



SISTEMA DE CÁLCULO DE DOSIS DE MEDICAMENTOS

**Solución digital para la dosificación de medicamentos
estandarizados**

Nombre del alumno o de la alumna: Sara Hevia López

Curso académico: 2 DAM

ÍNDICE PAGINADO

1. JUSTIFICACIÓN DEL PROYECTO	2
2. INTRODUCCIÓN	3
3. OBJETIVOS	4

4. DESARROLLO	4
5. DESARROLLO DE LA APP	7
6. PASOS SEGUIDOS EN EL DESARROLLO	10
7. CONCLUSIONES	11
8. LÍNEAS DE INVESTIGACIÓN FUTURAS	12
9. BIBLIOGRAFÍA	13
10. ANEXOS	13
11. OTROS PUNTOS	14



1. JUSTIFICACIÓN DEL PROYECTO

¿Por qué este proyecto?

Este proyecto nace de la necesidad de crear una herramienta que ayude a personalizar las dosis, facilitando la autogestión responsable de la medicación y reduciendo riesgos en la automedicación.

- La mayoría de los medicamentos se recetan con dosis estándar sin personalización.
- Personas con menor peso pueden estar **tomando más de lo necesario**, aumentando el riesgo de efectos secundarios.
- Personas con mayor peso pueden **estar infra dosificadas**, reduciendo la eficacia del medicamento.
- No todas las personas pueden consultar constantemente a un médico para ajustar la dosis.
- Facilita la **autogestión responsable de la medicación**.

Beneficios:

- Cálculo rápido y preciso de la dosis.
- Prevención de errores en la automedicación.
- Mayor control sobre la propia salud.

- Fácil de usar para cualquier persona.

2. INTRODUCCIÓN

El ajuste de dosis según el peso corporal tiene un fuerte respaldo científico. Muchos estudios médicos subrayan que las dosis estandarizadas para un peso de 70 kg no son adecuadas para pacientes con un peso significativamente diferente. Por ejemplo, la dosificación para una

persona muy delgada será distinta a la de un adulto promedio de 70 kg, lo que puede resultar en efectos adversos o en la ineptitud del tratamiento.



Solución propuesta:

Desarrollar una aplicación en la que el usuario introduzca:

1. La dosis indicada en el prospecto.
2. El peso estándar para el que está calculada.
3. Su peso real.

La aplicación calculará la dosis exacta adaptada al usuario.

3. OBJETIVOS

A. OBJETIVO GENERAL

Desarrollar una aplicación móvil que permita calcular la dosis exacta de medicamentos de manera personalizada según el peso del usuario para mayor seguridad.

B. OBJETIVOS ESPECÍFICOS

- Implementar un algoritmo que realice el cálculo de dosis basado en peso.
- Crear una interfaz intuitiva para facilitar el uso de la aplicación.
- Garantizar persistencia de datos.
- Permitir guardar medicamentos para consultas futuras.
- Crear un chat funcional para consultar con un enfermero o médico en directo.
- Ofrecer manejo fluido de la aplicación.
- Mantener una protección de los datos personales: nunca se guarda la información relativa al paciente.

4. DESARROLLO

1. FUNDAMENTACIÓN TEÓRICA:

El cálculo de dosis de medicamentos es un aspecto fundamental en la seguridad y eficacia de los tratamientos médicos. En la mayoría de los casos, las dosis indicadas en los prospectos están diseñadas para un peso estándar de 70 kg, lo que puede generar errores en la administración, especialmente en personas con pesos significativamente distintos.

Este proyecto propone el desarrollo de una aplicación móvil que permita calcular la dosis exacta basándose en la información proporcionada en el prospecto y el peso real del usuario.



2. PROCEDIMIENTOS:

La aplicación funciona siguiendo estos pasos:

- (i) El usuario introduce el nombre del medicamento, dosis recomendada en el prospecto.
- (ii) Si no incluye datos, el cálculo se realiza por defecto con un peso base de 70 kg.
- (iii) Registra su propio peso.
- (iv) La aplicación calcula automáticamente la dosis ajustada utilizando las siguientes fórmulas:

$$Dosis\ estimada\ por\ kg = \frac{dosis.mínima}{peso.promedio}$$

$$Dosis\ ajustada = dosis\ estimada\ por\ kg \cdot peso\ del\ paciente$$

- (v) Se verifica que la edad no sea inferior a 15 años ni superior a 75.

De otro modo, no se realiza el cálculo porque no sería seguro calcular dosis personalizadas para pacientes de edad pediátrica o geriátrica.

- (vi) Se muestran las dosis ajustadas en la pantalla:
 - Dosis mínima recomendada
 - Dosis máxima recomendada
 - Dosis máxima diaria
- (vii) Se puede acceder a un chat que funciona mediante *websocket*.

3. HIPÓTESIS INICIAL:



"El ajuste de dosis basado en peso mejora la precisión del tratamiento y reduce el riesgo de efectos secundarios o ineficacia del medicamento."

Para comprobar esta hipótesis:

- ✓ Comparamos dosis estándar con la dosis personalizada.
- ✓ Evaluamos cuánto se desvía la dosis recomendada para distintos pesos.

Ejemplo de caso práctico con **Ibuprofeno 500 mg**:

- Dosis en prospecto: **500 mg**
- Peso estándar: **70 kg**
- Peso del usuario: **50 kg**

$$Dosis\ estimada\ por\ kg = \frac{dosis.\acute{m}inima}{peso\ promedio}$$

$$Dosis\ estimada\ por\ kg = \frac{500\ mg}{70\ kg} = 7.14\ mg\ por\ kg$$

$$Dosis\ ajustada = dosis\ estimada \cdot peso\ paciente$$

$$Dosis\ ajustada = 7.14\ mg \cdot 50\ kg = 357\ mg$$

En este caso, la aplicación recomendaría **tomar 357 mg en lugar de 500 mg**, evitando una "sobredosis"¹ innecesaria.

¹ Entiéndase por sobredosis, ingesta o administración de una cantidad de una sustancia, generalmente un fármaco, por encima de los niveles recomendados o seguros, lo que puede llevar a efectos adversos.

4. MATERIALES Y MÉTODOS:



Para desarrollar la aplicación, se han consultado fuentes médicas y farmacéuticas:

- Normativas de dosificación en la **Agencia Española de Medicamentos y Productos Sanitarios (AEMPS)**.
- Manuales médicos y guías clínicas sobre ajuste de dosis en función del peso.

5. RESULTADOS Y ANÁLISIS

- Reducción del margen de error en la automedicación:
La aplicación ha demostrado ser útil al calcular dosis personalizadas basadas en el peso del usuario, reduciendo el riesgo de sobredosificación o infra dosificación.
- Mayor precisión en la toma de medicamentos basada en peso.
La personalización de las dosis asegura que los usuarios reciban la cantidad exacta de medicamento, optimizando su eficacia.
- Facilidad de uso sin necesidad de conocimientos médicos avanzados.
La interfaz intuitiva permite que cualquier persona, incluso sin conocimientos médicos avanzados, pueda usar la aplicación sin dificultades.
- Seguridad mejorada: Con las validaciones de edad (para no calcular dosis para pacientes menores de 15 o mayores de 75) y el uso de algoritmos precisos, la aplicación asegura la seguridad del paciente, minimizando riesgos de dosificación errónea.
- Aumento de la seguridad en la administración de medicamentos de venta libre.

5. DESARROLLO DE LA APP

TECNOLOGÍAS UTILIZADAS:

1. IDES

1.1. Intelli J

Utilizado principalmente para el desarrollo backend con **Java**. Este IDE es altamente eficiente para trabajar con frameworks como **Spring Boot**,



proporcionando autocompletado, depuración avanzada, integración con bases de datos y herramientas de refactorización.



INSTITUTO
NEBRIJA

Formación
Profesional

1.2. Visual Studio Code

Usado para el desarrollo frontend con **React** y **JavaScript**.

Este editor ligero permite trabajar con múltiples extensiones que facilitan la escritura de código y depuración.



2. LENGUAJES

2.1. Java + Spring Boot

Se ha utilizado **Java**, un lenguaje robusto y ampliamente utilizado, junto con **Spring Boot**, un framework de desarrollo que permite desarrollar aplicaciones autónomas de manera rápida y eficiente, con características como seguridad, persistencia y manejo de dependencias integradas.



2.2. React + JavaScript

Para la construcción de la interfaz de usuario, se ha utilizado **React**, una librería de JavaScript para crear interfaces de usuario interactivas y reactivas. React permite construir componentes reutilizables, facilitando el desarrollo de

interfaces de usuario dinámicas. **JavaScript** se utiliza para la lógica de las aplicaciones en el lado del cliente.



2.3. Bash Scripting

Para la gestión de la infraestructura del proyecto. A través de scripts bash, se facilitan tareas repetitivas como la configuración del entorno de desarrollo, despliegue de aplicaciones y manejo de contenedores Docker.

```
#!/bin/bash
```

2.4. Dockerfile

La creación de imágenes a través de un **Dockerfile** garantiza que la aplicación sea desplegada en cualquier entorno sin conflictos de configuración.

```
1 # Fase de construcción
2 FROM maven:3.9-eclipse-temurin-23 AS T build
3 WORKDIR /app
4 COPY pom.xml .
5 RUN mvn dependency:go-offline -B
6 COPY src ./src
7 RUN mvn clean package -DskipTests
8
9 # Fase de ejecución
10 FROM eclipse-temurin:23-jre-alpine
11 EXPOSE 8080
12 COPY --from=build /app/target/*.jar app.jar
13 ENTRYPOINT ["java", "-jar", "app.jar"]
```

2.5. SQL

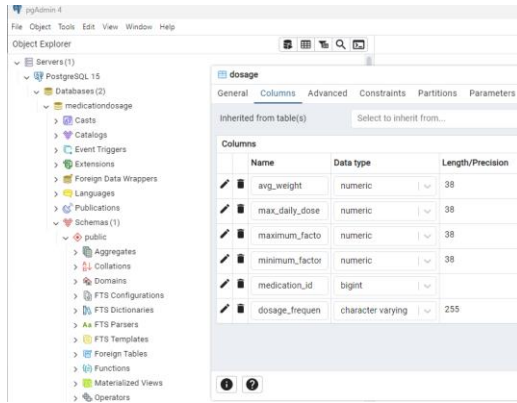
SQL es el lenguaje utilizado para la interacción con la base de datos, realizando consultas, actualizaciones y manejo de datos de manera eficiente.



3. TECNOLOGÍAS

3.1. PostgreSQL

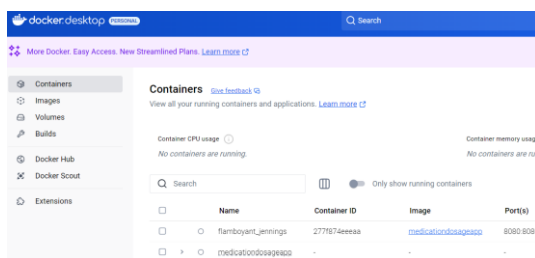
Para la gestión de la base de datos, se ha utilizado **PostgreSQL**, un sistema de gestión de bases de datos relacional robusto y de código abierto.



3.2. Hibernate

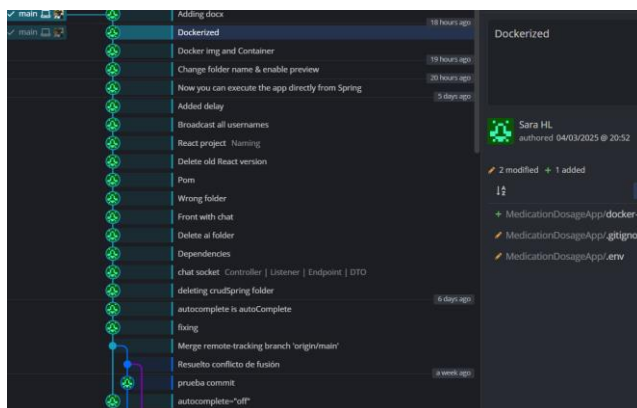
Se ha utilizado **Hibernate** como marco de persistencia para facilitar la interacción entre la base de datos y el código Java. Hibernate es un **ORM** que mapea las clases Java a las tablas de la base de datos, simplificando las operaciones CRUD y gestionando las transacciones de manera eficiente.

3.3. Docker



Docker se ha empleado para “dockerizar” la aplicación, permitiendo crear entornos aislados y consistentes para el desarrollo, prueba y despliegue.

3.4. GitHub+ GitKraken



Para el control de versiones, se ha utilizado **GitHub**, una plataforma basada en Git para gestionar el código fuente, crear ramas... **GitKraken**, una herramienta visual para trabajar con Git, ha sido útil para gestionar las ramas, hacer merges y visualizar el historial de commits de manera sencilla y eficiente.

6. PASOS SEGUIDOS EN EL DESARROLLO

1. **Definición del problema:** Investigué sobre la falta de personalización en las dosis de medicamentos.
2. **Diseño de las tablas y clases.**
3. **Diseño del algoritmo:** Se estableció la fórmula de cálculo basada en el peso.
4. **Elección de tecnologías**
5. **Desarrollo del backend:** Se crearon las estructuras de datos y funciones de cálculo de dosis.
6. **Implementación de la interfaz:** Se diseñaron pantallas intuitivas y fáciles de usar.
7. **Pruebas y validación:**



- Se realizaron test con distintos escenarios para comprobar la precisión del cálculo.
 - Se añaden validaciones para no calcular con edades menores de 15 y mayores de 75.
8. Se añaden websockets para el chat.
- **Chat en tiempo real** usando **React** con **STOMP sobre WebSockets** en el frontend y un canal de comunicación interna con **BroadcastChannel** para sincronizar usuarios entre pestañas. [OBJ]
9. Se corrige la comunicación entre sockets.
10. **Optimización y mejoras:** Se corrigieron errores
11. Dockerizar aplicación:
- Se generan archivos estáticos del Front y se introducen en el backend (carpeta static) para que la API y el frontend se ejecuten en el mismo servidor.
 - **Docker** para contenerizar el frontend como el backend en un solo servicio.
 - Creación del `docker-compose.yml`
Se definen 2 servicios (porque el front y back ya se han juntado)
Back por una parte y base de datos
12. Documentación.

7. CONCLUSIONES

El desarrollo de esta aplicación ha permitido abordar un problema cotidiano en la administración de medicamentos: la falta de personalización en la dosificación según el peso del paciente.

Conclusiones principales:

- ✓ Mayor control y autonomía en la administración de medicamentos.
- ✓ Aumento de la conciencia sobre la importancia de la dosificación correcta.
- ✓ Base sólida para futuras mejoras y ampliaciones del sistema.
- ✓ Reducción de errores: La aplicación ayuda a evitar tanto la sobredosis como la infra dosificación, mejorando la seguridad en la automedicación.
- ✓ Limitaciones detectadas: Actualmente, la aplicación no considera factores como patologías previas, interacciones medicamentosas o ajustes en pacientes con insuficiencia renal o hepática.

8. LÍNEAS DE INVESTIGACIÓN FUTURAS

Mejoras en el cálculo de dosis:

- Implementar ajustes para patologías específicas como insuficiencia renal o hepática.
- Añadir compatibilidad con medicamentos en forma líquida o intravenosa.
- Incluir ajustes de dosis para pediatría y geriatría bajo supervisión médica.
- Ofrecer recomendaciones basadas en inteligencia artificial según historial del usuario.

Integraciones con sistemas externos:

- Conectar la aplicación con bases de datos médicas oficiales para obtener información actualizada.
- Sincronización y compatibilidad con Google Fit o Apple Health para obtener datos de peso automáticamente.
- Implementar un escáner de códigos de barras para detectar la dosis del medicamento directamente del envase.
- Sincronización con aplicaciones de historial médico electrónico para médicos y pacientes.
- Implementar estándares médico-informáticos como FHIR (Fast Healthcare Interoperability Resources), que facilitan la interoperabilidad de los sistemas de salud, orientando la aplicación al ámbito profesional.

9. BIBLIOGRAFÍA

Normativas y fuentes médicas:

- Agencia Española de Medicamentos y Productos Sanitarios (AEMPS):
<https://www.aemps.gob.es>
- European Medicines Agency (EMA):
<https://www.ema.europa.eu>
- World Health Organization (WHO) - Guías de dosificación:
<https://www.who.int>

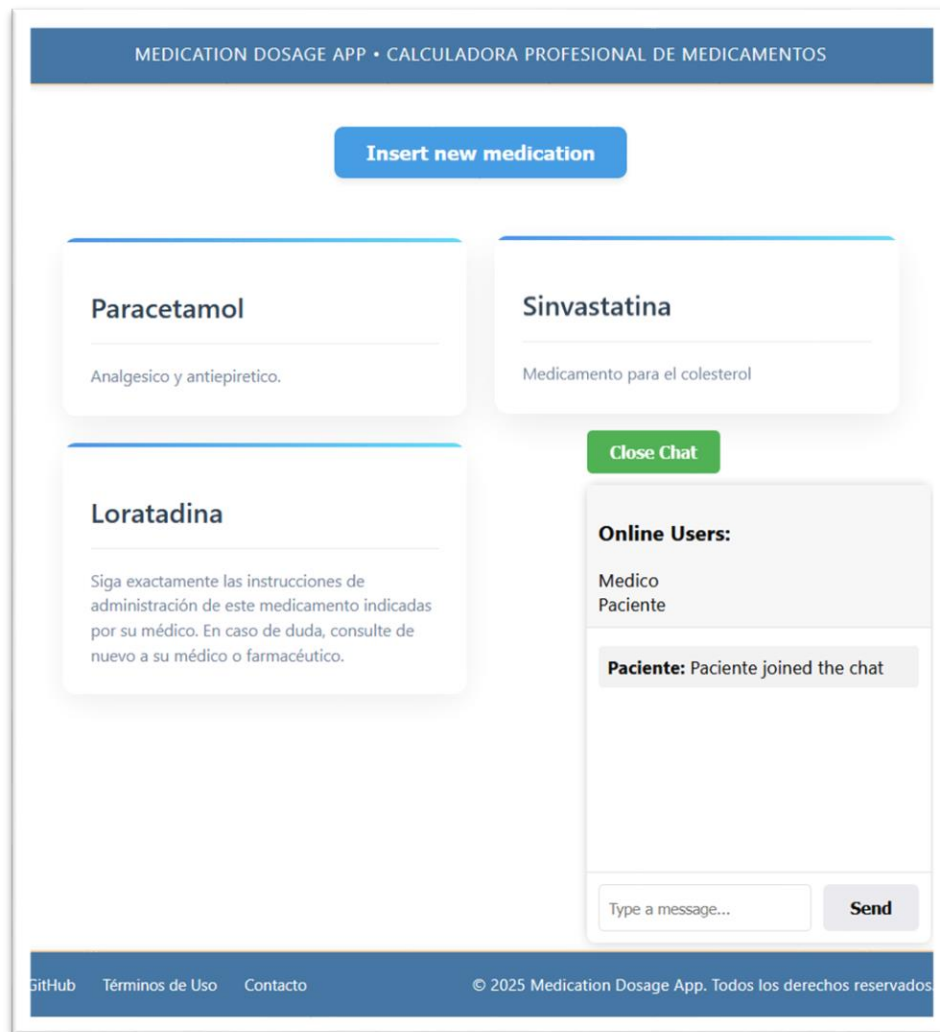
Documentación oficial de tecnologías usadas:

- Kotlin y Android Development:
<https://developer.android.com/kotlin>
- SQLite para almacenamiento de datos:
<https://www.sqlite.org/docs.html>
- Spring Boot para backend:
<https://spring.io/projects/spring-boot>
- React para interfaz de usuario:
<https://react.dev/>
- GitHub para control de versiones:
<https://docs.github.com/>

10. ANEXOS

Enlace a GitHub

<https://github.com/Shevial/Medication-Dosage-App>



11. OTROS PUNTOS

APORTACIONES PERSONALES:

A lo largo del proceso, he adquirido una visión más holística y estructurada del ciclo de vida de una aplicación, lo que me ha permitido anticipar posibles dificultades y evitarlas mediante una planificación más detallada y una mayor comprensión del código base.

Además, el uso de la aplicación ha sido esencial para mejorar mi capacidad para comprender la variación de las dosis en función del peso del paciente, para ser más consciente de la importancia de la seguridad de la administración de medicamentos de venta libre.

Lo que más me ha aportado:

- Mejor manejo de IntelliJ.
- Aplicación de Docker.
- Comprensión de SpringBoot

- Comprensión sobre la importancia de seguridad en aplicaciones de salud.
- Desarrollo de algoritmos de cálculo de dosis con precisión médica.



RETOS PROFESIONALES:

El universo médico y sanitario es un campo extremadamente delicado, donde la precisión y la fiabilidad de las herramientas tecnológicas son esenciales para el bienestar de los pacientes. Dado que la salud depende de sistemas que gestionan datos sensibles, es fundamental garantizar la seguridad, la privacidad y el correcto funcionamiento de las aplicaciones.

RETOS PERSONALES:

El impacto positivo de esta aplicación ha incrementado mi interés en desarrollar soluciones tecnológicas que sean útiles y seguras en el ámbito médico y sanitario. Esta experiencia ha reforzado mi motivación por contribuir al desarrollo de aplicaciones que no solo mejoren la eficiencia, sino que también impacten positivamente en la vida de los pacientes.

Es esencial cumplir con las normativas locales e internacionales en cuanto a seguridad y privacidad, y estar al tanto de los estándares médico-informáticos como FHIR (Fast Healthcare Interoperability Resources), que facilitan la interoperabilidad de los sistemas de salud.