

Debido al nivel de vida que llevamos actualmente y a los cambios que suponen la llegada de un bebé a una familia, en ciertas ocasiones a las familias nos cuesta mantener un registro preciso de todos estos eventos o ni siquiera sabemos que tenemos que registrarlos para cuando vayamos al pediatra y nos pregunten por estos datos.

Profesionalmente, el desarrollo de este proyecto me brinda la oportunidad, como desarrolladora, para aprender conocimientos nuevos, practicar los conocimientos adquiridos

durante el curso y seguir mejorando mis habilidades técnicas, además de enfrentarme a un proyecto desde cero y en el que tengo un interés personal.

Por otro lado, como madre, es una herramienta que me vendrá muy bien si viene otro bebé a nuestra familia, para poder registrar todo lo que no registré con la primera y tendría que haberlo hecho. Además, me ofrece una oportunidad de ayudar a otras familias a tener un primer año de vida con algo más de control en los hábitos y rutinas.

La idea para realizar este proyecto, y no otro, surge de la experiencia de maternidad que tengo. En el primer año de vida de nuestra hija no sabíamos que teníamos que llevar un control de los pañales manchados, el tiempo de sueño, cuánto comía, cuántas veces o cuándo cumplía ciertos hitos de desarrollo hasta la primera visita de control con la pediatra. Viendo las preguntas que nos hacían, pensamos que sería ideal tener una aplicación donde poder apuntar todos estos datos y que fuese fácil revisar estos datos cuando hiciese falta.

Encontramos una aplicación para registrar las tomas de lactancia materna y nos pareció una buena idea poder registrar el resto de cosas que te suelen preguntar en el médico, además de hacerla más sencilla e intuitiva, utilizando iconos representativos, de gran tamaño para cuando tienes que usar la aplicación mientras das de comer al bebé o lo tienes en brazos...

Esperamos que sea de gran utilidad para las familias y les ayude a afrontar ese primer año de vida de sus pequeños con más orden y tranquilidad.

1.2. Abstract

FirstYear is a mobile app that helps families record and track important routines and moments in their babies' first years of life.

This application helps families keep track of some of the activities that occur in the first years of life, such as: diaper changes, feeding, sleeping or developmental milestones. The main objective of the application is to facilitate the registration and consultation of these events, thus promoting better management of child care during a key stage of growth in which families have many changes.

The application has been made using Android Studio, with the Kotlin language and using Firebase to make the database with Firestore and save the data. The application has been designed to be used on Android operating systems and used on mobile devices.

We have created an application that is easy and intuitive to use, dominated by icons with explicit drawings, automatic date and time selectors or the view of the recorded data in an intuitive way.

Many families will be able to keep track of their babies' routines and manage this data for when they have appointments with the pediatrician or need to take stock of their habits in a simple and intuitive way, thus facilitating the daily lives of families in a complicated moment.

This version of the application is in development, there are functionalities that will be carried out in the future, updating the application and improving it to make it more functional, attractive and useful for families.

1.3. Objetivos propuestos

Para llevar a cabo este proyecto se han planteado varios objetivos, un objetivo general y varios objetivos específicos que profundizan en el diseño y las funcionalidades de la aplicación.

La definición de estos objetivos es importante, ya que con ellos podemos ver de forma general qué vamos a incluir en la aplicación y cuál queremos que sea la idea en torno a la que diseñamos la aplicación.

Objetivo general:

- Facilitar a las familias el registro y el seguimiento del cuidado diario y los hitos importantes en el desarrollo de sus bebés durante los primeros años de vida.

Para poder llevar a cabo el objetivo general, planteamos varios objetivos específicos que nos ayudan a concretar funcionalidades y características que se van a incorporar en la aplicación.

Objetivos específicos:

- Desarrollar una aplicación móvil en Android Studio que permita a las familias registrar las actividades diarias como cambios de pañal, alimentación o hitos del desarrollo, entre otros.
- Diseñar un entorno fácil e intuitivo para facilitar su uso.
- Crear un historial accesible donde las familias puedan modificar, eliminar y consultar el seguimiento de las actividades.
- Ofrecer una plataforma segura donde se almacenen los datos del bebé de manera privada, accesible solo por los usuarios autorizados.

2. Estado del Arte

En este apartado vamos a realizar una revisión sobre las aplicaciones de cuidado de bebés que hemos investigado, analizando sus características, fortalezas y limitaciones.

Se ha producido un aumento considerable en el uso del móvil, ya sea para buscar información, ver contenido que nos interesa o facilitarnos el día a día. Y es por ello, que han aparecido muchísimas aplicaciones relacionadas con la familia, la salud o el cuidado de bebés. Estas aplicaciones son especialmente útiles para llevar un registro de las actividades diarias como la alimentación, los cambios de pañal y el sueño, permitiendo ver patrones de comportamiento que pueden ayudar en el cuidado del bebé.

Sin embargo, la mayoría de aplicaciones que hemos encontrado se especializan en una o pocas áreas de cuidado, dejando oportunidades para el desarrollo de aplicaciones que ofrezcan una visión más completa del desarrollo infantil.

Por otra parte, muchas de las aplicaciones que hemos encontrado en la Play Store son complejas o no son realmente funcionales para las familias.

La selección de las aplicaciones que queremos revisar se ha llevado a cabo entre las aplicaciones con mejores puntuaciones de la Play Store, para que sean del sistema operativo Android, las que fuesen gratuitas, ya que no disponemos de presupuesto para pagar y probar las apps de pago y las que nos han recomendado otras familias. Entre todas las aplicaciones, hemos seleccionado tres de ellas para llevar a cabo un análisis más concreto: LactApp, Erby y Baby Daybook.

2.1. Análisis de las aplicaciones más relevantes

A continuación mostramos un breve análisis de las aplicaciones que hemos analizado, viendo una pequeña descripción de la aplicación, las fortalezas y las limitaciones que hemos identificado.

Para terminar este apartado, mostraremos unas conclusiones generales sobre las aplicaciones investigadas.

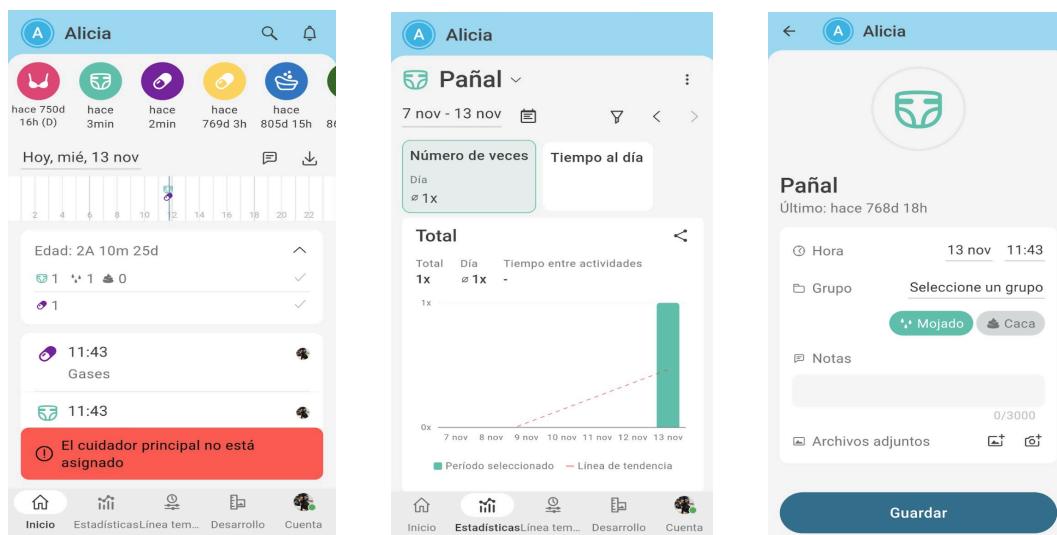
Baby Daybook

Esta aplicación permite registrar información relacionada con el cuidado del bebé. Además, facilita una visión general del desarrollo del bebé mediante gráficos y una línea temporal.

Entre las fortalezas de esta aplicación, destacan las funciones útiles para monitorizar las rutinas de los bebés.

Además de los gráficos que ayudan a visualizar de forma sencilla los registros de las diferentes áreas, la vista en formato línea temporal puede ser muy útil para ver un resumen del día de forma gráfica.

Alguna de las limitaciones que encontramos en esta aplicación es la cantidad de publicidad que tiene, haciendo muy complejo su uso, ya que de repente te saltan anuncios en mitad de una toma o un registro.



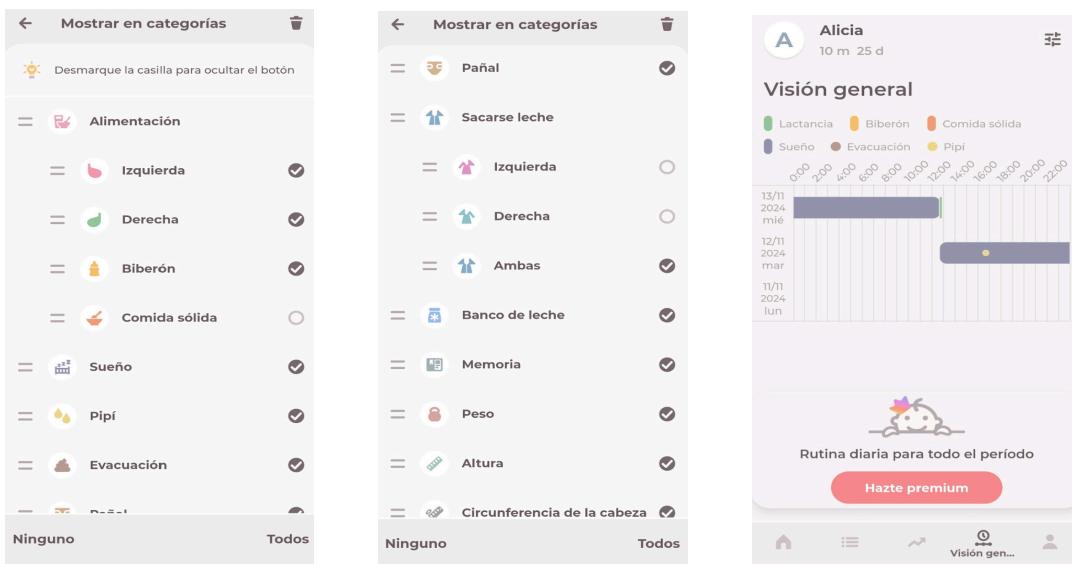
Ejemplo de pantallas de la aplicación Baby Daybook

Lactancia y Baby tracker Erby

Esta aplicación permite que los cuidadores registren la información relacionada con las rutinas del bebé. Además, facilita una visión general del desarrollo del bebé mediante gráficos y estadísticas.

Entre las fortalezas de esta aplicación, destacan las funciones útiles para monitorizar las rutinas de los bebés, además de los gráficos que ayudan a visualizar de forma sencilla los registros de las diferentes áreas.

Algunas de las limitaciones que encontramos en esta aplicación son la interfaz poco intuitiva y sobrecargada que tiene, además de incluir anuncios y tener una estética poco amigable, y la cantidad de opciones que tiene para registrar datos y rutinas. A primera vista puede parecer una fortaleza, pero a la hora de la verdad, abruma tener tantas opciones y tanto contenido.



Ejemplo de pantallas de la aplicación Erby

LactApp

Esta aplicación es una de las más avanzadas en el campo de la lactancia materna. Su objetivo principal es ofrecer asesoría y recursos educativos sobre lactancia a través de una plataforma interactiva.

Entre sus funcionalidades, permite registrar aspectos básicos como el tipo de lactancia (pecho o biberón), el cambio de pañal, y el crecimiento en talla y peso del bebé. Pero su parte más potente es la consulta de dudas sobre lactancia y la información que ofrece sobre ello, avalado por expertas en lactancia materna.

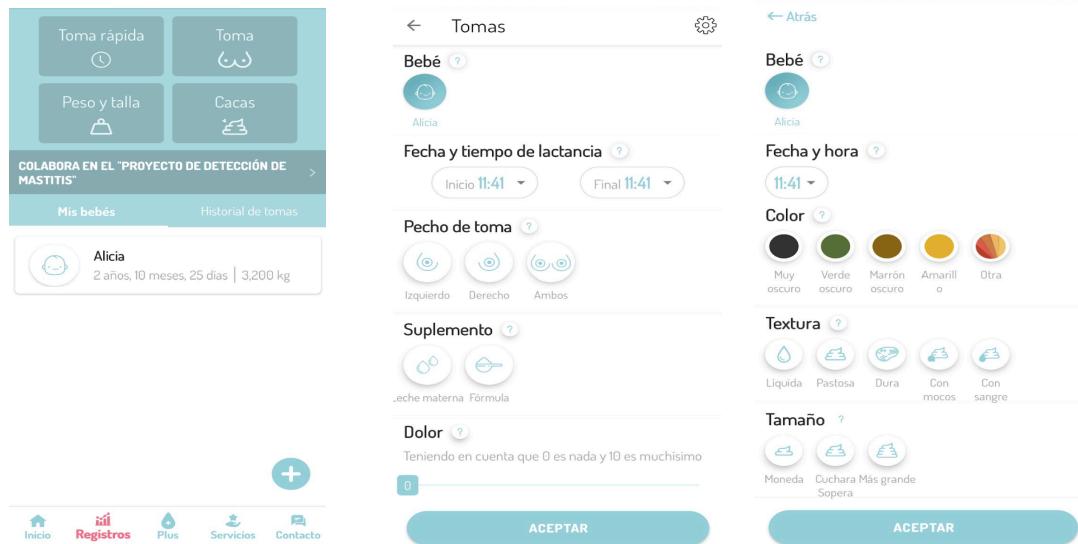
Entre las fortalezas de esta aplicación, destaca su contenido educativo y orientación específica sobre lactancia, lo que la convierte en un recurso muy apreciado para las madres que dan el pecho.

La interfaz de usuario es fácil de usar y no tiene publicidad, lo que mejora una interacción fluida.

Aunque LactApp es muy completa en el tema de asesoramiento en lactancia, tiene ciertas limitaciones, como, por ejemplo, que carece de funciones para el seguimiento de otras actividades esenciales en el desarrollo del bebé, como el sueño o los hitos de desarrollo.

Esta limitación la convierte en una herramienta que, si bien es muy potente y útil para la lactancia materna, no sirve para documentar de forma integral el desarrollo infantil.

Además, en el registro de datos se pide completar muchos campos, algo difícil cuando tienen que registrar los datos mientras cuidan del bebé.



Ejemplo de pantallas de la aplicación LactApp

2.2. Análisis de tendencias y vacíos en el mercado

El estudio de las aplicaciones seleccionadas muestra varias de las tendencias principales, así como las limitaciones más comunes.

Las aplicaciones de registro de rutinas de cuidados del bebé se centran mayoritariamente en la monitorización de la alimentación (pecho o biberón), cambio de pañales, así como el sueño. Este hecho responde a la necesidad de los padres de registrar las actividades cotidianas, ya que son patrones de comportamiento que sirven para el cuidado y el seguimiento de la salud del bebé en un momento en que la familia tiene muchos cambios.

Sin embargo, pese a ser de utilidad, también se han evidenciado múltiples vacíos. En primer lugar, algunas de estas herramientas se especializan en uno o pocos aspectos del cuidado del bebé (por ejemplo LactApp en la lactancia), lo que limita a las personas que deseen llevar un registro de varias categorías, teniendo que hacer uso de otras herramientas.

Por otra parte, aplicaciones como Erby y Baby Daybook presentan problemas de usabilidad provocados por interfaces muy sobrecargadas o poco intuitivas.

En el caso de Baby Daybook, este problema se incrementa debido a la experiencia intrusiva de la publicidad que interrumpe el flujo de uso en momentos críticos.

Un vacío importante es la falta de integración con la escuela infantil o permitir registrar las rutinas de un mismo bebé desde dos dispositivos diferentes.

Otro vacío es la falta de funcionalidades para registrar hitos de desarrollo y para registrar eventos especiales que son importantes para el seguimiento del crecimiento infantil.

Según lo que hemos podido ver en este análisis, **FirstYear** se plantea como una aplicación destinada a cubrir algunos de los vacíos identificados en el mercado de aplicaciones para el registro de cuidados de bebés.

Nuestra aplicación tiene como objetivo ofrecer una plataforma que permita registrar de manera integrada y personalizada las actividades diarias del bebé, no solo en categorías básicas como alimentación y sueño, sino también en áreas que las aplicaciones actuales no cubren de manera integral, como los hitos de desarrollo.

Buscamos diferenciarnos gracias a la interfaz limpia, intuitiva y sin publicidad, mejorando así la experiencia del usuario y facilitando el uso de la aplicación en momentos críticos.

Además, gracias a este análisis, se han podido plantear mejoras y ampliaciones para las siguientes versiones que se van a realizar de la aplicación.

3. Metodología utilizada

En este apartado vamos a presentar la metodología utilizada para llevar a cabo este proyecto, además de ver el ciclo de vida y las fases en las que hemos dividido el diseño de la aplicación.

3.1. Metodología kanban

En este caso hemos utilizado la metodología ágil kanban, hemos elegido este tipo de metodología ya que la hemos utilizado en otros proyectos con gran éxito. Según esta metodología, primero hemos identificado las tareas a realizar y después, las hemos organizado en un tablero digital de forma que quedan cuatro columnas:

- **Backlog:** contiene todas las tareas pendientes de desarrollo.
- **En progreso:** se incluyen las tareas en curso, priorizando las que son esenciales para el funcionamiento básico de la aplicación.
- **En revisión:** contiene las tareas completadas, pero que requieren revisión antes de cerrarlas.
- **Completadas:** se incluyen las tareas ya finalizadas y revisadas.

Gracias a esta forma de trabajar, hemos podido ver claramente qué tareas estaban pendientes, cuáles estaban en proceso y cuáles terminadas, ayudando de esta forma a organizarnos y trabajar de forma eficaz, ahorrando tiempo y esfuerzo.

3.2. Ciclo de vida iterativo incremental

En este caso hemos elegido un ciclo de vida iterativo incremental para desarrollar y mejorar las funcionalidades de la aplicación de forma progresiva y por ciclos. En cada iteración del ciclo se han agregado funcionalidades nuevas, teniendo el control sobre el programa y su avance.

A continuación se muestran las diferentes iteraciones que se han llevado a cabo:

Base del proyecto: en esta primera iteración, se estableció la estructura básica de la aplicación, organizando las pantallas y botones para crear un boceto de la interfaz gráfica de usuario. Además, se valoraron qué datos se recogerían en cada una de ellas.

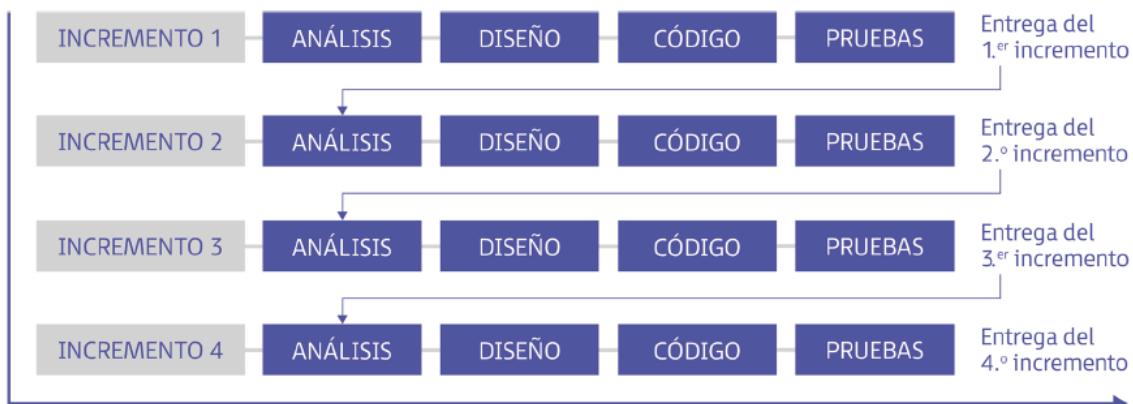
Interfaz gráfica de usuario: en esta iteración, se crearon las pantallas ideadas en la fase anterior, incluyendo la interactividad entre ellas.

Funcionalidades esenciales: se han añadido funcionalidades básicas para el funcionamiento de la aplicación como las pantallas de registro e inicio de sesión, además de los botones para seleccionar la fecha y hora actual de forma automática.

Base de datos: en esta fase, se ha configurado la base de datos, incorporando las funcionalidades al registrarse, iniciar sesión, añadir datos de los registros de los bebés y sus rutinas.

Mejora de la experiencia de usuario: se han añadido imágenes atractivas que representen bien cada opción, además de una paleta de colores para el fondo y los botones, de forma que sea atractiva y acorde a su uso por parte del usuario.

Pruebas y corrección de errores: en esta fase, se realizaron pruebas finales de usabilidad. Las pruebas incluyeron evaluaciones internas de cada proceso y pruebas con usuarios reales, que permitieron ajustar y perfeccionar detalles.



Esquema del modelo iterativo incremental. Fuente: Ilerna

A continuación, se detallan cada una de las fases de cada iteración:

- **Fase de análisis:** se definen los objetivos y funcionalidades a desarrollar en cada iteración.
- **Fase de diseño:** se crean los diseños y se define la estructura de cada funcionalidad.
- **Fase de código:** se desarrolla y codifica lo que se ha diseñado.
- **Fase de pruebas:** en esta fase se prueban las funcionalidades añadidas y se modifican los errores necesarios.

4. Tecnologías y herramientas utilizadas en el proyecto

Para desarrollar este proyecto hemos seleccionado varias tecnologías, herramientas y lenguajes. Cada una de ellas se ha elegido de forma minuciosa y teniendo en cuenta las características y ventajas de cada una de ellas según las necesidades del proyecto. A continuación se detallan cada una de ellas y por qué se han elegido:

Sistema operativo: Android

Android es el sistema operativo, o SO, elegido para este proyecto. Se ha seleccionado este sistema operativo, ya que es el más utilizado en los dispositivos móviles por ser de código abierto y gratuito para los fabricantes.

Además, en nuestro caso, era importante seleccionar este sistema operativo, ya que los dispositivos físicos de testing de los que disponemos tienen un sistema Android.

Lenguaje de programación: Kotlin

Kotlin es un lenguaje moderno desarrollado por JetBrains, diseñado para trabajar de forma conjunta con Java y optimizado para la creación de aplicaciones Android. Su punto fuerte en este caso es que su curva de aprendizaje es sencilla.

Es el lenguaje más utilizado para realizar aplicaciones en Android Studio y hay multitud de vídeos en YouTube para poder documentarnos y aprender sobre ello viendo ejemplos de otras personas que desarrollan aplicaciones con este lenguaje en Android Studio.

Entorno de desarrollo: Android Studio

Android Studio es un IDE, o entorno de desarrollo integrado, oficial para el sistema operativo Android. En este entorno de desarrollo se pueden diseñar, codificar y probar las aplicaciones móviles de Android.

Hemos seleccionado este entorno de desarrollo debido a que se integra perfectamente con el sistema Android y permite un buen flujo de trabajo al permitir crear la interfaz XML de forma fácil y rápida, poder codificar la aplicación e incluso probarla desde la propia herramienta.

Sistema de gestión de base de datos, SGBD: Firebase Firestore

Firebase Firestores es un sistema de gestión de base de datos NoSQL en la nube. Está desarrollada por Google y gracias al plan gratuito permite almacenar información y sincronizarla en tiempo real con un límite de almacenamiento, de escritura de datos y consultas suficientemente amplio para nuestra aplicación.

Hemos seleccionado Firestore por su capacidad para manejar datos en tiempo real, permitiendo que las actividades de registro de datos del bebé se sincronicen al instante entre múltiples usuarios.

Herramienta para desarrollar la documentación: Google Drive

Google Drive y sus herramientas como Hojas de cálculo, Presentaciones o Documentos nos permiten crear la documentación del proyecto, poder guardarla en la nube, de forma que se puede modificar desde diferentes equipos y mantener los documentos actualizados. Además, podemos guardar copias de seguridad del código desarrollado para recuperar versiones anteriores.

Herramienta de gestión de proyectos: Monday

Monday es una plataforma web en la que se pueden gestionar proyectos y tareas, permitiendo llevar una organización adecuada de las tareas. En nuestro caso, podemos crear tableros de Kanban y diagramas de Gantt, necesarios para gestionar y organizar las tareas del proyecto.

Se ha seleccionado esta herramienta por su interfaz sencilla de utilizar, visual e intuitiva, además de su gratuidad. Gracias a esta herramienta podemos ver el progreso del proyecto, el cumplimiento de plazos y la división de tareas de forma sencilla.

Herramienta para crear diagramas: Lucidchart

Lucidchart es una herramienta web de creación de diagramas. Permite crear diagramas como el diagrama de clases, diagrama de entidad-relación o diagrama de casos de uso.

Hemos seleccionado esta herramienta para elaborar los diagramas que representan visualmente la arquitectura de la aplicación y la estructura de la base de datos.

Su interfaz intuitiva facilita la creación rápida y detallada de diagramas, mejorando la claridad en la presentación del diseño y funcionamiento del sistema. Como pegas, hemos visto que solo permite crear 3 archivos que se pueden editar y hay un número limitado de objetos dentro de cada archivo.

Herramienta para crear iconos: ChatGPT Logo Creator

ChatGPT Logo Creator permite la creación personalizada de logos e iconos siguiendo una paleta de colores específica y un estilo visual uniforme.

Esta herramienta nos ha permitido desarrollar iconos que siguen el estilo visual definido, utilizando la gama de colores seleccionada para la interfaz de la aplicación, asegurando una estética profesional y sencilla.

Al no tener un equipo de arte para diseñar los iconos, hemos buscado alternativas como los bancos de imágenes libres, pero no encajaban bien con el diseño de la aplicación, por lo que se ha optado por el uso de inteligencia artificial para su creación.

Gracias al plan gratuito podíamos descargar tres imágenes cada día y, haciendo un uso correcto del prompt, hemos podido obtener unos iconos que representan de forma correcta cada elemento, siguiendo un estilo definido.

Herramienta de grabación de vídeo: OBS Studio

OBS Studio es un software de código abierto para grabación de vídeo que permite grabar la pantalla del ordenador a la vez que se graba la cámara del ordenador, algo necesario al crear el vídeo de presentación y demostración del proyecto.

Hemos seleccionado esta herramienta debido a que es fácil de utilizar y permite grabar en alta calidad.

5. Planificación, diagnóstico y contexto laboral

En este apartado vamos a mostrar la planificación que se ha llevado a cabo en el proyecto y un diagnóstico sobre las fortalezas y debilidades del mismo.

5.1. Planificación

En un proyecto, la planificación tiene gran relevancia, ya que tenemos que organizar el tiempo que tenemos para terminarlo y valorar si es factible o no. Incluso, valorando el tiempo que necesitamos para cada apartado, podemos realizar un presupuesto, en caso necesario, viendo el esfuerzo según el tiempo disponible.

Para la planificación de este proyecto hemos realizado un diagrama de Gantt, estimando los tiempos necesarios para cada tarea general. Estas tareas generales se subdividen en otras tareas específicas, necesarias para poder organizar el trabajo.

A continuación se muestra el diagrama del tiempo estimado y las dependencias entre tareas:

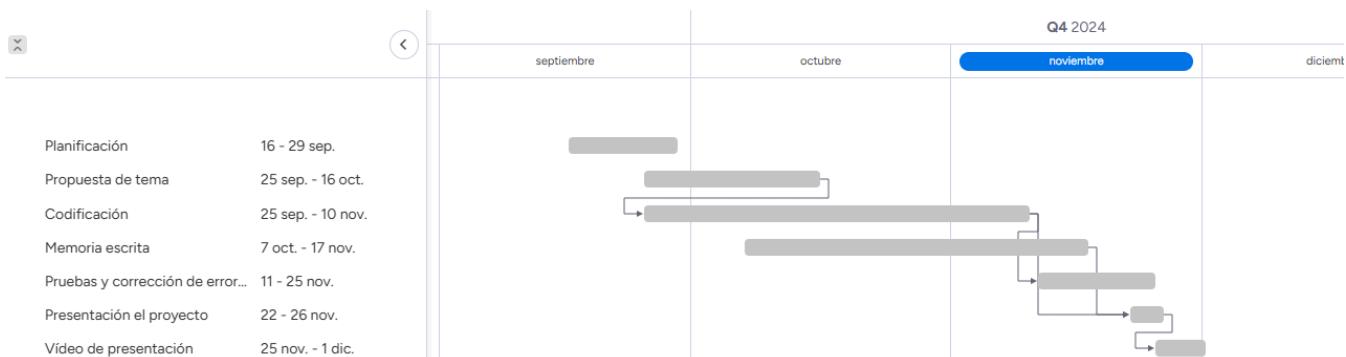


Diagrama de Gantt tiempo estimado. Realizado con [Monday](#). Se puede ver a página completa en el [anexo](#).

A lo largo del proyecto se han tenido que ajustar tiempos según avanzaba el desarrollo del mismo.

La comparación de estos diagramas permite observar la secuenciación de actividades, destacando las diferencias entre el tiempo previsto y el realmente empleado, lo cual es útil para comprender y ajustar el desarrollo del proyecto en tiempo real.

Por eso, vemos necesario presentar el diagrama real con las modificaciones de plazos que se han llevado a cabo.

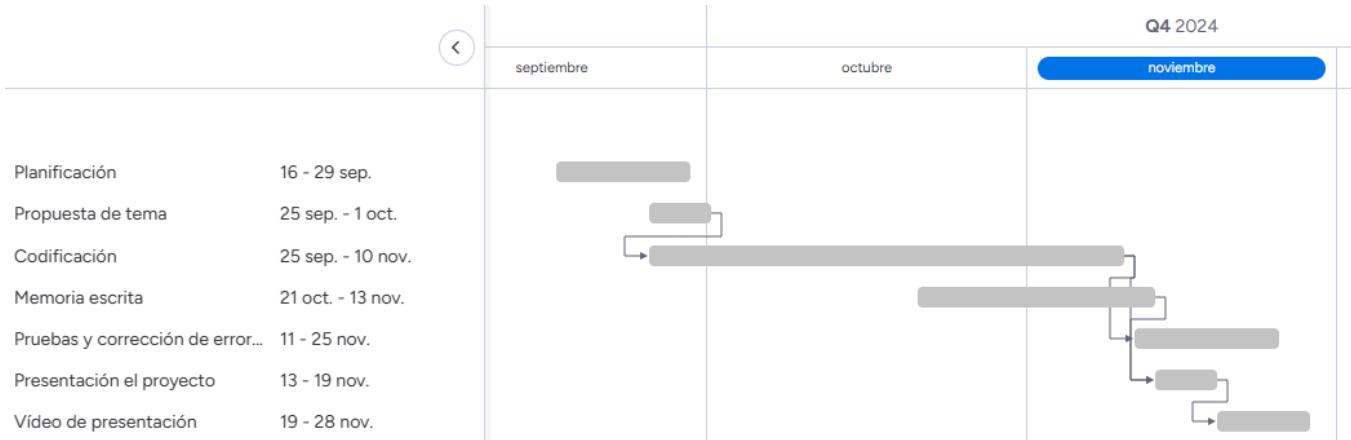


Diagrama de Gantt tiempo real. Realizado con [Monday](#). Se puede ver a página completa en el [anexo](#).

Viendo ambos diagramas, podemos comparar los tiempos reales con los estimados de cada tarea general:

Planificación: el tiempo se mantuvo constante, del 16 al 29 de septiembre, cumpliendo con el tiempo estimado inicialmente.

Propuesta de tema: previsto del 25 de septiembre al 16 de octubre, fue concluido antes, pasando a ser el tiempo del 25 de septiembre al 1 de octubre.

No vemos modificaciones en el tiempo de codificación, a pesar de que depende de esta tarea, ya que comenzamos a desarrollar el proyecto antes de tener la validación definitiva. Esto se pudo hacer así porque los cambios que podríamos tener eran mínimos y el tiempo era muy ajustado.

Codificación: se mantuvo el tiempo planificado, del 25 de septiembre al 10 de noviembre.

Redacción de la memoria: en este caso se ajustó, comenzando más tarde pero acortando el plazo, de un periodo inicial del 7 de octubre al 17 de noviembre a uno final del 21 de octubre al 13 de noviembre. Este tiempo se ajustó para intentar tener la mayor parte posible programada, facilitando en algunos apartados la redacción.

Pruebas y corrección de errores: no sufrió modificaciones, manteniéndose del 11 al 25 de noviembre.

Presentación: inicialmente del 22 al 25 de noviembre, se adelantó del 13 al 19 de noviembre.

Vídeo de la presentación: Pasó de ser del 25 de noviembre al 1 de diciembre a desarrollarse entre el 19 y el 28 de noviembre.

Este ajuste de tiempos en algunas de las tareas nos ha permitido tener una mayor flexibilidad y agilidad, reduciendo el margen de tiempo de algunos apartados sin afectar la calidad del trabajo final.

5.2. Diagnóstico

Para llevar a cabo un diagnóstico sobre el proyecto, es necesario realizar un análisis DAFO. Este análisis es clave para entender las características del proyecto en profundidad, revelando tanto sus aspectos fuertes como aquellos que necesitan ser mejorados.

A continuación mostramos las fortalezas de la idea propuesta:

- **Facilidad de uso:** se ha diseñado una aplicación con una interfaz sencilla e intuitiva, pensada para que los usuarios puedan realizar el registro de los datos con facilidad.
- **Enfoque específico y útil:** la aplicación ofrece una herramienta muy útil para las familias y cuidadores, facilitando el registro detallado y específico de las actividades diarias de los bebés.
- **Integración de herramientas actuales:** se han utilizado herramientas y lenguajes actuales en la elaboración de la aplicación, como Firebase, Kotlin o Android Studio. Permitiendo la actualización de forma sencilla de la aplicación, el uso de una base de datos segura e instantánea y permitiendo que los usuarios tengan un acceso sencillo a la información guardada.

Por contra, podemos ver ahora las debilidades de esta idea:

- **Depende de Internet:** esta aplicación depende de Internet para el acceso a los datos, por lo que, en zonas con baja cobertura, podría ser un problema.
- **Capacitación inicial del usuario:** a pesar de ser una aplicación intuitiva y sencilla de utilizar, puede que haya usuarios que necesiten una pequeña guía inicial para adaptarse al flujo de pantallas.
- **Requiere actualizaciones:** debido al uso de tecnologías que cambian constantemente y que esta aplicación está en su primera versión, se van a necesitar lanzar diferentes actualizaciones. Esto tiene como consecuencia la necesidad de un mantenimiento continuo y adaptación a las nuevas versiones de software y requisitos de seguridad.

A continuación mostramos las oportunidades:

- **Expansión de funcionalidades:** debido a que se lanza la primera versión de esta aplicación, se prevé que haya otras versiones en las que añadan nuevas funcionalidades y características como: notificaciones y recordatorios automáticos, informes personalizados, consejos de cuidado del bebé basándose en su edad y registros, entre otras.
- **Ampliación a nuevos mercados:** la demanda de aplicación para el seguimiento de las rutinas de los bebés aumenta cada vez más. Se podría ampliar el mercado, incluyendo una versión para escuelas infantiles o incluso para el mercado internacional, traduciendo el contenido y adaptándolo a la cultura.
- **Colaboración con pediatras:** se podría ampliar la ayuda que ofrece la aplicación incluyendo a pediatras para generar consejos, guías específicas o avisos.

Por último, incluimos las amenazas que hemos detectado:

- **Competencia:** el mercado de aplicaciones para el cuidado infantil cuenta con varias opciones, por lo que la diferenciación es imprescindible para captar y mantener usuarios.
- **Protección de datos:** la gestión de la información de los bebés requiere que se haga con cuidado y de forma segura para no incumplir las normas de privacidad.
- **Cambios rápidos en la tecnología:** al depender de plataformas como Firebase, puede requerir mantenimiento y actualizaciones periódicas, lo que incrementa el tiempo de trabajo y el coste del mismo.

6. Análisis del proyecto

Este apartado tiene gran importancia, ya que en él se definen las características, su estructura y objetivos.

En este caso, el objetivo principal es facilitar a las familias el registro y el seguimiento del cuidado diario y los hitos importantes en el desarrollo de sus bebés durante los primeros años de vida.

En relación a este objetivo y a los objetivos específicos planteados en la introducción de este documento se va a realizar un análisis en el que se identificarán los requisitos funcionales, no funcionales y se expondrán diferentes diagramas: entidad relación, casos de uso y diagrama de clases.

6.1. Requisitos funcionales y no funcionales

Es importante definir los requisitos funcionales y no funcionales de la aplicación, de esta forma nos aseguramos de que los objetivos propuestos se van a cumplir.

Para ello se han definido requisitos funcionales, especificando las funciones esenciales que debe cumplir, entre ellas el registro de usuarios, el login, el registro de bebés y sus datos de cuidados, así como la consulta y modificación de ese registro de datos.

Además de los requisitos funcionales, hay que tener en cuenta los no funcionales. Los requisitos no funcionales son esenciales para definir las características de calidad que debe tener la aplicación, como, por ejemplo, la usabilidad, la apariencia, el rendimiento o la compatibilidad entre otras.

Estos aspectos ayudan a que la aplicación, además de ser práctica y funcional, sea fácil de usar, rápida y segura.

A continuación, se detallan los requisitos funcionales y no funcionales que nos ayudan a crear la aplicación con todo lo necesario:

Requisitos funcionales:

Identificador		Descripción
RF1	Registro de usuarios	La aplicación debe permitir que se registren los usuarios. Los datos que se solicitan son: <ul style="list-style-type: none">- Nombre.- Correo electrónico.- Contraseña.- Número de teléfono (opcional)

RF2	Login de usuario	Los usuarios registrados deben poder iniciar sesión. Los datos que se piden son: <ul style="list-style-type: none"> - Correo electrónico. - Contraseña.
RF3	Permanencia del usuario	Una vez el usuario ha iniciado sesión, debe mantenerse iniciada aunque se cierre la app. Solo se cerrará la sesión si se pulsa el botón para ello.
RF4	Añadir bebés	La aplicación debe permitir añadir y eliminar bebés. Los datos que se solicitan para añadir bebés son: <ul style="list-style-type: none"> - Nombre. - Fecha de nacimiento. - Género. - Rol (madre, padre, familiar, cuidador/a).
RF5	Registro actividad bebé	La aplicación debe permitir registrar la actividad del bebé.
RF6	Registro lactancia materna	La aplicación debe permitir registrar las tomas de lactancia materna. Los datos que se guardan son: <ul style="list-style-type: none"> - Pecho derecho o izquierdo. - Fecha y hora en que empieza. - Fecha y hora en que termina. - Comentarios (es opcional).
RF7	Registro pañal	La aplicación debe permitir registrar los cambios de pañal. Los datos que se guardan son: <ul style="list-style-type: none"> - Pis, caca o ambos. - Fecha y hora del cambio. - Comentarios (es opcional).
RF8	Registro sueño	La aplicación debe permitir registrar las horas de sueño. Los datos que se guardan son: <ul style="list-style-type: none"> - Fecha y hora en que empieza. - Fecha y hora en que termina. - Comentarios (es opcional).
RF9	Registro alimentación con biberón	La aplicación debe permitir registrar las tomas de biberón. Los datos que se guardan son: <ul style="list-style-type: none"> - Leche materna o fórmula. - Fecha y hora en que empieza. - Fecha y hora en que termina. - Cantidad. - Comentarios (es opcional).
RF10	Registro hitos de desarrollo	La aplicación debe permitir registrar los hitos de desarrollo que consigue el bebé. Los datos que se guardan son: <ul style="list-style-type: none"> - Nombre del hito cumplido. - Fecha en la que se cumplió. - Comentarios (es opcional).
RF11	Visualización de registros	La aplicación debe ofrecer una vista de los registros. Mostrando un resumen de los registros

		y gráficos.
RF14	Seguridad	Los datos de los cuidadores y los bebés deben estar protegidos, evitando accesos no autorizados.

Requisitos no funcionales:

Identificador		Descripción
RNF1	Usabilidad	La aplicación debe ser fácil de usar, ser intuitiva y funcional, para que la puedan usar mientras cuidan de su bebé.
RNF2	Apariencia	La interfaz de usuario debe ser clara y contar con iconos que sean representativos y fáciles de entender.
RNF3	Rendimiento	La aplicación debe responder de forma rápida, evitando largos tiempos de carga.
RNF4	Compatibilidad	La aplicación debe ser compatible con las versiones recientes de Android.
RNF5	Adaptabilidad	La aplicación debe adaptarse a diferentes tamaños de pantalla.
RNF6	Escalabilidad de la aplicación	La aplicación debe permitir actualizaciones para poder añadir más funcionalidades.
RNF7	Escalabilidad de la base de datos	La aplicación debe permitir ampliar la base de datos si hay un gran número de usuarios.
RNF8	Seguridad de los datos	Los datos deben guardarse de forma segura.

6.2. Diagrama entidad-relación

En el diagrama entidad-relación podemos ver las entidades clave de la aplicación: cuidador, bebé, actividades que se registran. Además, podemos ver la relación entre ellas y los datos que almacena cada una para poder planificar la base de datos y su organización.

En el diagrama podemos ver que hay 8 tablas relacionadas, cumpliendo el requisito que se pide para el proyecto.



6.3. Casos de uso

Gracias a los diagramas de casos de uso podemos representar las interacciones posibles entre el usuario y la aplicación. En estos diagramas podemos ver que el cuidador se registra si no tiene cuenta, inicia sesión si ya está registrado, añade bebés, registra los datos del bebé y una vez se han registrado, se pueden ver los registros, modificarlos y eliminarlos.

Veremos estas situaciones de uso en el diagrama y los cuadros que se muestran a continuación:

Caso de uso:	Registro de usuario
Descripción	El sistema permite que un nuevo usuario se registre en la aplicación.
Precondición	El usuario no está registrado.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el sistema pide nombre, correo electrónico, contraseña y el teléfono que es opcional. ○ Paso 2: el usuario introduce los datos y selecciona <i>Crear cuenta</i>. ○ Paso 3: el sistema valida los datos, los guarda en la base de datos y finaliza el caso de uso.
Postcondición	El usuario está dentro de la aplicación.

Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si el usuario no ha introducido todos los datos obligatorios no puede pasar al paso 3 y muestra un mensaje de error y el caso de uso queda sin efecto. ○ Paso 3: si el correo electrónico ya está en uso o los datos no cumplen los requisitos, el sistema muestra un mensaje de error y el caso de uso queda sin efecto.
--------------------	--

Caso de uso:	Login
Descripción	El sistema debe permitir que el usuario registrado inicie sesión en la aplicación.
Precondición	El usuario está registrado.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el sistema pide correo electrónico y contraseña. ○ Paso 2: el usuario introduce los datos y selecciona <i>Acceder</i>. ○ Paso 3: el sistema verifica en la base de datos que los datos son válidos y finaliza el caso de uso.
Postcondición	El usuario está dentro de la aplicación.
Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si el usuario no ha introducido todos los datos obligatorios no puede pasar al paso 3 y muestra un mensaje de error y el caso de uso queda sin efecto. ○ Paso 3: si los datos no son válidos, el sistema no permite el acceso y el caso de uso queda sin efecto.

Caso de uso:	Añadir bebé
---------------------	-------------

Descripción	El sistema debe permitir que el usuario añada el perfil de un bebé.
Precondición	El usuario ha iniciado sesión.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el sistema muestra un formulario para añadir información del bebé (género, nombre, fecha de nacimiento,rol). ○ Paso 2: el usuario introduce la información y selecciona <i>Guardar</i>. ○ Paso 3: el sistema guarda los datos en la base de datos y finaliza el caso de uso.
Postcondición	El perfil del bebé está guardado en la aplicación.
Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si el usuario no ha introducido todos los datos obligatorios no puede pasar al paso 3 y muestra un mensaje de error y el caso de uso queda sin efecto.

Caso de uso:	Registrar actividad
Descripción	El sistema debe permitir que el cuidador registre actividades del bebé (pañales, alimentación, sueño, hitos).
Precondición	El usuario ha iniciado sesión y al menos ha añadido un bebé.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el sistema muestra un menú de actividades disponibles (alimentación, cambio de pañal, sueño, hitos desarrollo). ○ Paso 2: el usuario selecciona una actividad, completa los datos requeridos y pulsa <i>Guardar</i>.

	<ul style="list-style-type: none"> ○ Paso 3: el sistema guarda los datos en la base de datos y finaliza el caso de uso.
Postcondición	La actividad del bebé está guardada en la aplicación.
Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si el usuario no ha introducido todos los datos obligatorios no puede pasar al paso 3 y muestra un mensaje de error y el caso de uso queda sin efecto.

Caso de uso:	Visualizar historial de actividades registradas
Descripción	El sistema debe permitir que el cuidador vea las actividades registradas del bebé (pañales, alimentación, sueño, hitos).
Precondición	El usuario ha iniciado sesión, tiene al menos un bebé añadido y alguna actividad registrada.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el usuario selecciona el perfil del bebé cuyo registros quiere ver. ○ Paso 2: el sistema muestra una lista con las actividades registradas. ○ Paso 3: el usuario revisa la información y finaliza el caso de uso.
Postcondición	El usuario ha consultado el historial de actividades del bebé.
Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si no hay actividades registradas, el sistema muestra un mensaje indicando que no existen datos.

Caso de uso:	Eliminar actividades registradas
---------------------	----------------------------------

Descripción	El sistema debe permitir que el cuidador elimine actividades registradas del bebé (pañales, alimentación, sueño, hitos).
Precondición	El usuario ha iniciado sesión, tiene al menos un bebé añadido y alguna actividad registrada.
Secuencia normal	<ul style="list-style-type: none"> ○ Paso 1: el usuario selecciona el perfil del bebé cuyo registros quiere eliminar. ○ Paso 2: el sistema muestra una lista con las actividades registradas. ○ Paso 3: el usuario selecciona qué dato quiere eliminar, pulsa Eliminar y finaliza el caso de uso.
Postcondición	El usuario ha eliminado el registro seleccionado del bebé.
Excepciones	<ul style="list-style-type: none"> ○ Paso 2: si no hay actividades registradas, el sistema muestra un mensaje indicando que no existen datos.

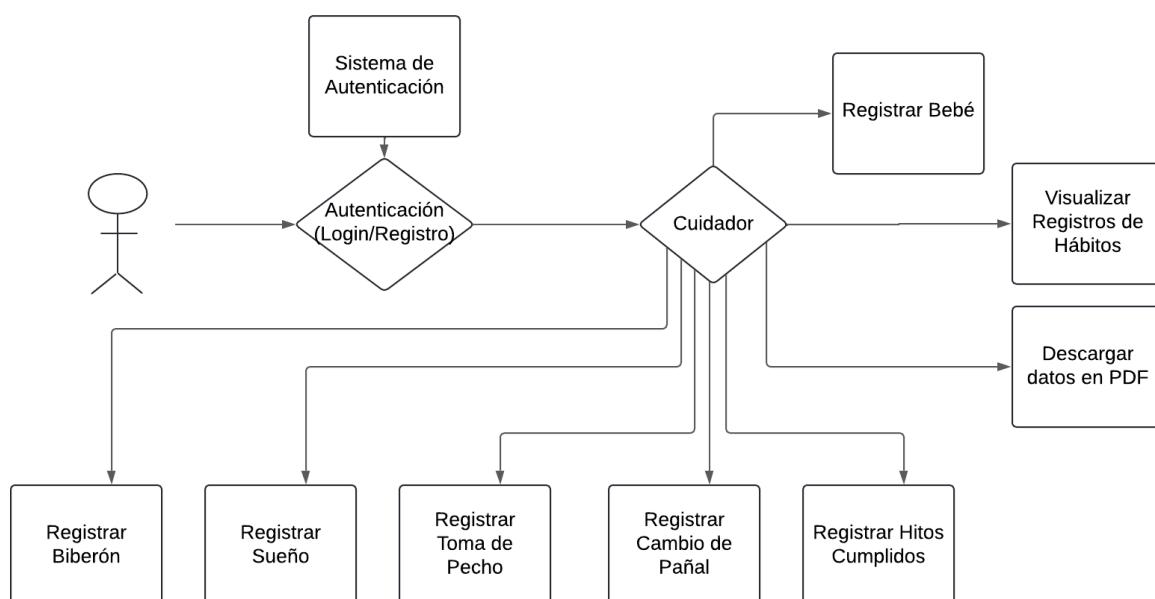


Diagrama casos de uso, elaboración propia.

6.4. Diagrama de clases

En este apartado vamos a mostrar la estructura interna de la aplicación, vista desde el punto de vista del código, ya que se representan las principales clases de la aplicación y la

relación entre ellas, mostrando los atributos y los métodos. Gracias a este gráfico se ve, de forma general, la organización del código y la lógica de la aplicación.

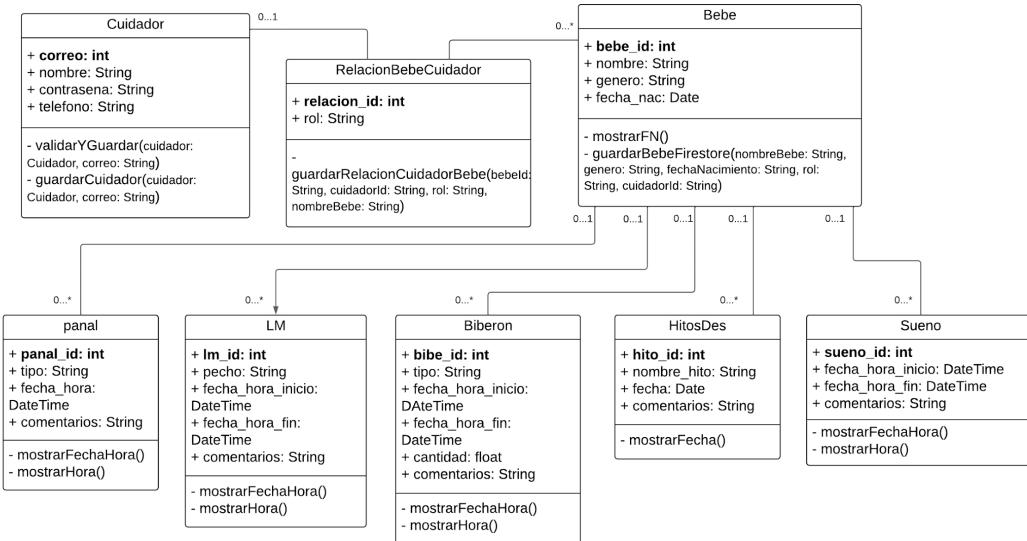


Diagrama de clases, elaboración propia. Se puede ver a página completa en el [anexo](#).

7. Diseño del proyecto

En este apartado vamos a detallar cómo se ha llevado a cabo este proyecto, viendo cosas como el diseño de las pantallas, la elección de colores, el diseño de los iconos, las clases utilizadas, las funciones o los ficheros.

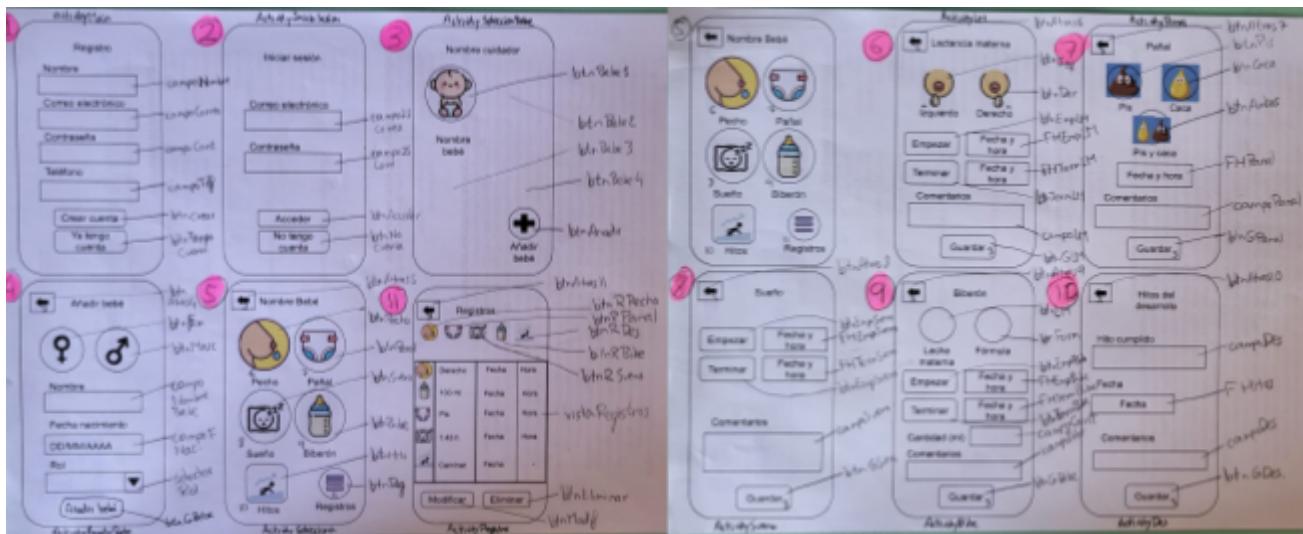
7.1. Diseño previo a la codificación

Para llevar a cabo el diseño de la aplicación hemos realizado bocetos a mano alzada de la estructura de cada pantalla, esto nos ha servido para pensar qué botones y opciones podía tener cada pantalla, cómo se navega entre las pantallas, la relación de unas a otras, el nombre de cada activity, qué datos tienen que guardar o incluso alguna idea de qué imagen poner en cada botón.



Boceto de las pantallas de la aplicación, elaboración propia.

Después de diseñar las pantallas a mano alzada, lo pasamos a digital para organizar la pantalla, ver tamaños, añadir la referencia de cada componente, seleccionar imágenes de ejemplo para los botones y añadir opciones que no habíamos contemplado, pero al realizar el diseño hemos valorado como necesarias. Obteniendo así una guía a la hora de realizar la interfaz de usuario y programar cada clase y sus componentes.



Diseño y organización de las pantallas de la aplicación, elaboración propia.

Una vez teníamos las clases, pantallas y componentes diseñados, seleccionamos una paleta de colores.

En este caso, al ser una aplicación de registro de rutinas de bebés, que en muchas ocasiones se va a utilizar en momentos de estrés, ya que se utiliza a la vez que los cuidadores atienden al bebé. Decidimos utilizar una gama de colores pastel, suaves, que transmitan calma y tranquilidad.

Entre los colores seleccionados encontramos violeta y amarillo como base de la aplicación. Los códigos de los colores utilizados como base son #FFF6D5 para el fondo y #9575CD para los botones.

Incluimos esta gama de colores suaves en los iconos de la aplicación.

En este caso, al no contar con un equipo de diseño y valorar la importancia de la homogeneidad de las ilustraciones mostradas en la aplicación, buscamos imágenes en diferentes bancos gratuitos sin éxito.

Por este motivo decidimos utilizar una tecnología en auge, la inteligencia artificial. Para ello decidimos minuciosamente el prompt a utilizar para obtener imágenes homogéneas que representarán lo que necesitábamos con la gama de colores que habíamos seleccionado y que quedan bien entre ellas.

Gracias a esta tecnología tenemos una aplicación con iconos representativos y que encajan perfectamente en el diseño.



Al no disponer de cuenta *premium*, hemos tenido que solicitar tres imágenes por día, para poder conseguir todos los iconos.

Una vez teníamos las imágenes, las hemos modificado en tamaño y forma, para redondear los bordes, con la herramienta web iloveimg.com.

Una vez tenemos todos los componentes necesarios, pasamos a programar.

7.2. Codificación de la interfaz

Gracias al diseño anterior, la codificación es mucho más sencilla.

Antes de pasar a detallar la codificación, creemos que es importante entender cómo es la aplicación.

Esta es una aplicación dirigida a las familias que tengan al menos un bebé y necesiten registrar datos de las rutinas del bebé, quizás tengan que hacer ese registro mientras cuidan del bebé. Por eso debe ser una aplicación amigable y agradable de utilizar en la que puedan registrar las rutinas de sus bebés de forma sencilla y sin tener que apuntar datos innecesarios.

Esta aplicación consta de once pantallas diferentes. Entre ellas encontramos la pantalla de registro de usuarios, en esta pantalla el usuario que no esté registrado debe introducir sus datos y recordarlos para poder volver a iniciar sesión cuando lo necesite.

Si el usuario ya tiene cuenta, puede acceder a la pantalla de inicio de sesión y acceder a la aplicación desde allí. Una vez ha iniciado sesión, el usuario se encuentra en una pantalla en la que puede seleccionar un bebé, si ya lo ha añadido, añadir bebés o cerrar la sesión.

Si el usuario decide añadir un bebé, tendrá que completar ciertos datos del bebé.

Una vez el usuario ha seleccionado el bebé en el que quiere registrar los datos, puede eliminar el bebé, si es necesario. También puede pulsar en cualquiera de las opciones para añadir datos: pecho, pañal, sueño, biberón o hitos de desarrollo. Lo que le llevará a la pantalla específica para guardar los datos de cada categoría.

Por último, puede acceder a la pantalla de registros, donde puede ver un resumen de los datos registrados e incluso eliminarlos si los selecciona en la tabla correspondiente.

Una vez tenemos claro en qué consiste la aplicación, cómo se han diseñado las diferentes pantallas y qué necesitamos, generamos las diferentes clases necesarias, con los nombres que ya hemos seleccionado en el diseño.

A continuación vemos las clases con las que vamos a trabajar en este proyecto y su función:

Activity_bibe: clase para llevar a cabo el registro de datos de la alimentación con biberón.

Activity_des: clase para registrar los hitos de desarrollo alcanzados.

Activity_panal: esta clase se utiliza para registrar los cambios de pañal del bebé.

Activity_sueno: clase para llevar a cabo el registro de los datos del sueño del bebé.

ActivityLM: clase para registrar las tomas de pecho que realiza el bebé.

Activity_seleccion_bebe: clase en la que se puede seleccionar el bebé con el que quieras registrar los datos, cerrar sesión o acceder a la pantalla para añadir un bebé.

activity_selecciona: en esta clase se puede seleccionar dentro de qué actividad quieras realizar el registro, acceder a la pantalla para ver los datos registrados y borrar el bebé.

ActivityInicioSesion: en esta clase está la programación para realizar el inicio de sesión.

ActivityAnadirBebe: clase para añadir un bebé en el cuidador correspondiente.

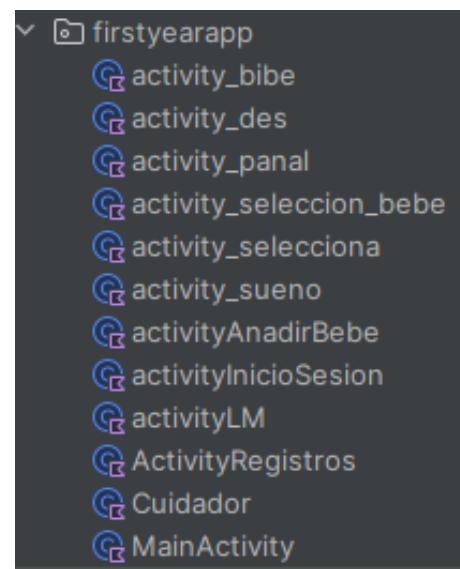
ActivityRegistros: gracias a esta clase podemos ver los datos registrados e incluso borrarlos de la base de datos.

MainActivity: en esta clase se ve la programación de la pantalla principal, la de registro.

A continuación, generamos todos los documentos XML con el diseño de cada pantalla, organizando los componentes y nombrándolos como habíamos diseñado.

Añadimos la interactividad entre pantallas modificando el archivo .kt para poder ver en el dispositivo móvil cómo queda cada una de las pantallas y ajustar tamaños y relaciones entre componentes.

A continuación creamos las funciones que necesitamos para ciertos botones:



Captura de la carpeta de Android Studio en la que se ven los archivos de las clases.

- **Botones para seleccionar fecha y hora y etiquetas para mostrarla:** para conseguir la fecha y la hora actuales hemos creado las funciones *mostrarFechaHora* y *mostrarHora* que se utilizan dentro de la función anterior.
- **Botón para seleccionar fecha:** en el caso de la pantalla de registro de desarrollo de hitos, solo hay que seleccionar la fecha, por lo que creamos una función llamada *mostrarFecha* en la que se puede seleccionar la fecha actual u otra.
- **Botón para seleccionar la fecha de nacimiento:** en este caso, como la fecha de nacimiento no puede ser la actual, se añade una función para seleccionar una fecha anterior. Se llama *mostrarFN*.
- **Mostrar botones sin color o con color:** se utiliza para mostrar los botones de los bebés añadidos según su estado. Con color si están añadidos en la base de datos o sin colores si no lo están.

Para conseguir esto hemos creado varias funciones, nombradas como *configurarBotones* para mostrar los botones sin color al principio, *mostrarBebes*, para mostrar los bebés que tiene el cuidador con el que se ha accedido.

- **Cerrar sesión:** se utiliza para cerrar la sesión que tenemos iniciada, esto es importante, ya que al guardarse la sesión iniciada en el dispositivo, habría que desinstalar la aplicación o borrar datos de la aplicación para poder iniciar sesión con otra cuenta. Para ello hemos creado la función *cerrarSesion*.
- **Guardar los datos en la base de datos:** se han creado diferentes funciones para guardar los datos en la base de datos. Estas se llaman: *guardarDatosSueno*, *guardarDatosPanal*, *guardarDatosBiberon*, *guardarDatosHito*, *guardarDatosLM*, *validarYGuardar*, *guardarCuidador*, *guardarRelacionCuidadorBebe* y *guardarBebeFirestore*.

Debido al equipo informático del que disponemos, el emulador de la herramienta Android Studio va muy lento o se bloquea, así que decidimos generar los archivos .apk necesarios para probar cada una de las pantallas o modificaciones de ellas.

Una vez tenemos toda la interfaz creada y la interactividad entre pantallas con los botones y la funcionalidad de los botones, pasamos a crear la base de datos.

7.3. Base de datos

Para poder continuar con la programación de las pantallas de la aplicación, una vez las tenemos ya creadas, pasamos a crear la base de datos. En este caso hemos seleccionado Firebase Firestore de Google para nuestra base de datos. Seguimos los pasos que nos indican en firebase.google.com para crear y vincular la base de datos con nuestra

aplicación, añadiendo el código que nos indican y permitiendo la lectura y escritura de datos.

Una vez creada la base de datos, ya podemos probar la programación creada para guardar los datos y ver que se han guardado bien en la web de Firebase.

bebeId	"8be7aecf-e538-4331-87e6-c93dfeb08870"
comentario	"jhjj de la luz de la luz"
fechaInicio	"3/10/2024 02:15"
fechaTermina	"24/10/2024 19:15"
lmId	"47a9a9bb-eacb-426c-9bdc-89750975c02d"
tipoPecho	"Izquierdo"

Vista de la base de datos con varios registros guardados.

7.4. Codificación de la vista y eliminación de datos

Una vez se ha probado la programación y la base de datos, pasamos a realizar el código para mostrar los datos registrados, que es la parte que más nos ha costado llevar a cabo.

Para poder ver los datos que ha registrado el cuidador, hemos creado varias funciones:

- **agregarEncabezado:** creada para añadir un encabezado según la opción que se pulse.
- **Mostrar datos:** funciones para recuperar los datos y mostrarlos en la vista, según la categoría que selecciones: mostrarDatosPecho, mostrarDatosPanal, mostrarDatosSueno, mostrarDatosBibe y mostrarDatosDesarrollo.
- **calcularDuracion:** función necesaria para calcular la duración de las tomas y el sueño, restando los tiempos marcados como inicio y fin.
- **actualizarVista:** para mostrar los datos después de borrar datos.
- **limpiarVistaRegistros:** creada para quitar los datos que hay en la tabla al cambiar de una opción a otra.
- **agregarRegistroEnVista:** función para mostrar los datos en la vista, se utiliza dentro de las funciones de cada categoría.

Una vez se realizó la codificación necesaria para que la aplicación tuviese las funciones básicas planteadas, se llevó a cabo un testing general de cada pantalla. Hay que tener en cuenta que anteriormente se han realizado pruebas al incorporar nuevas funcionalidades o servicios. Podemos ver la organización de las pruebas en el apartado 8 del documento, Despliegue y pruebas.

Una vez se han realizado las pruebas y se ha visto que todo funciona de forma correcta, tal cual se ha diseñado, se lleva a cabo una fase de mejoras y puesta a punto final de la aplicación.

7.5. Mejoras y puesta a punto

En esta fase se han revisado las funcionalidades y se han añadido opciones como poder eliminar al bebé que se añadió anteriormente, poner el nombre del bebé en la pantalla de selección de registro de datos, marcar con un borde los botones de la vista de datos, entre otras mejoras.

7.6. Ficheros y carpetas

A continuación mostramos un pequeño resumen con las carpetas y los ficheros más relevantes que hemos utilizado para la realización de la aplicación.

Dentro de la carpeta "app" encontramos la carpeta "manifest", "kotlin+java" y "res".

En la carpeta "manifest" tenemos el archivo AndroidManifest.xml en el que se ha añadido un icono representativo de la aplicación y tenemos todas las activity.

En la carpeta "kotlin+java" encontramos la carpeta "com.example" que contiene la carpeta "firstyearapp" con todos los archivos .kt que se han utilizado para realizar esta aplicación.

En la carpeta "res" tenemos varias carpetas. Empezamos por la carpeta "drawable", que contiene todos los archivos de imagen que necesitamos para la aplicación.

La siguiente carpeta se llama "layout", que contiene todos los archivos XML con los que se han diseñado las interfaces de las diferentes pantallas.

A continuación, la carpeta "mipmap", que tiene los archivos para cambiar el icono de la aplicación.

Y por último, podemos encontrar el archivo de colores de la aplicación, llamado colors.xml, en la carpeta "values".

Además de la carpeta "app" y sus subcarpetas, encontramos la carpeta "Gradle Scripts" con los archivos build.gradle.kts necesarios para configurar la aplicación.

8. Despliegue y pruebas

En este apartado vamos a mostrar las diferentes herramientas y pruebas utilizadas para comprobar el funcionamiento correcto de la aplicación y sus funciones.

8.1. Herramientas utilizadas

Hemos utilizado diferentes herramientas de depuración para llevar a cabo las pruebas, utilizando avisos que se muestran en la pantalla cuando hay un error o cuando se lleva a cabo la tarea, además de mostrar avisos en Android Studio para saber qué va mal o bien.

Hemos utilizado las notificaciones en pantalla para avisar al usuario sobre errores como intentar registrar un correo que ya está en la base de datos o intentar guardarla sin completar todos los datos necesarios, entre otros. Para ello hemos utilizado las notificaciones con Toast.

Por otro lado, hemos utilizado los avisos Log para mostrar información detallada sobre los procesos en la consola Logcat. Esta herramienta ha sido muy útil en la creación de la aplicación para poder verificar que los procesos se llevaban a cabo correctamente.

El único problema es que, debido al dispositivo que utilizamos para programar la aplicación, teníamos que cerrar todos los procesos para que se pudiese cargar el emulador de Android Studio y ver los avisos del Logcat.

Pero gracias al uso de Logcat, fue posible identificar excepciones, errores de código y problemas en el flujo lógico que no eran visibles para el usuario con los avisos Toast. Esto nos permitió realizar correcciones precisas y rápidas, asegurando la funcionalidad adecuada de cada pantalla y caso de uso.

El uso combinado de estas herramientas facilitó la validación de los casos de uso desde múltiples perspectivas, asegurando que tanto la interacción del usuario como el funcionamiento interno del sistema cumplieran con los requisitos especificados.

8.2. Pruebas de caja negra

Una vez se ha programado la aplicación, funciona, no se cierra de forma inesperada y se han añadido los avisos utilizando Toast, pasamos a realizar las pruebas de los casos de uso. Para ello vamos a realizar una .apk y probar cada uno de los casos de uso como si fuésemos usuarios.

A continuación mostramos las pruebas realizadas:

Caso de uso:	Registro de usuario
Objetivo probado	Verificar que el sistema permite registrar correctamente a un usuario nuevo, cumpliendo con los requisitos de validación de datos.
Requisitos probados	<ul style="list-style-type: none"> ● El sistema solicita y valida los datos obligatorios (el número de teléfono es el único dato opcional). ● El sistema detecta y notifica errores si los datos no son válidos. ● Los datos del usuario se almacenan correctamente en la base de datos.
Pruebas realizadas	
<ol style="list-style-type: none"> 1. Entrada válida: introducir todos los campos correctamente y confirmar que el sistema registra al usuario. 2. Falta de datos obligatorios: dejar campos obligatorios vacíos y verificar que se muestra un mensaje de error. 3. Correo ya registrado: intentar registrar un correo existente y confirmar que el sistema muestra un mensaje de error. 	

Caso de uso:	Login
Objetivo probado	Validar que el sistema permite iniciar sesión solo con credenciales válidas.
Requisitos probados	<ul style="list-style-type: none"> ● El sistema solicita y valida las credenciales del usuario. ● Los usuarios con datos válidos acceden correctamente. ● Los usuarios con datos inválidos reciben un mensaje de error adecuado.
Pruebas realizadas	
<ol style="list-style-type: none"> 1. Credenciales correctas: ingresar un correo y contraseña válidos y confirmar que el usuario accede a la aplicación. 	

2. **Campos vacíos:** dejar uno o ambos campos vacíos y comprobar que se muestra un mensaje de error.
3. **Correo no registrado:** usar un correo no registrado y verificar que el sistema no permite el acceso.
4. **Contraseña incorrecta:** introducir una contraseña incorrecta para un correo registrado y validar que el acceso es rechazado.

Caso de uso:	Añadir bebé
Objetivo probado	Comprobar que el sistema permite añadir el perfil de un bebé con datos válidos.
Requisitos probados	<ul style="list-style-type: none"> • El sistema solicita todos los datos obligatorios (nombre, género, fecha de nacimiento, rol). • Los datos se guardan correctamente en la base de datos. • Se manejan adecuadamente los errores si faltan datos obligatorios.
Pruebas realizadas	
<ol style="list-style-type: none"> 1. Datos válidos: completar todos los campos correctamente y verificar que el perfil del bebé se guarda. 2. Campos obligatorios vacíos: dejar algunos campos obligatorios vacíos y confirmar que se muestra un mensaje de error. 	

Caso de uso:	Registrar actividad (lactancia materna, pañal, sueño, biberón y hitos de desarrollo)
Objetivo probado	Validar que el sistema permite registrar actividades del bebé correctamente.
Requisitos probados	<ul style="list-style-type: none"> • El sistema permite seleccionar el tipo de actividad y solicita los datos necesarios. • Los datos se guardan en la base de datos. • Los errores se manejan adecuadamente si faltan datos obligatorios.

Pruebas realizadas
<p>1. Datos válidos para cada actividad: completar correctamente los datos para cada tipo de actividad y verificar su registro.</p> <p>2. Datos incompletos: dejar campos obligatorios vacíos para actividades específicas y confirmar que el sistema muestra un error.</p>

Caso de uso: Visualizar historial de actividades registradas	
Objetivo probado	Comprobar que el sistema muestra correctamente el historial de actividades registradas de un bebé.
Requisitos probados	<ul style="list-style-type: none"> • El sistema permite seleccionar un perfil de bebé y consultar su historial. • Las actividades registradas se presentan en formato claro y accesible. • Se manejan adecuadamente los casos donde no hay actividades registradas.
Pruebas realizadas	
<p>1. Historial con datos: consultar el historial de un bebé con actividades registradas y verificar que se muestran correctamente.</p> <p>2. Historial sin datos: consultar el historial de un bebé sin actividades registradas y confirmar que no aparecen.</p>	

Caso de uso: Eliminar actividades registradas	
Objetivo probado	Verificar que el sistema permite al cuidador eliminar actividades registradas de un bebé y actualiza correctamente la vista de datos.
Requisitos probados	<ul style="list-style-type: none"> • El sistema debe listar las actividades registradas del bebé seleccionado. • El sistema debe permitir eliminar un registro específico.

	<ul style="list-style-type: none">• El sistema debe actualizar la vista automáticamente tras la eliminación de un registro.
Pruebas realizadas	
1. Dato seleccionado: seleccionar un dato y pulsar Eliminar, se elimina el dato de la base de datos.	2. Dato sin seleccionar: pulsar Eliminar sin seleccionar ningún dato, aparece el mensaje de error.

9. Conclusiones

A lo largo del proyecto, hemos trabajado para cumplir con el objetivo general que se ha planteado, "Facilitar a las familias el registro y el seguimiento del cuidado diario y los hitos importantes en el desarrollo de sus bebés durante los primeros años de vida".

Para ello se han llevado a cabo los diferentes objetivos específicos, consiguiendo completar todos, a excepción de la modificación de los datos guardados. Este ítem no se ha podido llevar a cabo debido a la limitación de tiempo y formación que tenemos.

Sin embargo, no es algo crítico, ya que se pueden borrar los datos y volver a registrarlos de forma correcta.

Gracias a los objetivos que sí se han completado, hemos conseguido crear una aplicación en Android Studio que permite a las familias registrar actividades básicas como cambio de pañales, alimentación y otros datos muy importantes en la primera etapa de la infancia.

También hay un historial para que los cuidadores vean y eliminen datos, lo que resulta útil para un seguimiento detallado.

Además, se ha desarrollado un entorno intuitivo y fácil de usar para los cuidadores, permitiendo que registren los datos de forma cómoda mientras cuidan de sus bebés. La seguridad de los datos ha sido otro pilar en la realización de la aplicación, garantizando el acceso privado de los cuidadores.

Durante el proceso de realización del proyecto, nos hemos dado cuenta de la falta de tiempo para poder llevarlos a cabo. Hemos tenido que invertir mucho tiempo diario para poder entregar el proyecto a tiempo, cumpliendo con todos los requisitos y documentación.

Además, la formación recibida ha sido muy escasa. Hemos tenido que suplir esta falta de formación inicial con el visionado de videotutoriales en YouTube, cursos externos sobre Android Studio, Firebase y los lenguajes utilizados, como los de pildorasinformáticas o mouredev, y la consulta en diferentes webs sobre programación, como stackoverflow.

Aunque satisfactoria en términos de aprendizaje y autoconocimiento, la experiencia fue intensa y por momentos abrumadora. Terminamos el proyecto con la sensación de haber aprendido en dos meses en solitario lo que tendríamos que haber aprendido a lo largo del ciclo completo.

En definitiva, el proyecto nos permitió alcanzar nuestros principales objetivos: proporcionar a las familias herramientas útiles y experiencias de desarrollo significativas. Aunque se han identificado áreas de mejora, particularmente en términos de funcionalidad adicional y mejoras en la interfaz de usuario, la aplicación en su estado actual cumple con los puntos descritos en los objetivos originales.

10. Vías futuras

En este apartado vamos a ver los objetivos que no se han podido cumplir y se realizarán en un futuro, además de mostrar mejoras futuras para la aplicación.

Durante el desarrollo de la aplicación se plantearon varios objetivos que, debido a limitaciones de tiempo y alcance, uno de ellos no pudo ser implementado en la versión actual.

El objetivo que no pudimos llevar a cabo al completo es "Crear un historial accesible donde las familias puedan modificar, eliminar y consultar el seguimiento de las actividades".

En este caso no se pudo implantar la modificación de los datos guardados. No supone un gran problema, ya que se puede eliminar el dato y volver a registrarlo de forma correcta.

A continuación, se detallan las mejoras propuestas, las cuales podrían enriquecer el proyecto en futuras versiones y permitir su evolución en el tiempo.

Compartir el perfil del bebé con más cuidadores: una opción para que varios cuidadores puedan incluir datos en el mismo bebé. Esta función permite a los cuidadores tener un seguimiento más completo y que los datos estén actualizados. Así, si uno de los cuidadores no puede apuntar algún dato, el otro cuidador lo apuntará y estarán actualizados.

Agregar una curva de crecimiento: con esta función se podría registrar la circunferencia de la cabeza, el peso y la altura del bebé, consiguiendo así los datos para crear una curva de crecimiento. En este caso habría que contar con la ayuda de un pediatra para ver los valores que hay en cada percentil. Esta función ayudará a los cuidadores a comparar el progreso de su hijo con las pautas de crecimiento estándar, brindando una vista más detallada de su desarrollo.

Descargar informes en PDF: gracias a esta función los cuidadores podrán exportar los registros en un archivo PDF.

Visualizar gráficos: con esta función se integran gráficos de los datos registrados para mostrar visualmente los cambios en diferentes aspectos como el tiempo de sueño, el número de comidas o la cantidad de tomas. Las representaciones gráficas facilitarán la interpretación de los datos y ayudarán a los cuidadores a identificar patrones y posibles cambios en la rutina diaria del bebé.

Versión para la escuela infantil: se crearía una versión adecuada para su uso en escuelas infantiles, permitiendo a los cuidadores registrar información sobre varios bebés. Crear informes detallados para varios bebés. Esto permitirá a las maestras y familias mantener registros completos y compartir la información de sus bebés.

Establecer categoría de actividad: esta función ofrece una pantalla de configuración para seleccionar categorías de registros que el cuidador puede activar o desactivar según sea necesario. Esta mejora hará que la interfaz de usuario sea más clara y más adaptada a las prioridades de los cuidadores, facilitando su uso y evitando el tener más funciones de las necesarias en la pantalla.

Crear versión iOS: desarrollar una versión de la aplicación para el sistema operativo iOS permitiría llegar a un público más amplio, asegurando que familias con dispositivos de Apple también puedan beneficiarse de las funciones de la aplicación. Esta expansión aumentaría la presencia de la aplicación en el mercado y fomentaría su crecimiento.

Monetizar la aplicación: estudiar opciones de monetización que permitan mantener y mejorar el proyecto. Entre las alternativas se podrían considerar versiones premium con funcionalidades avanzadas, suscripciones para el almacenamiento de datos, la descarga de informes personalizados o una versión sin publicidad para los usuarios que deseen una experiencia sin interrupciones.

Notificaciones inteligentes y recordatorios: añadir recordatorios automáticos para actividades recurrentes, como cambios de pañal, alimentación o sueño, basados en los patrones de cada bebé junto con el consejo de un pediatra. También se podrían incluir notificaciones cuando se detecten patrones inusuales en los registros, ayudando a los cuidadores a llevar un mejor seguimiento de la salud y el bienestar del bebé.

Integración con dispositivos inteligentes y wearables: permitir la sincronización con dispositivos inteligentes como *Alexa* o *Siri*, de forma que al pedirles que inicien la toma de pecho, se guarde en la aplicación. Además, se pueden incluir funciones en wearables (como monitores de sueño o pulseras de actividad) para registrar automáticamente aspectos como el sueño, las tomas u otros, ofreciendo una visión aún más completa de la salud del bebé.

Opción multilingüe: ampliar la selección de idiomas para que la aplicación pueda ser usada en diversas regiones y por personas que hablen distintos idiomas, facilitando su acceso a una audiencia global. Dentro de esta nueva función habría que tener en cuenta también la cultura, no solo el idioma.

Estas vías de desarrollo ofrecen importantes oportunidades para mejorar el proyecto, lo que permite a la aplicación adaptarse a las circunstancias cambiantes y ser más competitiva.

11. Bibliografía/Webgrafía

Beltapis (2024). *Baby Daybook Lactancia materna* (5.20.7) [Aplicación móvil]. Google Play.

<https://play.google.com/store/apps/details?id=com.drillyapps.babydaybook&hl=es>

Canva. (2024). *Canva*. <https://www.canva.com/>

Códigos de programación - MR. (18 de marzo de 2021). *Mostrar registros en lista dinámica | CRUD Android y SQLite* [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=UfZJWqm2A5w&t=517s>

Codigofacilito. (11 de diciembre de 2017). *Tabla Dinámica Android* [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=rSafkhA6TZA>

Curso Kotlin. (2024). *Curso programación Kotlin Android*.

<https://cursokotlin.com/curso-programacion-kotlin-android/>

Demapps LLC (2024). *Lactancia y Baby tracker Erby* (5.2.0) [Aplicación móvil]. Google Play.

<https://play.google.com/store/apps/details?id=com.whisperarts.erby&hl=es>

Developer Android. (2024). *Guía para desarrollar interfaces en Android*.

<https://developer.android.com/develop/ui?hl=es-419>

El arte de la programación. (s.f.). *CRUD en Firebase y Android Studio*. [Lista de reproducción]. YouTube.

https://www.youtube.com/watch?v=tIs_k-Ov6BY&list=PL2LFsAM2rdnxv8bLBZrMtd_f3fsqLzH7

Firebase. (2024). *Firebase*. <https://firebase.google.com/?hl=es-419>

iLoveIMG. (2024). *Herramientas de edición de imágenes online*.

<https://www.iloveimg.com/es>

LactApp Women Health (2024). *LactApp: Embarazo y Lactancia* (7.3.4) [Aplicación móvil].

Google Play.

<https://play.google.com/store/apps/details?id=com.whisperarts.erby&hl=es>

Lucid Software. (2024). *Lucidchart*. <https://lucid.app/>

Monday.com. (2024). *Monday*. <https://monday.com/>

MoureDev by Brais Mouré. (23 de marzo de 2020). *KOTLIN: Curso ANDROID desde CERO*

para PRINCIPIANTES [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=ebQphhLpJG0&t=6789s>

MoureDev by Brais Mouré. (6 de septiembre de 2020). *FIRESTORE Android BASE de*

DATOS con Firebase [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=t5yyC1XfQrs>

MoureDev by Brais Mouré. (s.f.). *Curso programación Android con Android Studio* [Lista de

reproducción]. YouTube.

https://www.youtube.com/playlist?list=PLNdFk2_brsRdYF0FXDtSaGvluzBNHRbNe

OpenAI. (2024). *ChatGPT Logo Creator*. <https://chatgpt.com/g-gFt1ghYJI-logo-creator>

Píldoras Informáticas. (2024). *Curso de Android con Kotlin*.

<https://www.pildorasinformaticas.es/course/curso-android-kotlin/>

Píldoras Informáticas. (s.f.). *Android con Kotlin*. [Lista de reproducción]. YouTube.

https://www.youtube.com/playlist?list=PLU8oAIHdN5BkdfBPpNv_lVCJxJgE87cr0

Pixabay. (2024). *Pixabay - Banco de imágenes gratuito*. <https://pixabay.com/es/>

Programador Novato. (1 de marzo de 2022). *Llenar una tabla de forma dinámica*

(*TableLayout*) *Android Studio* [Vídeo]. YouTube.

https://www.youtube.com/watch?v=N_s8iR0eY8g

Stack Overflow. (2024). *Comunidad de preguntas y respuestas*.

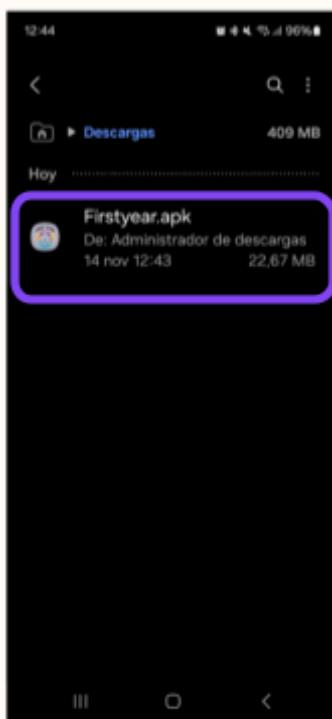
<https://es.stackoverflow.com/>

Surfaweb. (22 de julio de 2022). *Como crear un tableLayout dinámico en Android Studio con*

Kotlin [Vídeo]. YouTube. <https://www.youtube.com/watch?v=oUmIGN7dhtw>

12. Anexos

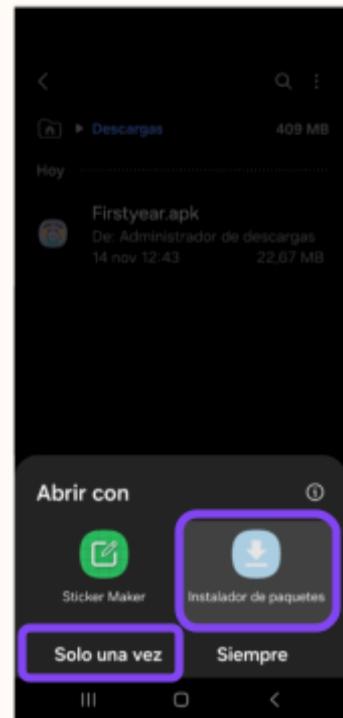
12.1. Manual de instalación



Debido a que aún estamos en la primera fase y no se ha subido la aplicación a ninguna tienda, es imprescindible tener el archivo .apk para poder instalar la aplicación.

Descargar el archivo .apk

Buscar en la carpeta **Descargas** el archivo .apk o en la carpeta que tengas configurada.



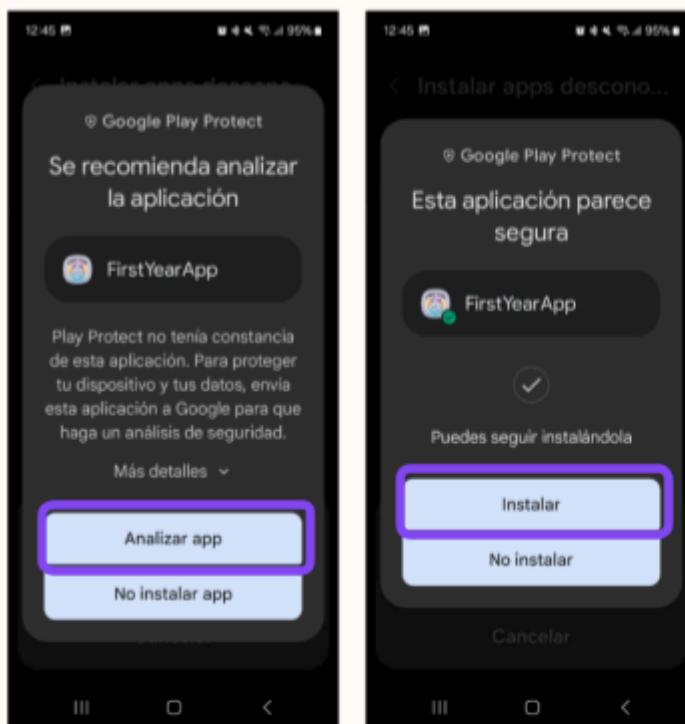
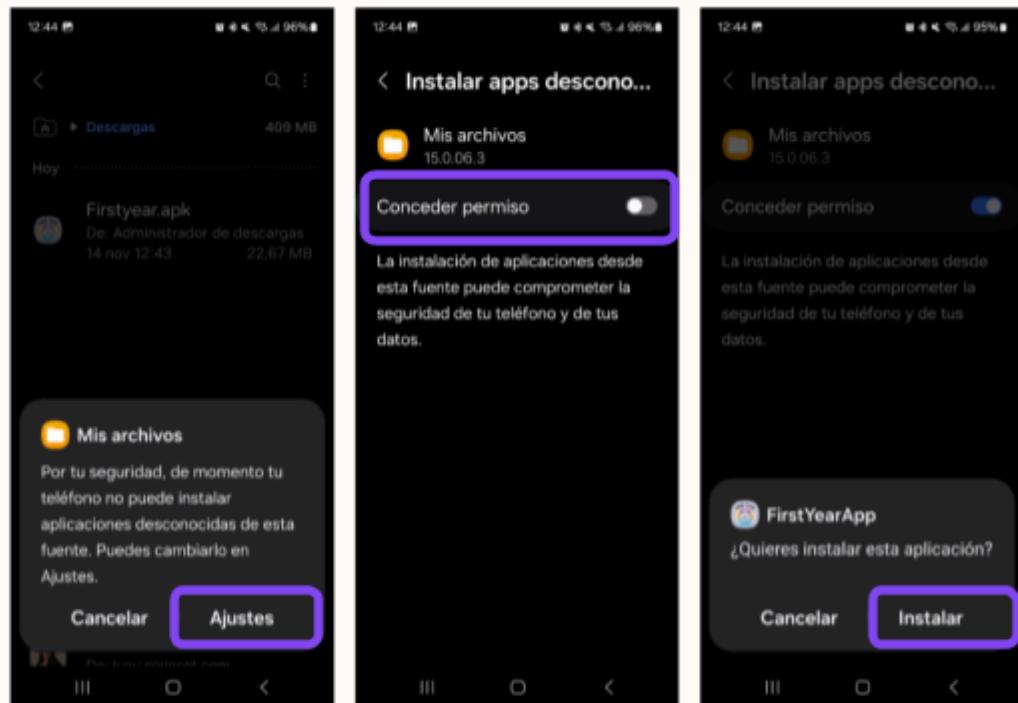
Instalar el archivo .apk

Al pulsar sobre el archivo .apk aparece una aviso para seleccionar con qué abrir el archivo, pulsamos en la opción marcada en la imagen o la que más se le parezca, esto puede cambiar según el dispositivo.

Conceder permisos

Al intentar instalar el archivo .apk el dispositivo tiene un sistema de seguridad por el que no te dejará instalarlo a no ser que concedas el permiso para instalar apps de orígenes desconocidos.

Pulsa en las opciones marcadas en cada captura.



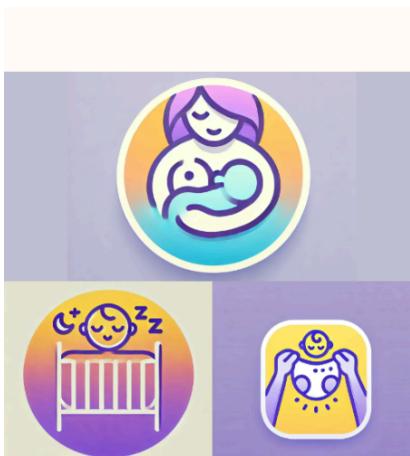
Analizar la aplicación

Los dispositivos Android tienen otro sistema de seguridad por el que Google PlayProtect analiza la aplicación antes de instalarla.

Pulsa en las opciones marcadas en cada captura.

**¡Ya puedes disfrutar
de la aplicación!**

12.2. Manual de usuario



FirstYear App

Manual de usuario

Registro de usuarios

Completa los datos y pulsa **Crear cuenta**.

Si ya tienes cuenta, pulsa **Ya tengo cuenta**.

Registro de usuarios

Nombre	<input type="text" value="Ingrese su nombre"/>
Correo electrónico	<input type="text" value="Ingrese su correo electrónico"/>
Contraseña	<input type="text" value="Ingrese su contraseña"/>
Teléfono (opcional)	<input type="text" value="Ingrese su teléfono"/>

Crear Cuenta

Ya tengo cuenta

Iniciar sesión

Correo electrónico

Contraseña

Acceder

No tengo cuenta

Inicio de sesión

Si ya estás registrado completa los datos y pulsa **Acceder**.

Si no tienes cuenta, pulsa **No tengo cuenta**.

Área de bebés

Aquí puedes seleccionar el bebé con el que quieras registrar los datos.

Para ello antes tienes que haberlo añadido, puedes hacerlo en la opción **Añadir bebé**.

Si quieres cerrar la sesión que has iniciado, pulsa el botón **Cerrar sesión**.

Área de bebés



Añadir bebé

←

Femenino **Masculino**

Nombre
Ingrese el nombre del bebé

Fecha de nacimiento
Selecciona la fecha de nacimiento

Rol
Madre

Guardar

Añadir bebé

Registra un bebé completando los datos y pulsando **Guardar**.

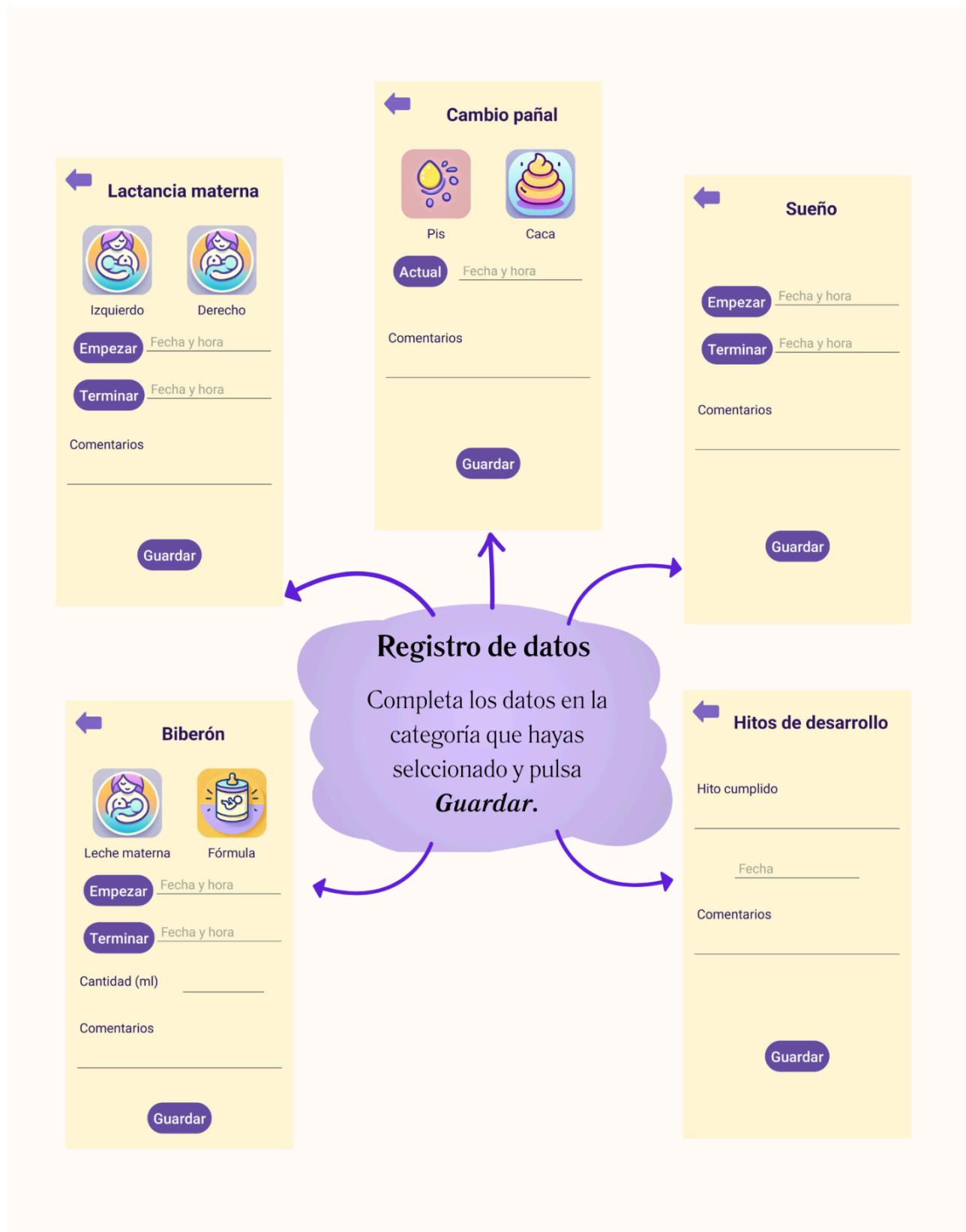
Categorías de registro

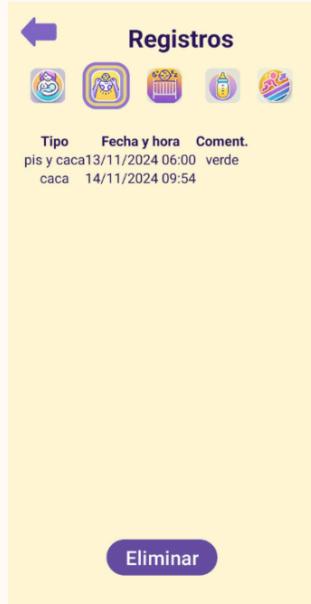
En esta pantalla puedes seleccionar en qué categoría añadir datos pulsando sobre cada una de ellas.

Puedes ver los registros de cada categoría pulsando en **Registros**.

Si quieres borrar el bebé con el que has entrado pulsa en el ícono de la esquina superior derecha.







Ver registros

Seleccionar la categoría de la que se quieren ver los datos registrados.



Borrar registros

Seleccionar la línea del dato que se quiere borrar y pulsar el botón **Eliminar**.



12.3. Imágenes

Diagrama de Gantt estimado y real



Diagrama entidad-relación

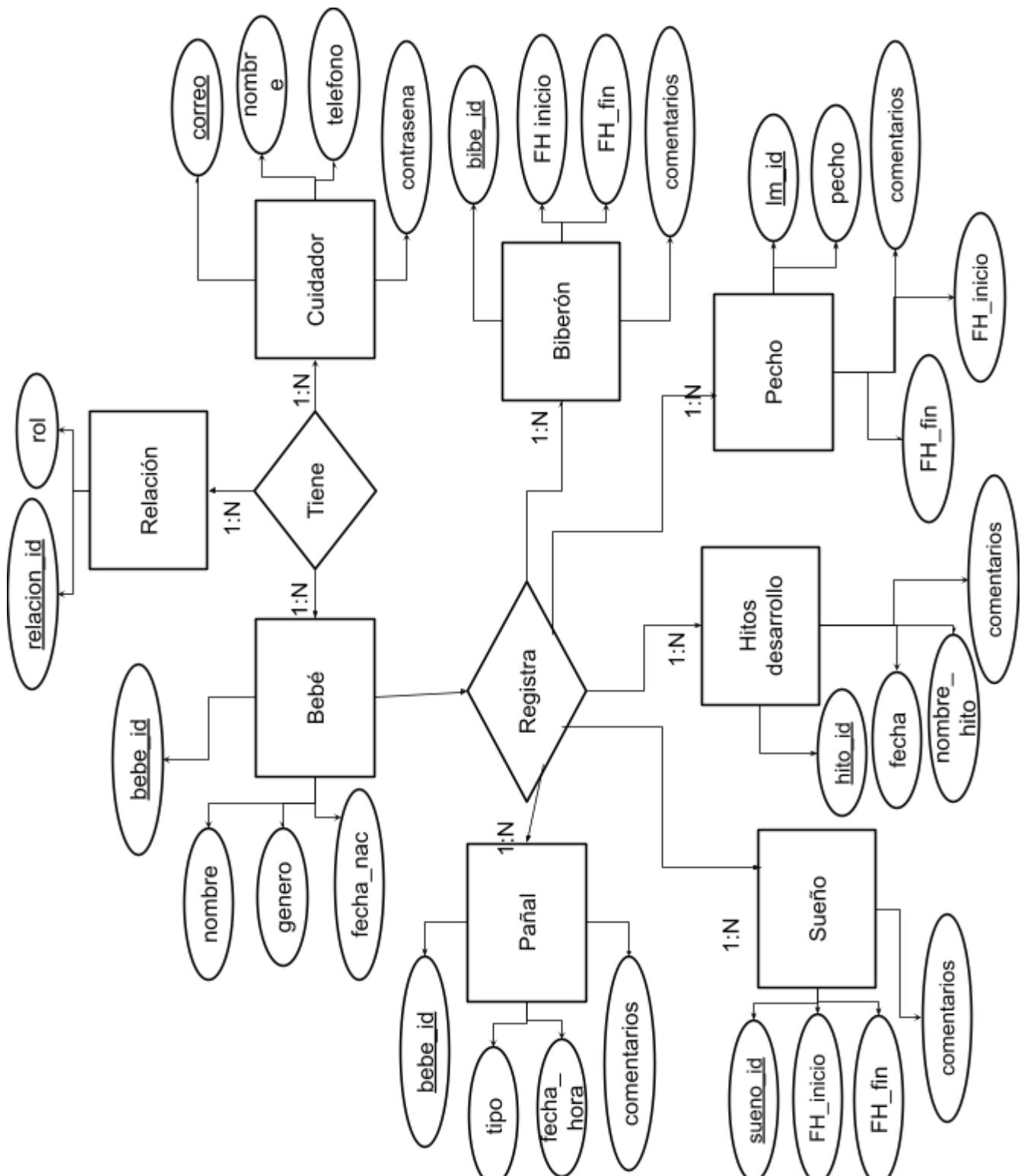


Diagrama de clases

