

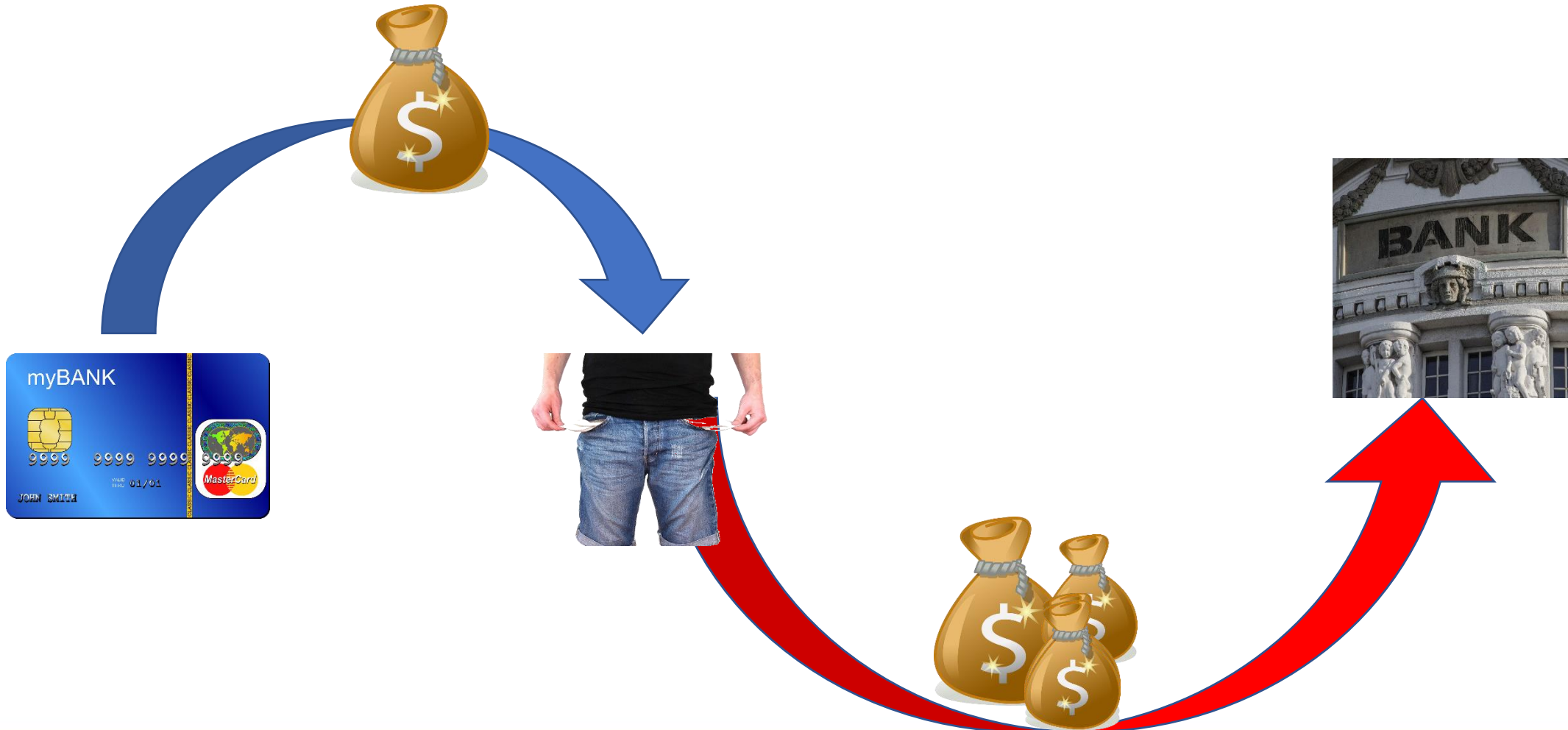


Deuda Técnica

Ing. PMP Nancy Belén Ramírez Martínez
Miguel Ángel Martínez Rivera
Jorge Eduardo García Santander.



El concepto de deuda





El concepto de deuda técnica (Cunningham, 1992)





El precio por omitir las buenas prácticas

La deuda técnica, es la deuda de **trabajo** que se adquiere al construir el software sin tomar en cuenta buenas prácticas, por ejemplo:

1. Definición inicial insuficiente.
2. Componentes estrechamente acoplados, donde las funciones no son modulares.
3. Ejecución de pruebas mínimas o en ambientes no controlados.
4. Falta de documentación.
5. Refactorización retrasada.
6. Deficiencias en arquitectura.
7. Cambios de última hora, etc.

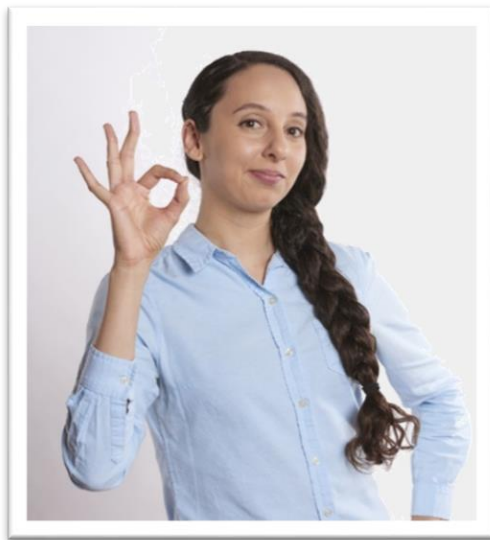


¿Cómo se paga la deuda?





La deuda técnica puede ser “invisible”.





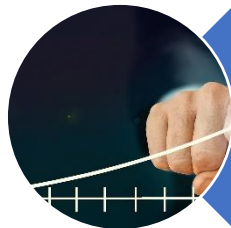
Gestión de la deuda técnica



Registro y medición Deuda Técnica



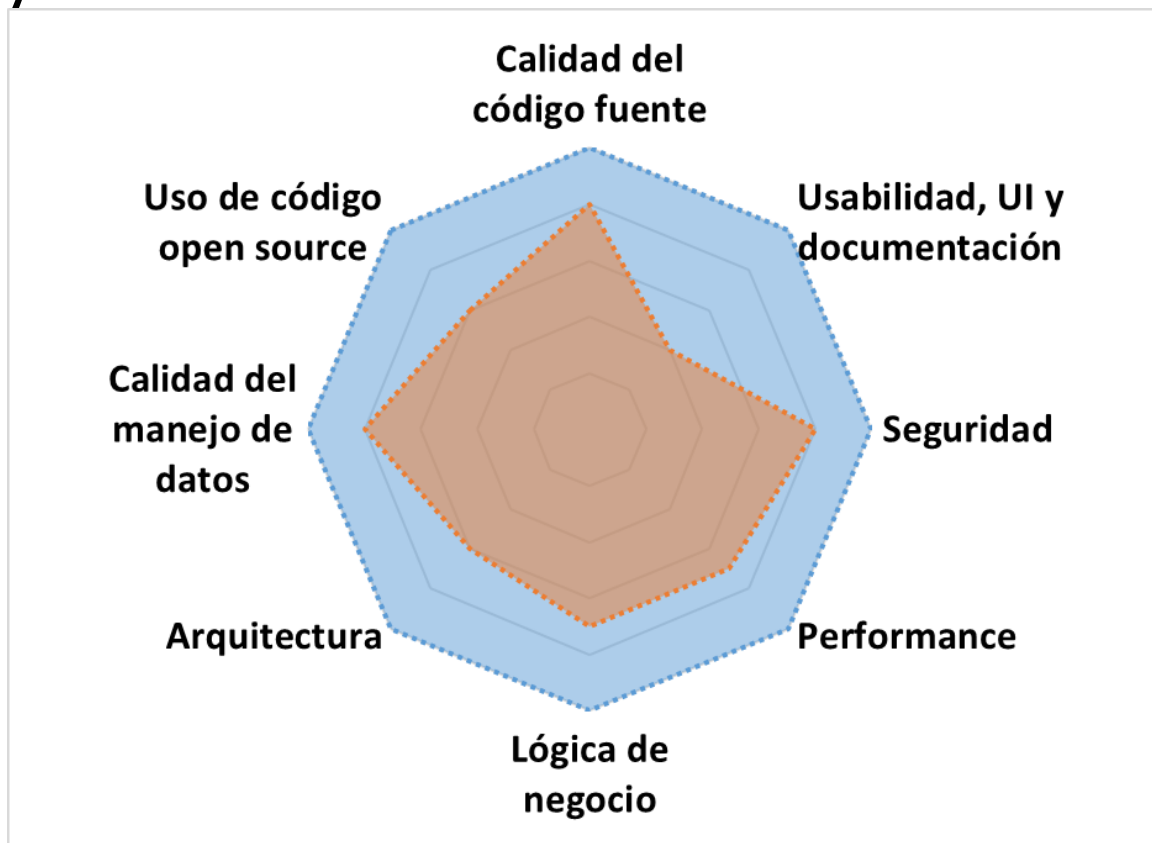
Análisis Costo-Beneficio



Controlar la Deuda Técnica



1. Registro y medición de la deuda técnica.



Plataforma de reducción de la deuda técnica, TETRA.
(Kontsevoi, Soroka, & Terekhov, 2019)



Cómo evaluar cada dimensión:

- **Calidad del código fuente:** SonarQube
- **Usabilidad, UI y documentación:** W3C Markup Validation Service, Usabilla, Browser Shots, etc.
- **Seguridad:** Burp Suite, OWASP ZAP, and Nessus.
- **Performance:** JMeter, LoadRunner, WebLoad and LoadUI Pro
- **Lógica de negocio:** Retroalimentación del área de negocio (Efectividad, flujo de negocio, seguridad de datos, simplicidad competitividad, reglas y políticas de negocio)
- **Arquitectura:** Evaluación de la estructura central del software y modelo de datos.
- **Calidad del manejo de datos:** Talend Open Studio y pruebas manuales.
- **Uso de código open source:** Black Duck and Palamida.



2.- Análisis Costo-Beneficio

1. Visibilizar la deuda técnica a los tomadores de decisiones.
2. Considerar la capacidad operativa del equipo de trabajo.
3. Estimar el esfuerzo / costo de la deuda con el equipo disponible.
4. Identificar las posibles consecuencias de “omitir” la deuda técnica.





Tipos de deuda técnica (Fowler, 2004)

	Imprudente	Prudente
Deliberada	“No tenemos tiempo para hacer el diseño”	“Tenemos que entregar ahora y lidiar con las consecuencias”
Inadvertida	“¿Qué es un diseño por capas?”	“Ahora sabemos como debimos haberlo hecho”



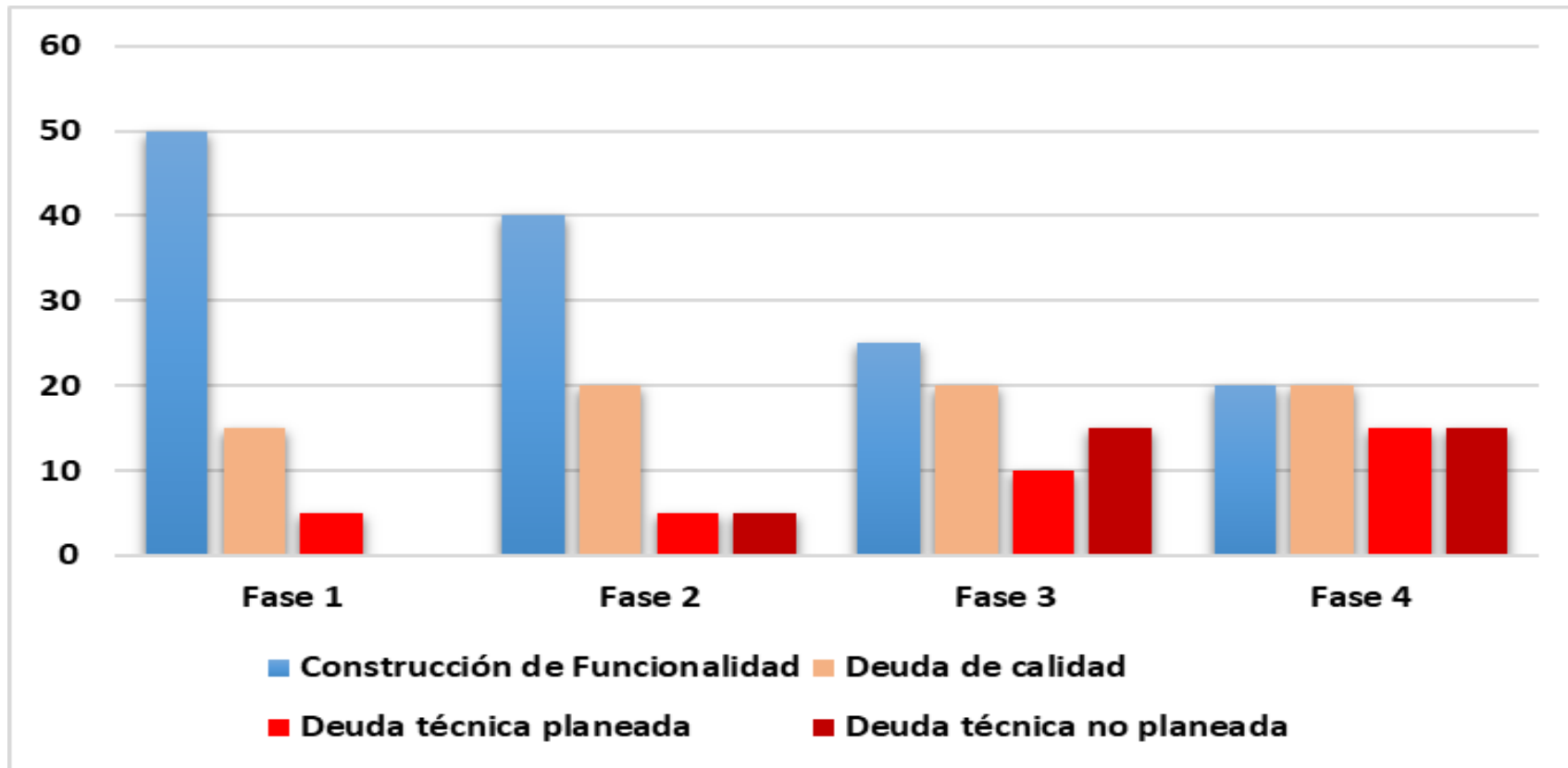
3. Control de la deuda técnica.

1. Planificar la atención gradual de la deuda técnica como parte del proyecto.
2. Monitorear la deuda técnica durante el desarrollo del proyecto, incluyendo la deuda técnica planeada y la no planeada.
3. Identificar los defectos que son provocados por deuda técnica.





¿Qué pasa si ignoramos la deuda técnica? (Power, 2013)





¿Por dónde comenzar?

1. Utiliza las herramientas disponibles, ejemplo SonarQube.
2. ¿Quién es el responsable de la deuda técnica?, 2 casos de estudio.



¿Qué es SonarQube?



Es una herramienta de revisión automática de código para detectar errores y vulnerabilidades.



Relación de la deuda técnica con SonarQube

El cálculo de la deuda técnica está basado en la metodología SQALE
(Software Quality Assessment based on Lifecycle Expectations -
Evaluación de la calidad del software basada en las expectativas del ciclo
de vida)



Características SQALE	
Reusability	Reusabilidad
Portability	Portabilidad
Maintainability	Mantenibilidad
Security	Seguridad
Usability	Usabilidad
Efficiency	Eficiencia
Changeability	Posibilidad de cambiar
Reliability	Fiabilidad
Testability	Testabilidad

▼ Maintainability ⓘ

Overview ⓘ

On new code

- Code Smells 30
- Debt 5h 53min
- Debt Ratio 0.6%
- Rating **A**

Overall

- Code Smells 51
- Debt 1d 4h
- Debt Ratio 0.5%
- Rating **A**
- Effort to Reach A 0



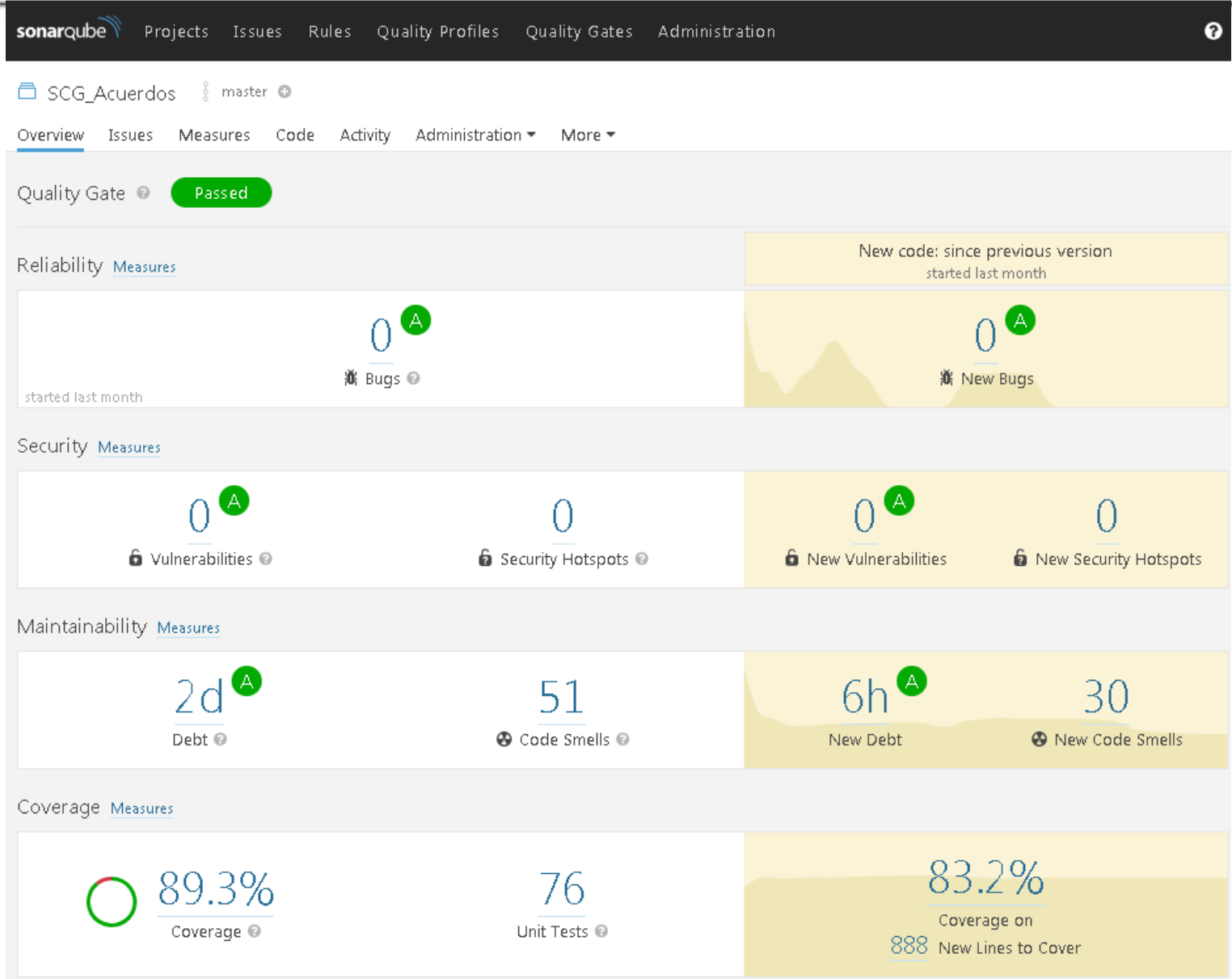
Color: Maintainability Rating Size: Code Smells

■ A ■ B ■ C ■ D ■ E

Calificación SQALE		Categoría	Relación Deuda Técnica en %
A	0-0.05	Calidad excelente	$\leq 5\%$
B	0.06-0.1	Calidad notable	$6\% \leq \text{TD Ratio} \leq 10\%$
C	0.11-0.20	Calidad insuficiente	$11\% \leq \text{TD Ratio} \leq 20\%$
D	0.21-0.5	Calidad baja	$21\% \leq \text{TD Ratio} \leq 50\%$
E	0.51-1	Calidad muy baja	$> 50\%$
TD (Technical Debt)			
Remediation cost / (Cost to develop 1 line of code * Number of lines of code)			



Ejemplo



Análisis de aplicación con SonarQube.

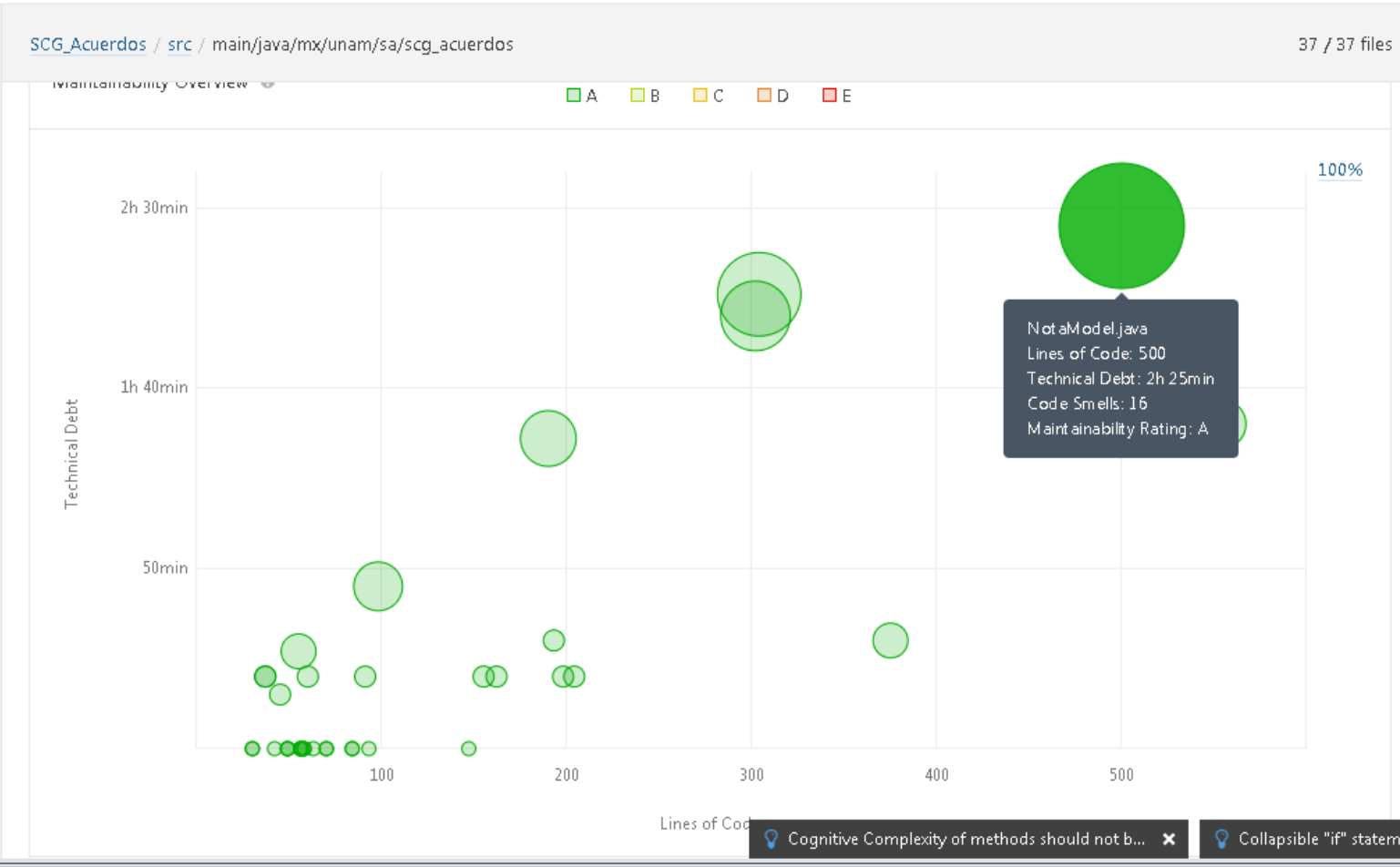


sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration + A

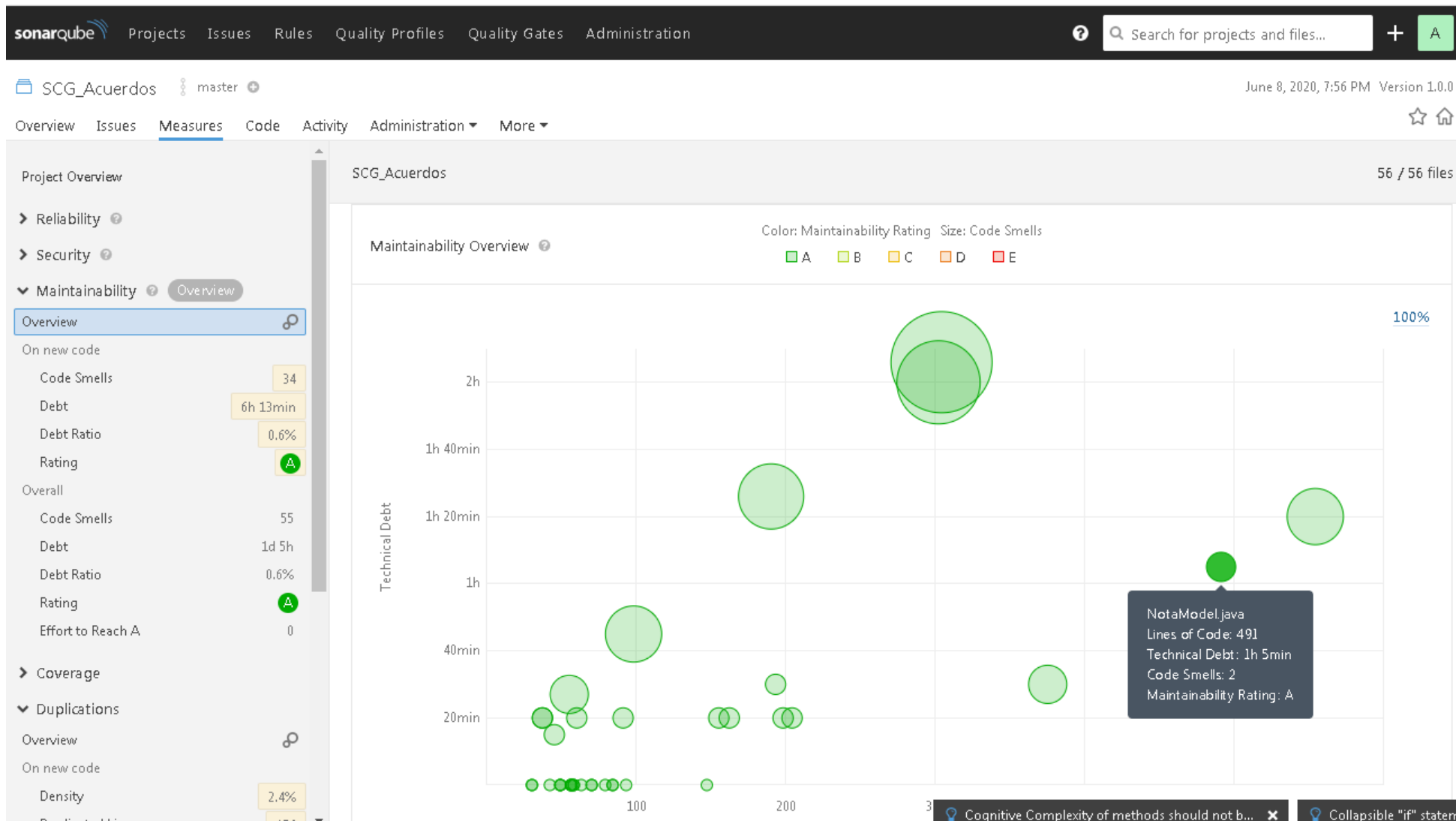
SCG_Acuerdos master June 8, 2020, 2:50 PM Version 1.0.0
Overview Issues Measures Code Activity Administration More

Project Overview

- Reliability
- Security
- Maintainability **Overview**
 - Overview
 - On new code
 - Code Smells 32
 - Debt 6h 13min
 - Debt Ratio 0.7%
 - Rating A
 - Overall
 - Code Smells 69
 - Debt 2d 2h
 - Debt Ratio 0.8%
 - Rating A
 - Effort to Reach A 0
 - Coverage
 - Duplications
 - Overview
 - On new code
 - Density 5.7%
 - Duplicated Lines 146



Antes: Gráfica de la deuda técnica del código fuente.

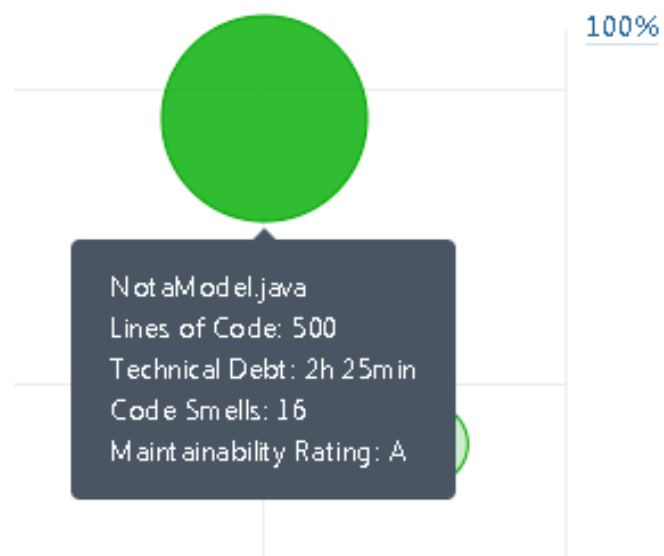


Después: Gráfica de la deuda técnica del código fuente.

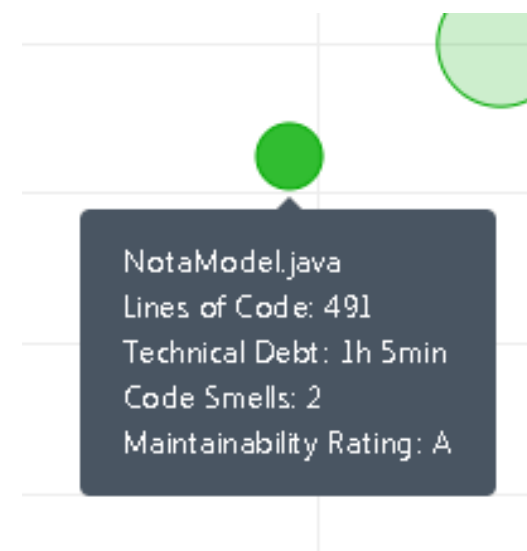


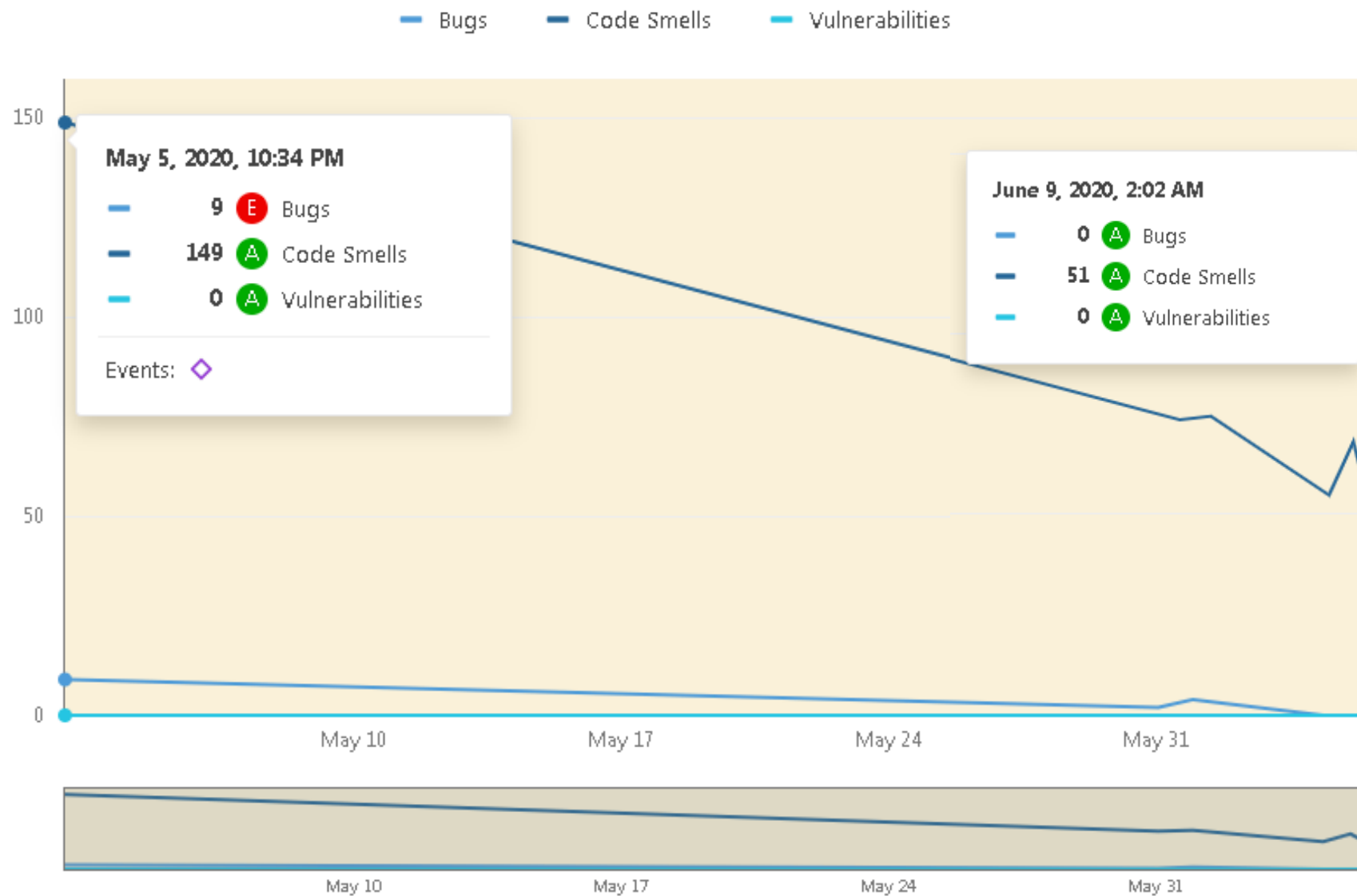
Comparación

Antes



Después





Evolución de la calidad del código fuente con SonarQube.



La deuda técnica no es “exclusiva” del equipo técnico

- Selección de herramientas sin consultar al equipo técnico
- Reasignación de recursos sin el conocimiento en las herramientas.
- Levantamiento incorrecto de requerimientos.
- Cambios de prioridades durante el desarrollo.





Caso 1 -

- La elección de lenguaje se hizo por preferencias personales del programador.
- No se le exigieron buenas prácticas.
- El programador se retiró por problemas administrativos.
- Se tomaron miembros de otras áreas en lugar de contratar otro especialista.
- Cambiaron de lenguaje al no conocer el original





- Inevitablemente se contrató otro programador.
- Existía código repetido.
- El código no estaba comentado.
- No existía control de versiones.
- Se desecharon grandes bloques de código.

- El lanzamiento se retrasó cerca de un año
- El nuevo especialista fue más caro.
- Las demás áreas sufrieron retrasos.





Caso 2 – Malos requerimientos

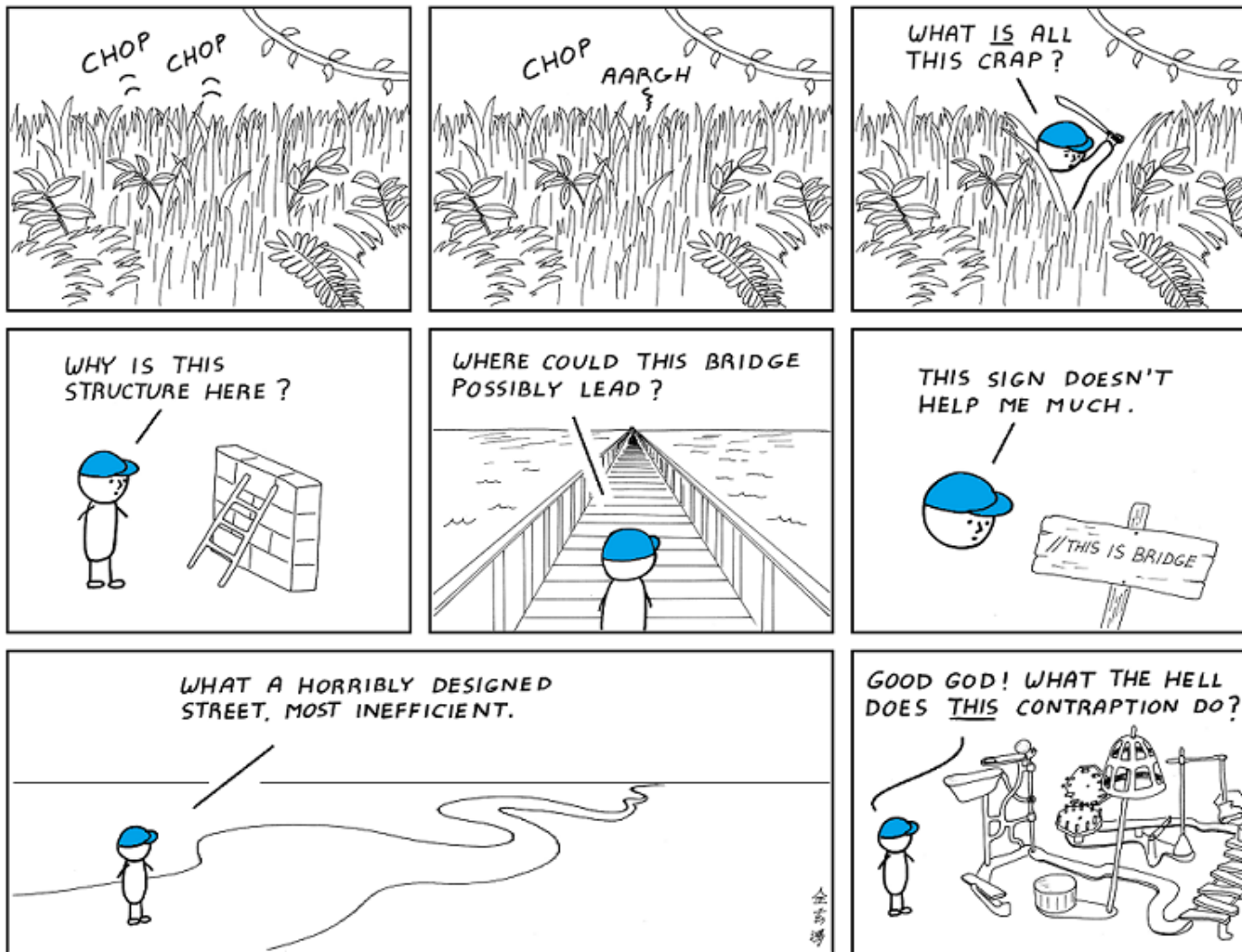
- Problemas organizacionales jerárquicos
- Soberbia del liderazgo: “Tenemos fecha límite para entregar y una eternidad para corregir”.
- El cliente rechazó el producto en fase de pruebas.





- La necesidad del sistema era alta y se decidió lanzar incompleto.
- Los desarrolladores se convirtieron en un equipo de mantenimiento semipermanente.
- Se pospuso el material adicional de capacitación.
- Fue difícil integrar personal adicional de apoyo.





I hate reading other people's code.

Referencia:
<https://medium.com/better-programming/7-habits-of-highly-effective-programmers-563ee3b63f33>



Conclusiones

- Tu decides cómo pagar la deuda técnica.
- Visibilizar la deuda técnica a los “tomadores de decisiones”.
- Hacer mucho, no significa “generar valor”.



Referencias

- W. Cunningham, "The WyCash Portfolio Management System," presented at the 7th Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA'92), Vancouver, British Columbia, Canada, 1992.
- Kontsevoi, B., Soroka, E., & Terekhov, S. (2019). TETRA, as a set of techniques and tools for calculating technical debt principal and interest. *Proceedings - 2019 IEEE/ACM International Conference on Technical Debt, TechDebt 2019*, 64–65.
<https://doi.org/10.1109/TechDebt.2019.00021>
- Power, K. (2013). Understanding the impact of technical debt on the capacity and velocity of teams and organizations: Viewing team and organization capacity as a portfolio of real options. *2013 4th International Workshop on Managing Technical Debt, MTD 2013 - Proceedings*, 28–31.
<https://doi.org/10.1109/MTD.2013.6608675>



Referencias

- <https://docs.sonarqube.org/display/SONARQUBE52/Technical+Debt>
- <http://www.sqale.org/details/details-indices-indicators>
- <https://docs.sonarqube.org/display/SONARQUBE52/Technical+Debt#TechnicalDebt-Wheretostartpayingthetechnicaldebt?>
- <https://docs.sonarqube.org/latest/>
- <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
- <http://www.sqale.org/wp-content/uploads/2016/08/SQALE-Method-EN-V1-1.pdf>



Gracias

Ing. PMP Nancy Belén Ramírez Martínez: belen@dgp.unam.mx

Miguel Ángel Martínez Rivera: miguelangel@dgp.unam.mx

Jorge Eduardo García Santander: jsantander@dgp.unam.mx

Israel Ortega Cuevas: iortega@unam.mx



ISO 25000

