

## **REFLEXIÓN FINAL**

### **Aprendizajes Obtenidos**

Durante el desarrollo de SmartBite, el equipo experimentó una curva de aprendizaje significativa dentro del manejo de tecnologías y procesos de desarrollo de software que fue más allá de la simple codificación. El hecho de llegar a una versión v3 del proyecto nos enseñó la importancia de la refactorización. Aprendimos a no solo escribir código que funcione, sino a estructurarlo mejor para que fuera escalable, corrigiendo errores de lógica que arrastramos desde las primeras versiones; El uso de GitHub fue fundamental. Aprendimos a manejar conflictos de merge, a organizar las carpetas del proyecto y a coordinar cambios simultáneos sin sobrescribir el trabajo de los compañeros.

### **Temas no abordados o a profundizar**

Aunque el temario fue completo para el alcance del proyecto, identificamos áreas que agregarían mucho valor en futuras ediciones, nos hubiera gustado aprender cómo llevar SmartBite de un entorno local a un servidor real o en la nube (como AWS, Azure o Heroku), para entender el ciclo completo de vida del software pero el desarrollo se enfocó a hacer funcionar las recomendaciones implementando inteligencia artificial; Si bien implementamos funcionalidades de usuario, sería valioso profundizar en buenas prácticas de seguridad como encriptación de contraseñas, tokens, protección contra inyecciones SQL; Quizás el tema de pruebas unitarias o automatizadas se podría abordar con mayor profundidad para garantizar la estabilidad del código antes de la entrega final.

### **Problemas, limitaciones y restricciones**

El proceso no estuvo exento de dificultades que impactaron nuestro ritmo de trabajo, la curva de aprendizaje de las herramientas, al principio configurando del entorno de desarrollo y las dependencias de nos consumió tiempo valioso que hubiéramos preferido dedicar al desarrollo de funcionalidades, también definir el alcance exacto de SmartBite fue un reto. A veces queríamos añadir funcionalidades complejas que, por las restricciones de tiempo y la carga académica de otras materias, tuvimos que simplificar o descartar en la v3.

### **Propuestas de mejora para el curso**

Implica un poco mas de trabajo pero sería ideal tener pequeñas revisiones parciales (sprints) más seguidas antes de la entrega final, para validar que la arquitectura del proyecto va por buen camino y no tener que corregir errores de estructura de código al final.