



ANÁLISIS DE COMPORTAMIENTO DE CÓDIGO

YAGO FERNÁNDEZ LÓPEZ – UO289549

RUBÉN PÉREZ DAFONTE – UO289337

ÍNDICE

- Introducción
- Base del Análisis de Comportamiento de Código
- Deuda Técnica y Hotspots
- Organización de los Equipos
- Familiaridad y Grupos de Conocimiento
- Integración en el Ciclo de Vida del Software
- Ética y Moral
- Regla del Boy Scout
- El Futuro del Análisis de Comportamiento de Código
- Los límites actuales
- Herramientas

INTRODUCCIÓN

¿Qué es análisis de comportamiento de código?

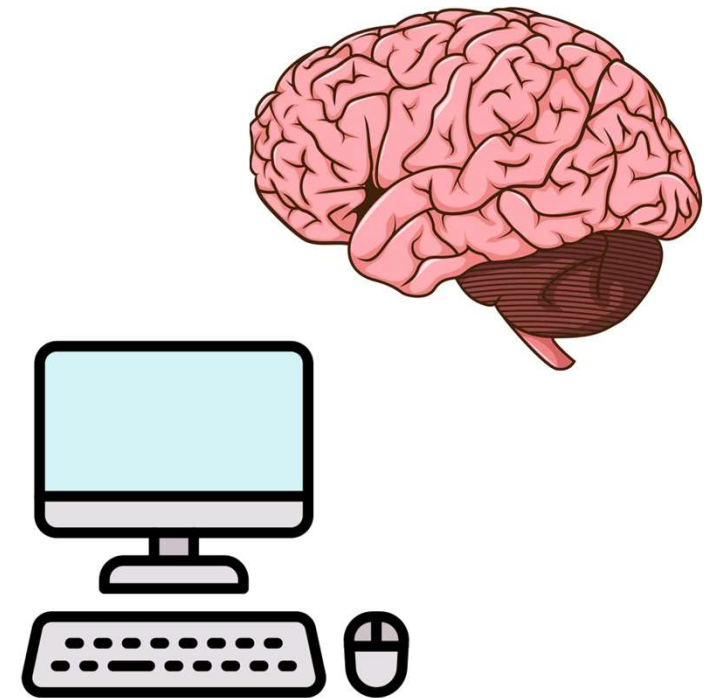
- Adam Thornhill (ingeniería y psicología)
- Análisis de comportamiento de código => Comprender y evaluar comportamiento



BASE DEL ANÁLISIS DEL COMPORTAMIENTO DE CÓDIGO

El análisis de comportamiento de código identifica como interactúan las partes del código

- Permite comprender:
 - La calidad
 - La seguridad
 - El rendimiento
- Permite identificar:
 - Posibles vulnerabilidades
 - Cuellos de botella
 - Deuda técnica y hostspots



DEUDA TÉCNICA Y HOTSPOTS



	DEUDA IMPRUDENTE	DEUDA PRUDENTE
DEUDA DELIBERADA	Nos falta de tiempo para diseñar	Tenemos que priorizar la entrega de software y lidiar con las consecuencias
DEUDA INVOLUNTARIA	¿Qué es un diseño por capas?	Ahora sabemos cómo deberíamos haberlo hecho

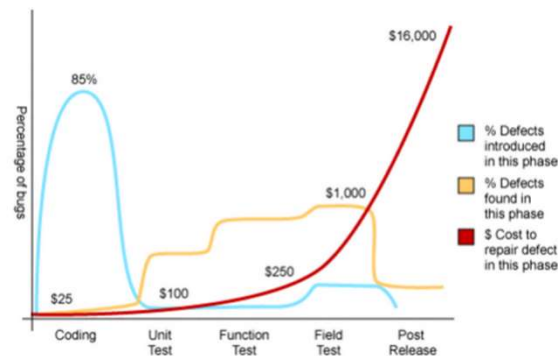
Deuda Técnica

- La deuda técnica son los compromisos adquiridos durante el desarrollo que pueden afectar negativamente al software

Casos de Deuda Técnica:

- Código mal estructurado
- Falta de documentación
- Falta de pruebas
- Diseño deficiente
- ...

DEUDA TÉCNICA Y HOTSPOTS



El 85 % de los defectos, se producen en la **fase de codificación**, y es ahí donde nos costaría menos solucionarlos.

Áreas de código fuente:

- Contienen potenciales problemas
- Son críticas para la calidad, el rendimiento o la seguridad
- Hace falta refactorizarlas

Suelen ser secciones de código:

- Complejas
- Mal estructuradas
- Altamente acopladas
- Manejan datos sensibles o críticos

ORGANIZACIÓN DE LOS EQUIPOS



Organización

- Favorece la organización y coordinación de los equipos
- Ha dado solución a problemas en equipos
- Ayuda a coordinar equipos



ORGANIZACIÓN DE LOS EQUIPOS

- Sirve para proyectos Green Field y Brown Field
- La arquitectura debe basarse en los equipos y no al revés
- Las empresas pueden poner palos en la rueda sin saberlo



FAMILIARIDAD Y BRECHAS DE CONOCIMIENTO



Brechas de conocimiento

- Diferencia entre el conocimiento que tenemos y el que necesitamos
- Dificulta escribir software de calidad
- Distorsiona nuestra visión del código



Como solucionarlas:

Utilizando análisis de comportamiento de código

- Permite medirlas
- Viendo cómo se relacionan las personas con el código
- Con tiempo

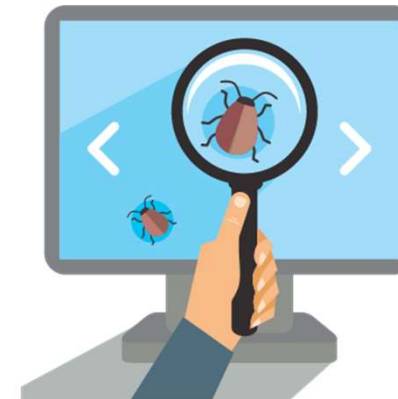
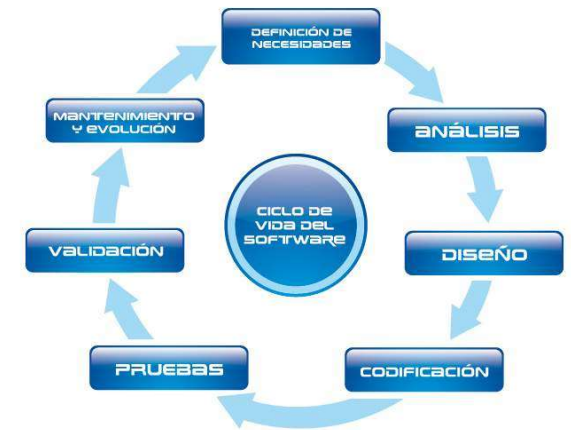
INTEGRACIÓN EN EL CICLO DE VIDA DEL SOFTWARE

Ciclo de vida del software

- Se puede integrar el análisis de comportamiento de código en el ciclo de vida
 - Al hacer un pull request
 - En retrospectivas
 - En reuniones
 - ...

Test

- Puede encontrar deuda técnica en test
- Gran problema en proyectos Legacy



ÉTICA Y MORAL



- El análisis de comportamiento de código puede tener problemas éticos y morales
- Se puede usar para clasificar desarrolladores
- Problemas dentro de los equipos
- Distorsión de métricas
- Peor rendimiento
- Los datos serán falsos

REGLA DEL BOY SCOUT

“Deja el código más limpio de como te lo encontraste”

No siempre hace falta aplicarla (según Adam Thornhill)

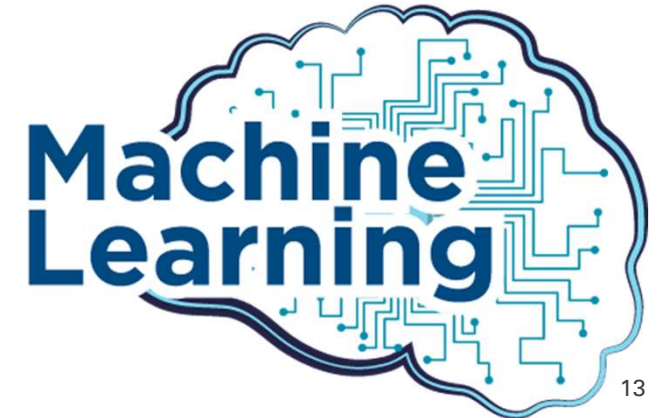
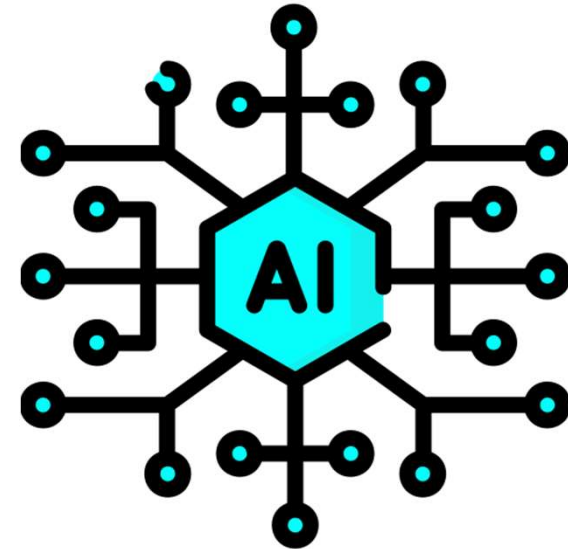
- Aplicarla en sitios con cambios frecuentes
 - Hay mucho trabajo en esa parte
 - Es importante para familiarizarnos con el código
- No aplicarla en sitios con cambios poco frecuentes
 - Es una zona estable
 - Es una zona crítica
- El análisis de comportamiento de código facilita encontrar dichos trozos de código
 - Mapa de hotspots



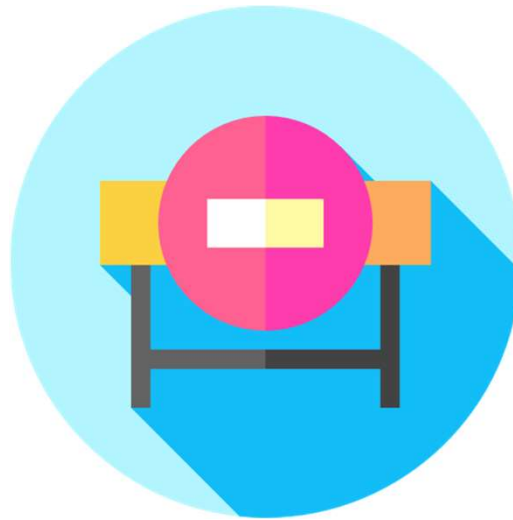
EL FUTURO DEL ANÁLISIS DE COMPORTAMIENTO DE CÓDIGO

Hacia dónde camina el análisis de comportamiento de código:

- Uso de la Inteligencia Artificial – Machine Learning
- Ventajas:
 - Automatiza el análisis
 - Ahorra tiempo a los desarrolladores
- Inconvenientes:
 - Hay que entrenar a la IA
 - No se debe abusar de ella (supervisión humana)



LOS LÍMITES ACTUALES



El análisis de comportamiento de código es reciente y tiene límites

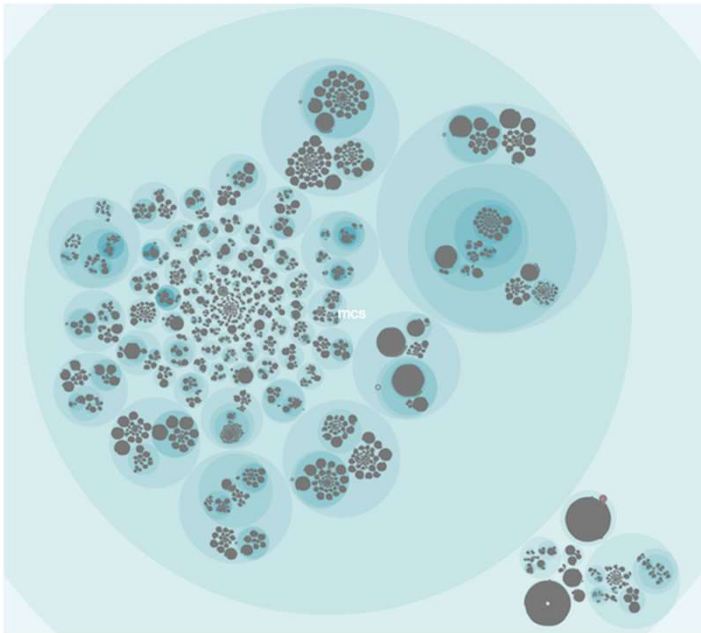
- Las herramientas de análisis no funcionan con todos los sistemas de control de versiones
- No hay una integración estable de la IA todavía

Change coupling, principal problema

- Problemas con las dependencias
- Componentes que cambian a la vez
- Demasiado abstracto para el análisis

HERRAMIENTAS

Code Maat



CodeScene



Git



