



UNIVERSITÀ
DEGLI STUDI DI BARI
ALDO MORO

Documentazione di Progetto

Dipartimento di Informatica
Corso di laurea “Metodi e Modelli”

Metodi e Modelli
A.A. 2022/2023

Prof. Baldassare Maria Teresa

GRUPPO: e-tech

Membri

Davide Sebastiano Barbarossa Mat: 754695
Filippo Bilanzuoli Mat: 736640
Francesco Sabatelli Mat: 744409

Contatti

d.barbarossa2@studenti.uniba.it
f.bilanzuoli1@studenti.uniba.it
f.sabatelli11@studenti.uniba.it

INDICE

Il nostro progetto	3
Premessa	4
Report Preliminare	4
ActionPlan1	7
ActionPlan2	9
ActionPlan3	12
ActionPlan4	14
ActionPlan5	22
ActionPlan6	31
ActionPlan7	33
ActionPlan8	40
ActionPlan9	41
ActionPlan10	43
ActionPlan11	46
ActionPlan12	49
ActionPlan13	51
ActionPlan14	53
Parte Finale SonarCloud	55
ActionPlan1_Fortify.....	58
Note finali	60
Conclusioni	61

Il nostro progetto

Nome del progetto: PLC4X

Breve descrizione del progetto: Apache PLC4X è uno sforzo per creare un set di librerie per comunicare con controllori logici programmabili (PLC) di livello industriale in modo uniforme.

Porta utilità autonome (Java) come:

- Server OPC-UA: consente di comunicare con dispositivi legacy utilizzando PLC4X con OPC-UA.
- Server PLC4X: consente di comunicare con un server PLC4X centrale che poi comunica con i dispositivi tramite PLC4X.

Fornisce inoltre strumenti (Java) per l'utilizzo all'interno di un'applicazione:

- Connection Cache: nuova implementazione del nostro framework per il riutilizzo e la condivisione di connessioni PLC
- Pool di connessioni: vecchia implementazione del nostro framework per il riutilizzo e la condivisione di connessioni PLC
- OPM: Object-Plc-Mapping: consente di associare i campi PLC alle proprietà in java POJO simili a JPA
- Scraper: Utilità per eseguire la raccolta dati programmata e ripetuta.

Link GitHub del progetto:

<https://github.com/Filippo2020/e-tech.git>

Link al progetto SonarCloud:

<https://sonarcloud.io/summary/overall?id=Analisi-1>

Il progetto è interamente in Java, per modificarlo, abbiamo unitamente deciso di utilizzare l'IDE "Visual Studio Code 1.77.0", il quale ci ha facilitato la risoluzione di molte issue in molti casi grazie alle sue funzionalità.

Altri software utilizzati per l'esecuzione di questo progetto sono:

- Redmine
- Tortoise SVN
- SonarCloud
- Fortify

Durante l'ActionPlan4 abbiamo inserito nei nostri IDE il plugin "SonarLint", il quale analizza il progetto su cui si sta lavorando ed evidenzia i pezzi di codice in cui ci sono delle issue; perciò, esso ci ha permesso di notare subito se una determinata issue era effettivamente risolta o meno, facendoci risparmiare tempo.

Premessa

A causa dell'inaccessibilità alla piattaforma Redmine e Fortify, non ci è stato possibile recuperare alcuni file e dati, fondamentali, per fornire un'ottima visione dello stato del progetto in ogni actionPlan. Cercheremo di fornire una documentazione quanto più possibile dettagliata. Ci scusiamo per l'inconvenienza.

Report Preliminare

1. Valore di hash:

0bb2b4ba

2. Interfaccia principale di SonarCloud:

The screenshot shows the SonarCloud interface for the PLC4X project. The main dashboard provides an overview of the code quality across several dimensions:

- Reliability:** 70 Bugs (E)
- Maintainability:** 3.7k Code Smells (A)
- Security:** 0 Vulnerabilities (A)
- Security Review:** 84 Security Hotspots (0.0% Reviewed) (E)
- Coverage:** 14.6% Coverage (Ongoing)
- Duplications:** 7.0% Duplications (Ongoing)

A central message indicates that the "Quality Gate" has not been computed. The interface also shows the project path (mmqs-e-tech-2022 > PLC4X > master), the version (0.11.0-SNAPSHOT), and the last analysis date (2 days ago).

3. Maven test:

```

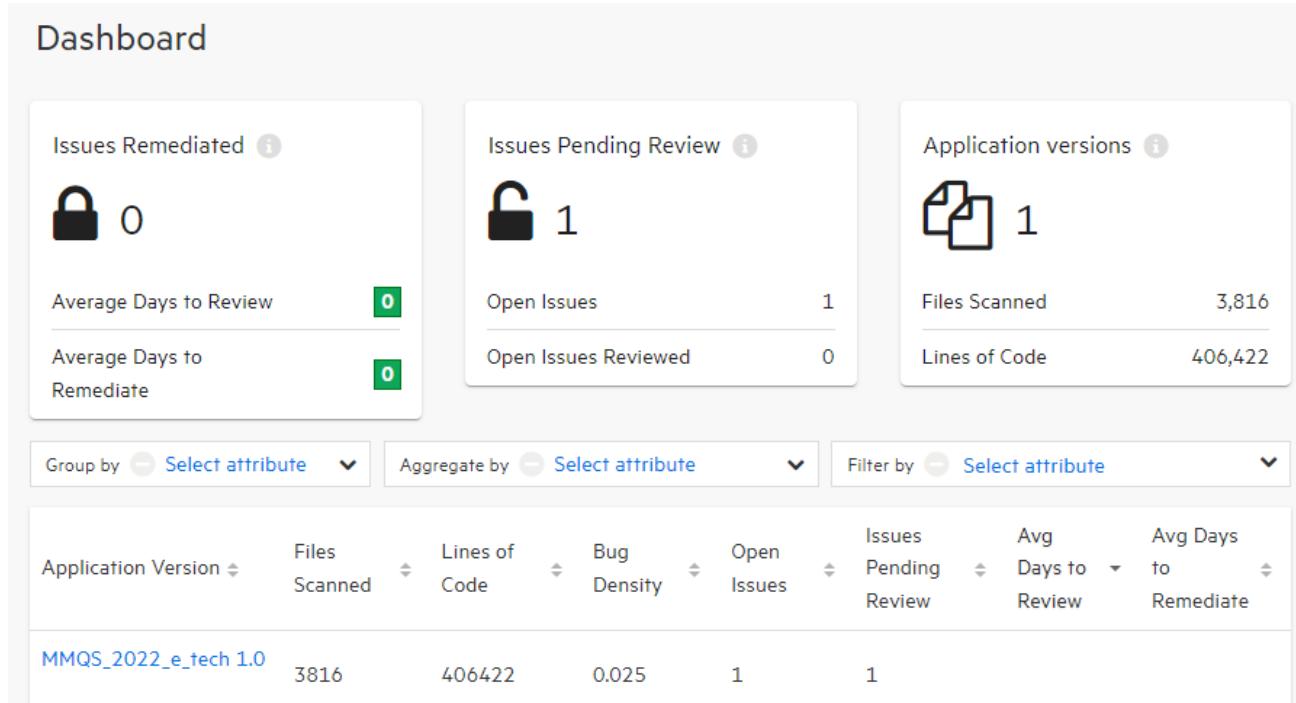
[INFO] PLC4J: Driver: PLC4X (Proxy-Protocol) ..... SUCCESS [ 29.271 s]
[INFO] PLC4J: Driver: Profinet ..... SUCCESS [ 17.723 s]
[INFO] PLC4J: Driver: S7 (Step7) ..... SUCCESS [ 27.492 s]
[INFO] PLC4J: Driver: Simulated ..... SUCCESS [ 28.564 s]
[INFO] PLC4J: Examples ..... SUCCESS [ 0.064 s]
[INFO] PLC4J: Examples: Cloud: Azure IoT Hub ..... SUCCESS [ 26.596 s]
[INFO] PLC4J: Examples: Cloud: Google Cloud IoT Core ..... SUCCESS [ 3.098 s]
[INFO] PLC4J: Examples: Connectivity: MQTT ..... SUCCESS [ 28.315 s]
[INFO] PLC4J: Examples: Hello Connectivity: MQTT ..... SUCCESS [ 01:17 min]
[INFO] PLC4J: Examples: Hello InfluxDB Data Collection ..... SUCCESS [ 0.078 s]
[INFO] PLC4J: Tools ..... SUCCESS [ 1.247 s]
[INFO] PLC4J: Tools: Object PLC Mapping ..... SUCCESS [ 15.390 s]
[INFO] PLC4J: Tools: Connection Pool ..... SUCCESS [ 10.124 s]
[INFO] PLC4J: Examples: Hello OPM ..... SUCCESS [ 1.413 s]
[INFO] PLC4J: Examples: Hello Webservice ..... SUCCESS [ 1.247 s]
[INFO] PLC4J: Examples: Hello-World: Kotlin ..... SUCCESS [ 08:22 min]
[INFO] PLC4J: Examples: Hello-World: PLC4X (Discover And Browse) ..... SUCCESS [ 0:00:387 s]
[INFO] PLC4J: Examples: Hello-World: PLC4X (Read) ..... SUCCESS [ 1.130 s]
[INFO] PLC4J: Examples: Hello-World: PLC4X (Subscribe) ..... SUCCESS [ 1.093 s]
[INFO] PLC4J: Examples: Hello-World: PLC4X (Write) ..... SUCCESS [ 1.068 s]
[INFO] PLC4J: Examples: Poll Loop Example ..... SUCCESS [ 1.006 s]
[INFO] PLC4J: Examples: S7Event example ..... SUCCESS [ 1.428 s]
[INFO] PLC4J: Tools: Connection Cache ..... SUCCESS [ 16.793 s]
[INFO] PLC4J: Tools: Scraper ..... SUCCESS [ 43.481 s]
[INFO] PLC4J: Integrations ..... SUCCESS [ 0.057 s]
[INFO] PLC4J: Integrations: Apache Camel ..... SUCCESS [ 02:57 min]
[INFO] PLC4J: Integrations: Apache Nifi ..... SUCCESS [ 02:57 min]
[INFO] PLC4J: Integrations: Apache Nifi: Processors ..... SUCCESS [ 23:15 min]
[INFO] PLC4J: Integrations: Apache Nifi: NAR ..... SUCCESS [ 12.008 s]
[INFO] PLC4J: Integrations: Apache Calcite ..... SUCCESS [ 01:23 min]
[INFO] PLC4J: Integrations: Eclipse Milo OPC UA Server ..... SUCCESS [ 13.146 s]
[INFO] PLC4J: Tools: Capture Replay ..... SUCCESS [ 0.451 s]
[INFO] PLC4J: Tools: PLC4x Server ..... SUCCESS [ 5.176 s]
[INFO] PLC4J: Tools: UI ..... SUCCESS [ 29.330 s]
[INFO] PLC4J: Utils: PLC-Simulator ..... SUCCESS [ 14.441 s]
[INFO] PLC4J: OSGi ..... SUCCESS [ 1.026 s]
[INFO]
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:24 h
[INFO] Finished at: 2022-11-10T12:09:00+01:00
[INFO] -----

```

Activity SonarCloud:



4. Dashboard di Fortify:



Dopo aver effettuato la prima analisi del progetto sia di SonarCloud che di Fortify, i resoconti erano i seguenti:

SonarCloud:

- 3700 Code smells;
- 70 Bugs;
- 0 Vulnerabilità;
- 84 Security Hotspots.

Secondo la prima analisi di SonarCloud, l'effort stimato era di 53 giorni.

Poiché il numero di Code Smells era molto grande rispetto ai bugs e ai Security Hotspots, l'effort stimato è diminuito regolarmente ad ogni actionPlan a seconda del numero di issus risolte.

Fortify:

- 16.053 issus, di cui:
 - 16.000 LOW
 - 53 HIGH

I target che dovevamo raggiungere per SonarCloud e Fortify erano:

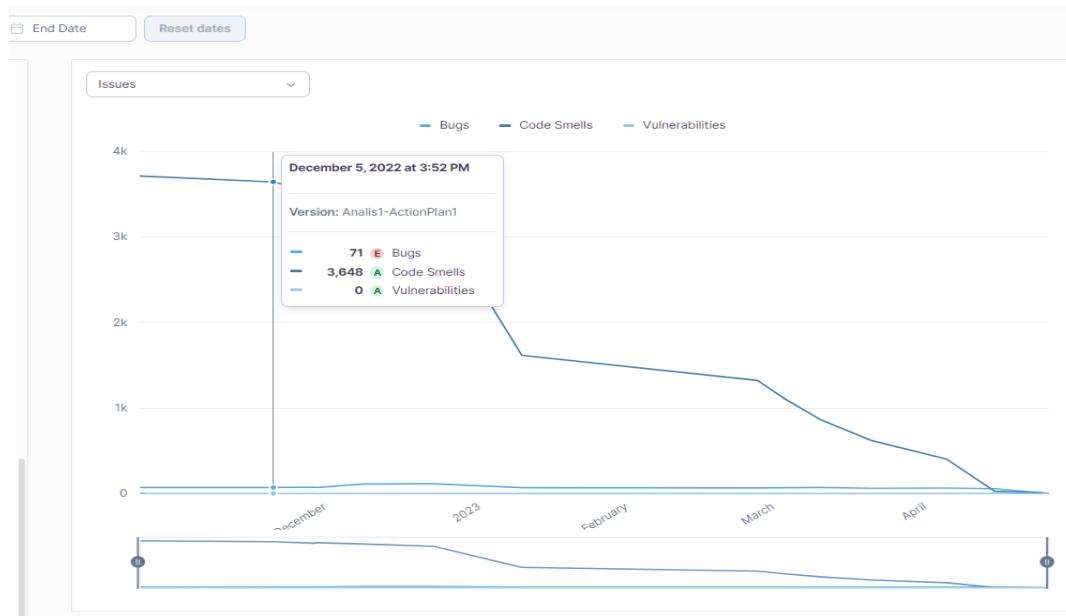
SonarCloud:

- Bugs: A (0)
- Security Hotspot: A (0)
- Vulnerabilities: A (0)
- Technical Debt (Maintainability): 0 days

Fortify:

- CRITICAL: 0
- HIGH: 50
- MEDIUM: 0
- LOW: 16.000

ActionPlan1



Dalla prima analisi (Analisi1.csv) sono stati estratti 186 issus che sono state divise in tre actionPlan(actionPlan1, actionPlan2, actionPlan3) divisi per numero e per tempo stimato di issus . Il target che ci eravamo fissati per l'ActionPlan1 era quello di risolvere 62 Code Smells per il tempo stimato di 5 minuti ciascuna.

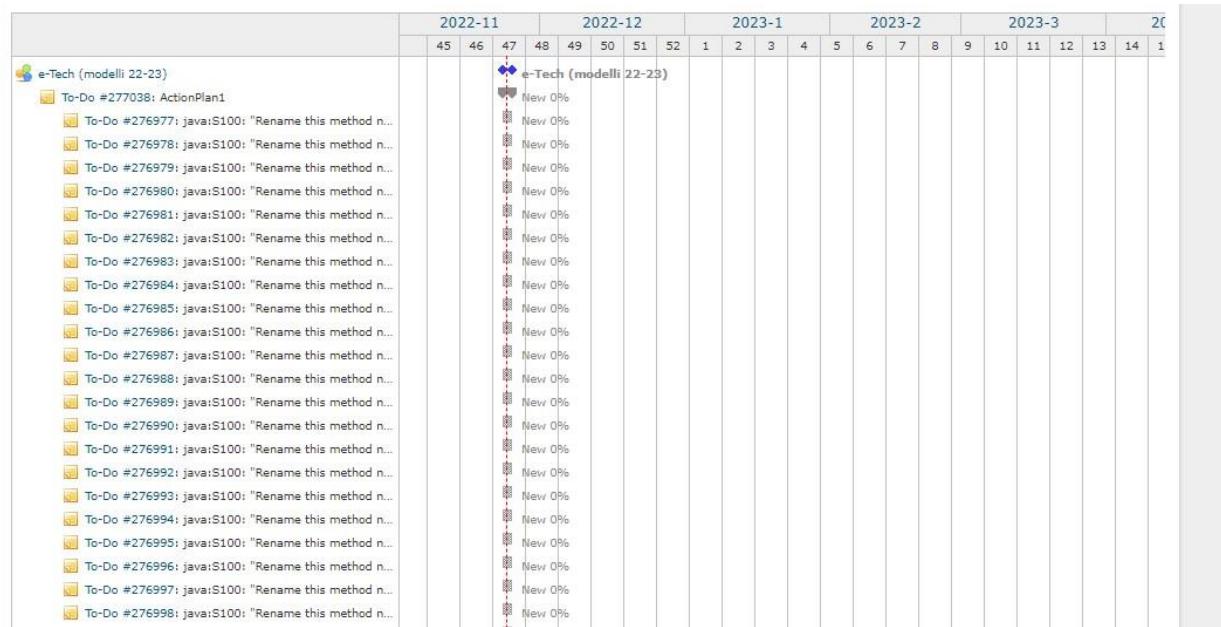
I Code smells in questione erano tutti dello stesso tipo (java: S100); quindi, dopo averli selezionati e copiati in un altro file chiamato ActionPlan1.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 20/21 a testa.

L'estimated time di questo ActionPlan è di 5.10 ore; invece, lo spent time è di circa 3.00 ore.

Questo ActionPlan aveva come data di partenza il 23/11/2022 e come data di scadenza il 3/12/2022, è stato chiuso, invece, il 5/12/2022, con due di ritardo perché non avevamo familiarità con la piattaforma di redmine.

Nelle pagine seguenti ci sono gli screen delle versioni del Gantt di Progetto, del Gannt esecutivo e in coda un grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:

Gannt di Progetto:



Gantt Esecutivo:

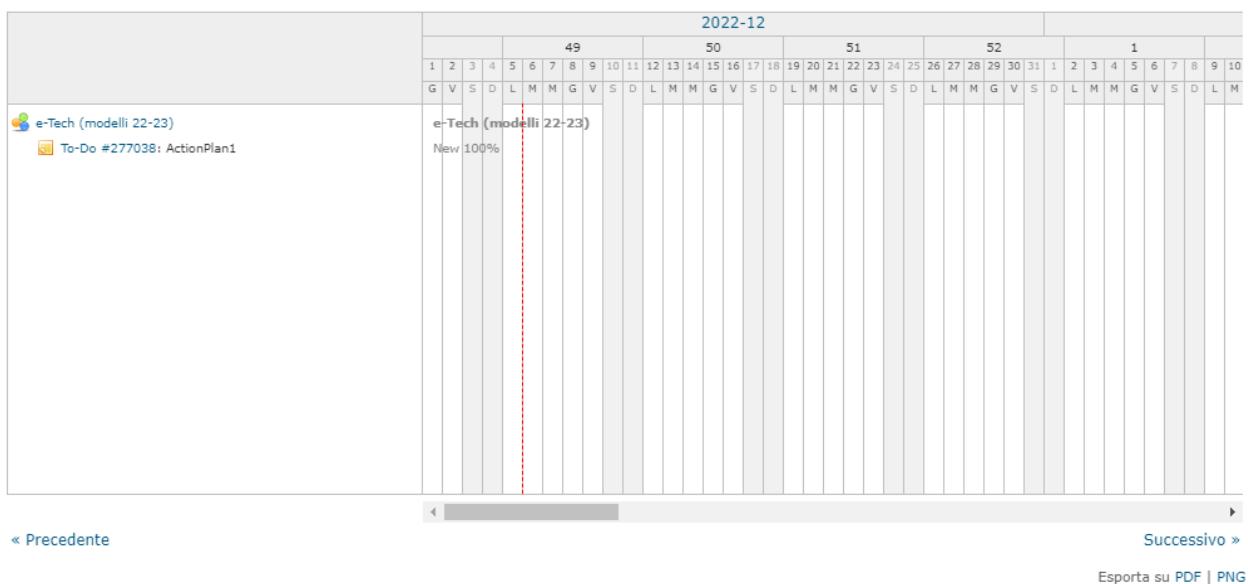
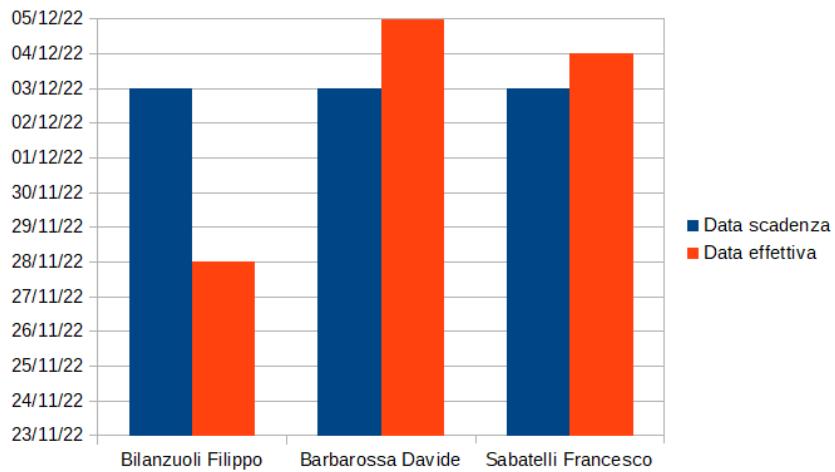
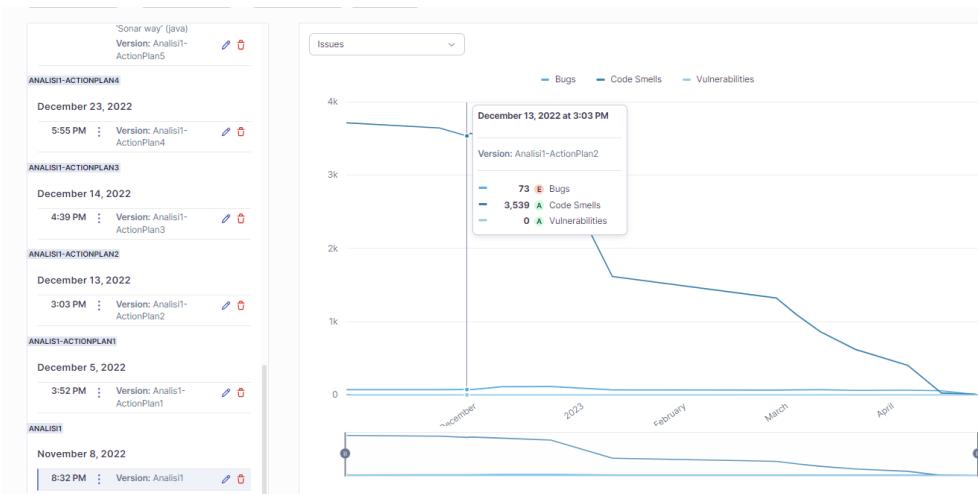


Grafico dell'ActionPlan1



ActionPlan2



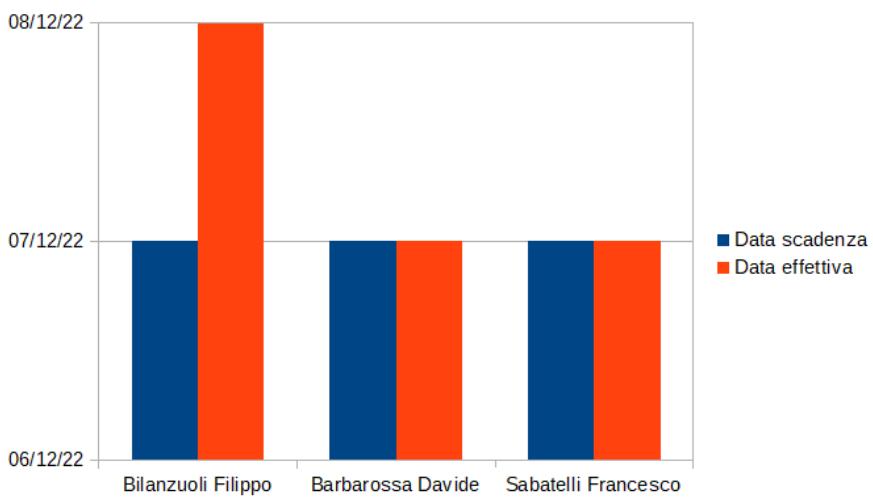
Dopo aver completato l'ActionPlan1, siamo passati all'ActionPlan2.

Dopo averli selezionati e copiati in un altro file chiamato ActionPlan2.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 21/22 a testa.

L'estimated time di questo ActionPlan è di 10.70 ore; invece, lo spent time è di 3.2 ore.

Questo ActionPlan aveva come data di partenza il 6/12/2022 e come data di scadenza il 7/12/2022, è stato chiuso, invece, l'8/12/2022, con un giorno di ritardo.

Di seguito riportiamo il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan3



Dopo aver completato l'ActionPlan2, siamo passati all'ActionPlan3.

Una volta selezionate e copiate in un altro file chiamato ActionPlan3.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 20 a testa.

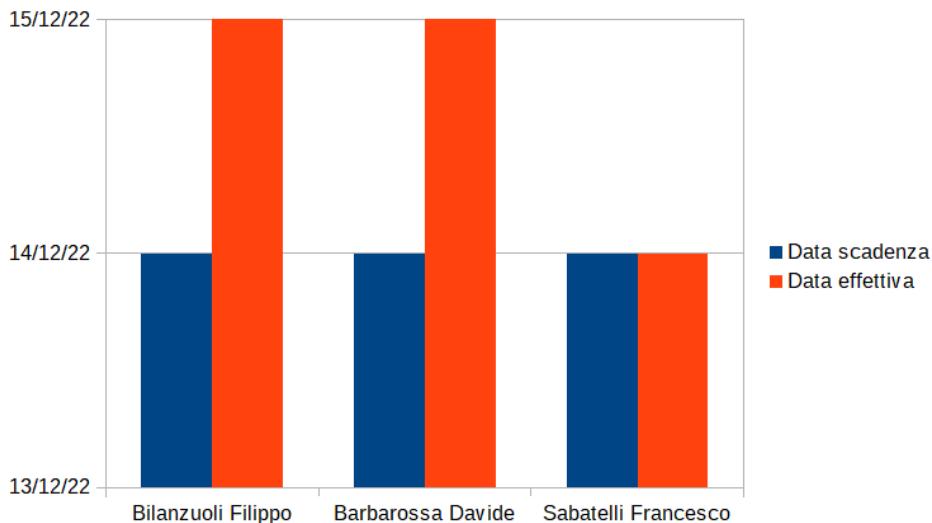
L'estimated time di questo ActionPlan è di 5.00 ore; invece, lo spent time è di 2.00 ore.

Questo ActionPlan aveva come data di partenza il 13/12/2022 e come data di scadenza il 14/12/2022, è stato chiuso, invece, l'15/12/2022, con un giorno di ritardo.

Di seguito riportiamo il Gantt di progetto il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:

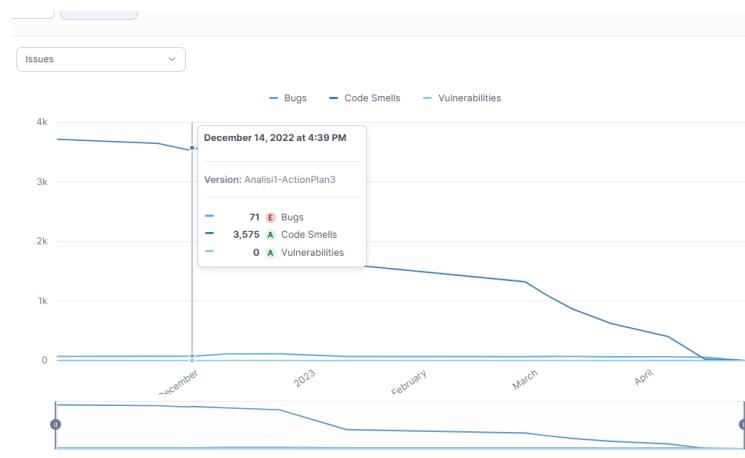


Grafico actionPlan3:



ActionPlan4

Al termine dell'ActionPlan3, abbiamo eseguito l'analisi di SonarCloud e abbiamo notato che si erano risolte le Issues svolte durante l'ActionPlan, senza generarne altre.

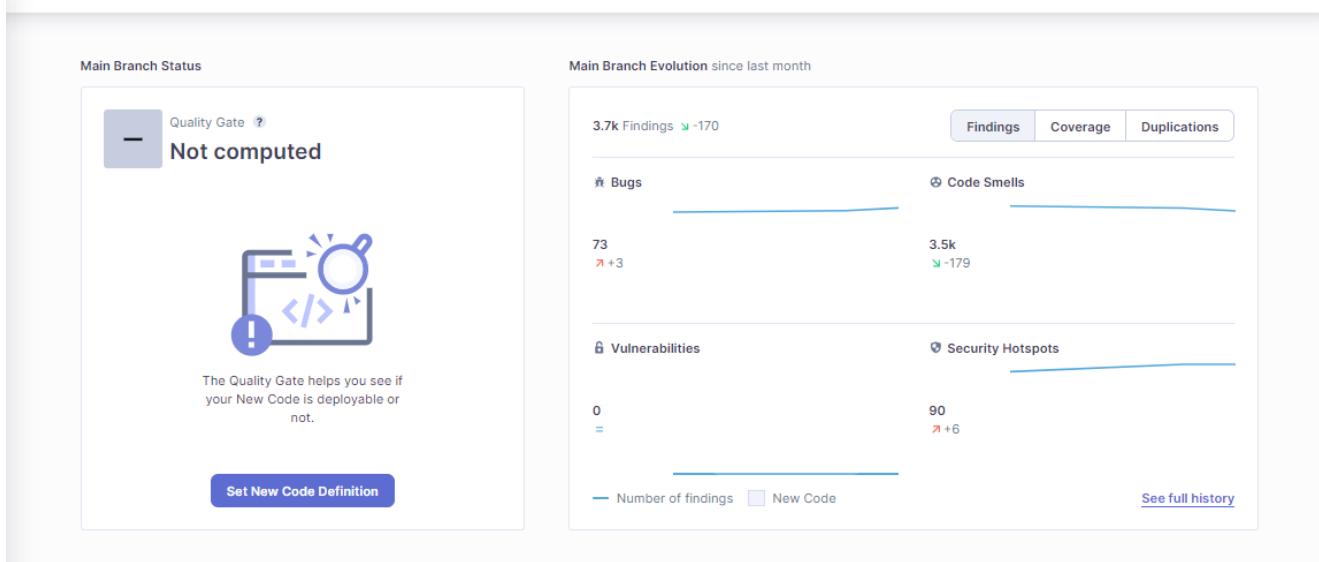


Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 160 Code Smells che erano di diverso tipo e che si trovavano nel file Analisi.3csv.

Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan4.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 53/54 a testa.

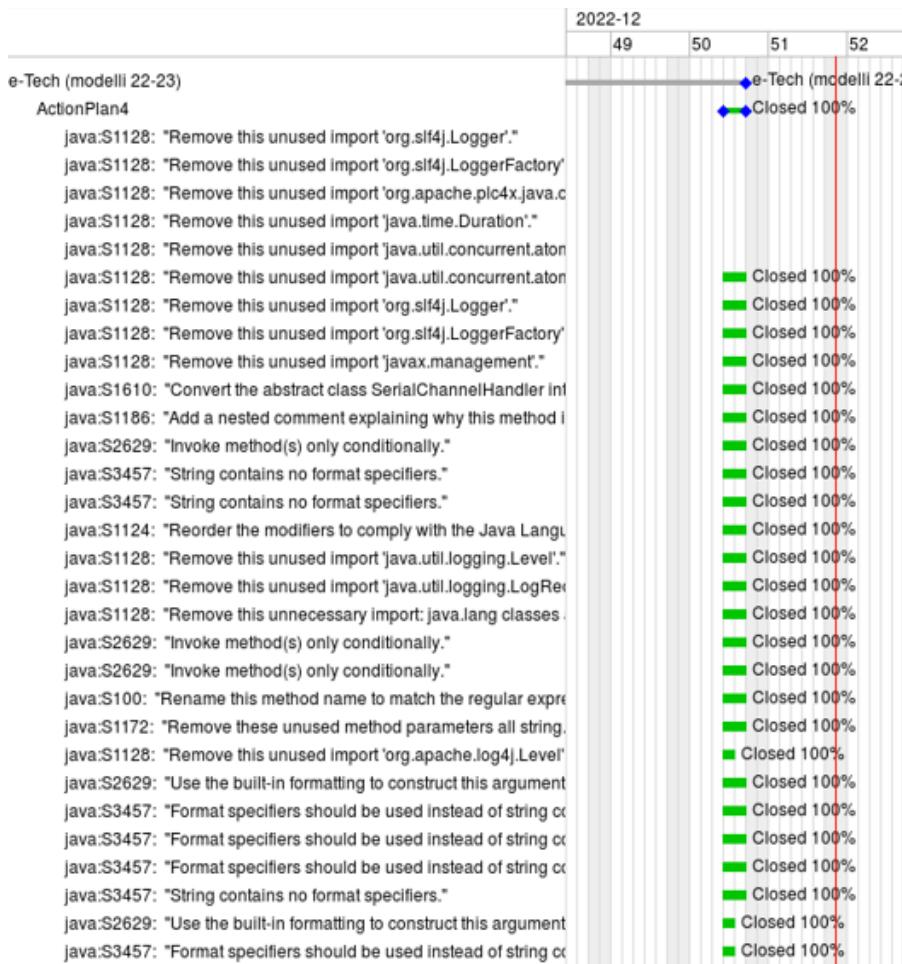
L'estimated time di questo ActionPlan è di 23.34 ore; invece, lo spent time è di 15.40 ore.

Questo ActionPlan aveva come data di partenza il 16/12/2022 e come data di scadenza il 26/12/2022, ed è stato chiuso il 29/12/2022, con 3 giorni di ritardo dovuti al mal funzionamento della piattaforma Redmine.



Gantt di Progetto:

Ganntt Esecutivo:

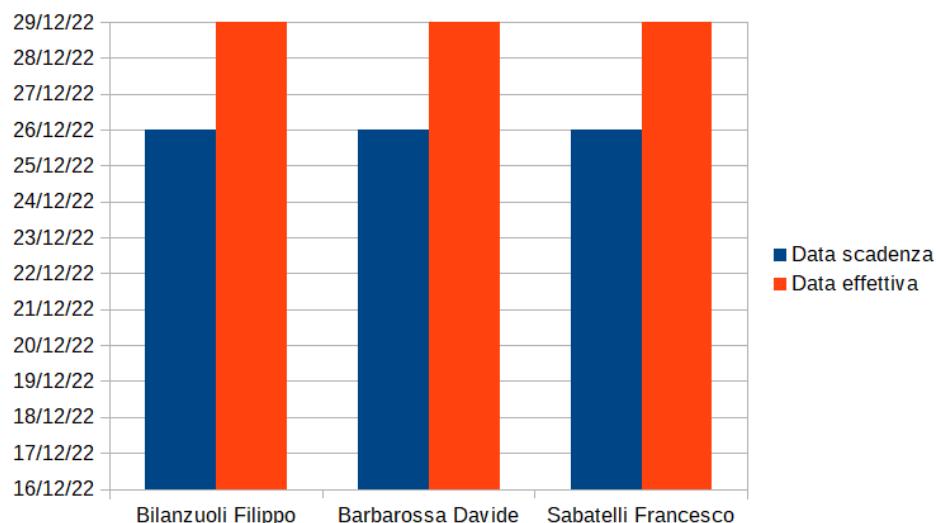


java:S1128: "Remove this unused import 'org.apache.plc4x.java.a
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a
java:S1128: "Remove this unused import 'java.util.List'."
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a
java:S1319: "The type of queries should be an interface such as M
java:S1602: "Remove useless curly braces around statement"
java:S1319: "The type of tags should be an interface such as Map
java:S112: "Define and throw a dedicated exception instead of us
java:S1602: "Remove useless curly braces around statement"
java:S1117: "Rename values which hides the field declared at line
java:S1319: "The type of tags should be an interface such as Map
java:S112: "Define and throw a dedicated exception instead of us
java:S1192: "Define a constant instead of duplicating this literal D
java:S112: "Define and throw a dedicated exception instead of us
java:S1319: "The type of tags should be an interface such as Map
java:S112: "Define and throw a dedicated exception instead of us
java:S1128: "Remove this unused import 'java.time.Instant'."
java:S1128: "Remove this unused import 'java.time.ZoneId'."
java:S2176: "Rename this class."
java:S3776: "Refactor this method to reduce its Cognitive Complexity."
java:S3776: "Refactor this method to reduce its Cognitive Complexity."
java:S1452: "Remove usage of generic wildcard type."
java:S1604: "Make this anonymous inner class a lambda"
java:S1488: "Immediately return this expression instead of assigning it."
java:S3011: "This accessibility bypass should be removed."
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a
java:S106: "Replace this use of System.out or System.err by a log
java:S106: "Replace this use of System.out or System.err by a log
java:S106: "Replace this use of System.out or System.err by a log
java:S1075: "Refactor your code to get this URI from a customization."
java:S1075: "Refactor your code to get this URI from a customization."
java:S1192: "Define a constant instead of duplicating this literal D
java:S106: "Replace this use of System.out or System.err by a log
java:S106: "Replace this use of System.out or System.err by a log
java:S106: "Replace this use of System.out or System.err by a log
java:S106: "Replace this use of System.out or System.err by a log
java:S1192: "Define a constant instead of duplicating this literal T

Closed 100%
Resolved 100%
Closed 100%
Closed 100%
Closed 100%
Resolved 100%
Closed 100%

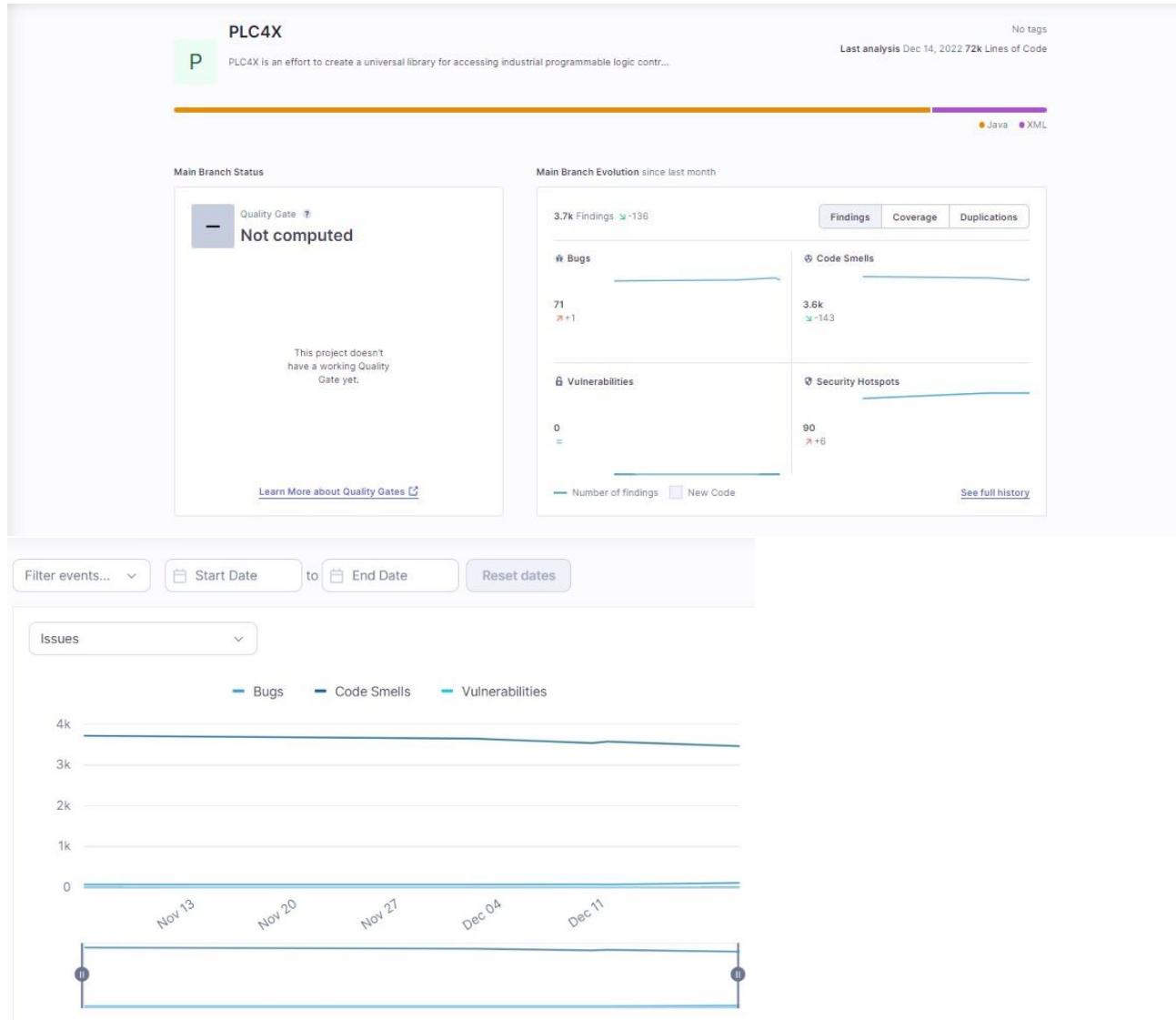
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S1172: "Remove this unused method parameter data."	Closed 100%
java:S117: "Rename this local variable to match the regular expre	Closed 100%
java:S117: "Rename this local variable to match the regular expre	Closed 100%
java:S117: "Rename this local variable to match the regular expre	Closed 100%
java:S1319: "The type of textlists should be an interface such as M	Closed 100%
java:S3776: "Refactor this method to reduce its Cognitive Comple	Closed 100%
java:S1172: "Remove this unused method parameter textlists."	Closed 100%
java:S1319: "The type of textlists should be an interface such as M	Closed 100%
java:S3776: "Refactor this method to reduce its Cognitive Comple	Closed 100%
java:S1172: "Remove this unused method parameter textlists."	Closed 100%
java:S2629: "Invoke method(s) only conditionally."	Closed 100%
java:S3457: "No need to call <code>toString()</code> method as formatting and :	Closed 100%
java:S4274: "Replace this assert with a proper check."	Closed 100%
java:S4274: "Replace this assert with a proper check."	Closed 100%
java:S1068: "Remove this unused TAG_DOC private field."	Closed 100%
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a	Closed 100%
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a	Closed 100%
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a	Closed 100%
java:S1128: "Remove this unused import 'java.util.List'."	Closed 100%
java:S1128: "Remove this unused import 'org.apache.plc4x.java.a	Closed 100%
java:S1319: "The type of queries should be an interface such as M	Closed 100%
java:S1602: "Remove useless curly braces around statement"	Closed 100%
java:S1319: "The type of tags should be an interface such as Map	Closed 100%
java:S112: "Define and throw a dedicated exception instead of us	Closed 100%
java:S1602: "Remove useless curly braces around statement"	Closed 100%
java:S1117: "Rename values which hides the field declared at line	Closed 100%
java:S1319: "The type of tags should be an interface such as Map	Closed 100%
java:S112: "Define and throw a dedicated exception instead of us	Closed 100%
java:S1192: "Define a constant instead of duplicating this literal D	Closed 100%
java:S112: "Define and throw a dedicated exception instead of us	Closed 100%
java:S1319: "The type of tags should be an interface such as Map	Closed 100%

Grafico actionPlan4:



ActionPlan5

Al termine dell'ActionPlan4, rieseguendo l'analisi di SonarCloud, sono aumentati di 44 i bugs, mentre i Code Smells che avevamo assegnato sono stati risolti con successo.



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 210 code smells di diverso tipo che trovavamo nel file Analisi5.csv.

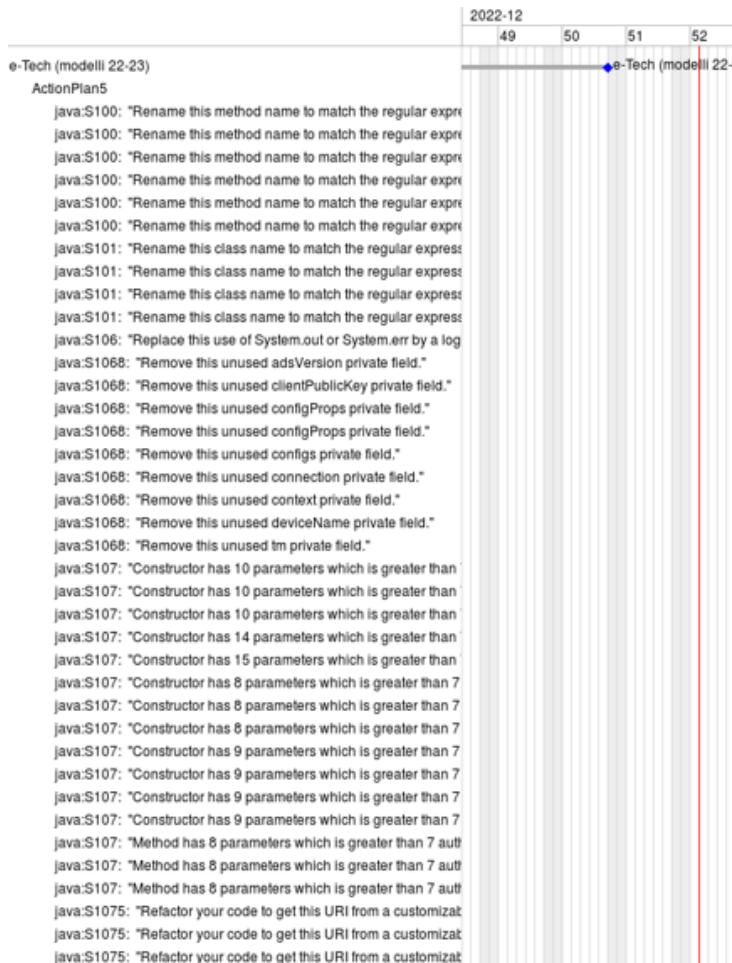
Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan5.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 70 a testa.

L'estimated time di questo ActionPlan è di 64.20 ore; invece, lo spent time è di 17.50 ore.

Questo ActionPlan aveva come data di partenza il 26/12/2022 e come data di scadenza il 6/01/2023, ed è stato chiuso il 6/01/2023, in tempo con la data di scadenza.

Nelle pagine seguenti ci sono gli screen delle versioni del Gantt di Progetto e del Gantt Esecutivo ed in coda un grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:

Gannt di Progetto:

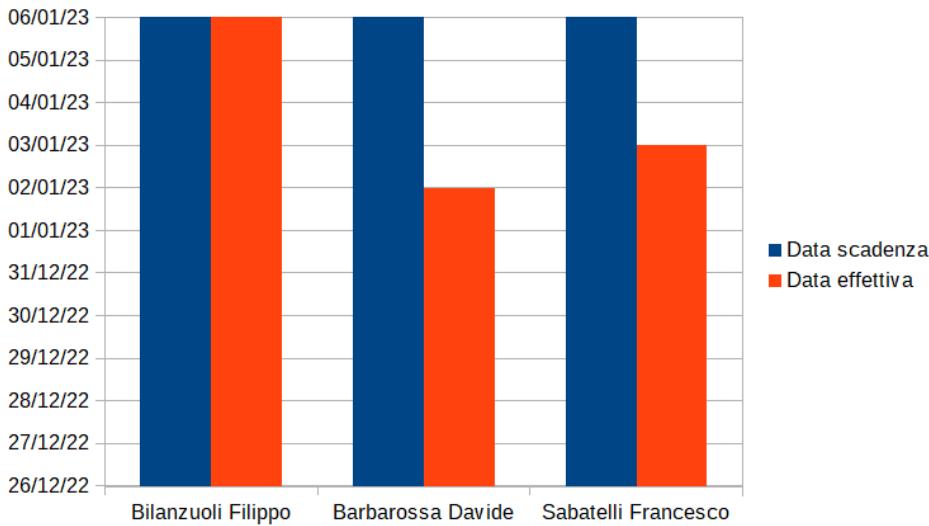


Gantt Esecutivo:

	2022-12	2023-1					
	49	50	51	52	1	2	3
e-Tech (modelli 22-23)							
ActionPlan5							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S100: "Rename this method name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S101: "Rename this class name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S101: "Rename this class name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S101: "Rename this class name to match the regular expression: ^[a-zA-Z_][a-zA-Z_0-9]*\$."							
java:S106: "Replace this use of System.out or System.err by a log statement."							
java:S107: "Constructor has 8 parameters which is greater than 7."							
java:S107: "Constructor has 8 parameters which is greater than 7."							
java:S107: "Constructor has 9 parameters which is greater than 7."							
java:S107: "Constructor has 9 parameters which is greater than 7."							
java:S107: "Constructor has 9 parameters which is greater than 7."							
java:S107: "Method has 8 parameters which is greater than 7 and is not static."							
java:S107: "Method has 8 parameters which is greater than 7 and is not static."							
java:S107: "Method has 8 parameters which is greater than 7 and is not static."							
java:S1075: "Refactor your code to get this URI from a customizable source."							
java:S1075: "Refactor your code to get this URI from a customizable source."							
java:S1075: "Refactor your code to get this URI from a customizable source."							
java:S1075: "Remove this hard-coded path-delimiter."							
java:S108: "Either remove or fill this block of code."							
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S108: "Either remove or fill this block of code."					Closed 100%		
java:S110: "This class has 6 parents which is greater than 5 and is not abstract."					Closed 100%		
java:S110: "This class has 6 parents which is greater than 5 and is not abstract."					Closed 100%		
java:S110: "This class has 6 parents which is greater than 5 and is not abstract."					Closed 100%		
java:S110: "This class has 6 parents which is greater than 5 and is not abstract."					Closed 100%		
java:S110: "This class has 6 parents which is greater than 5 and is not abstract."					Closed 100%		
java:S1104: "Make DebugHex a static final constant or non-public field."					Closed 100%		
java:S1104: "Make DEFAULT a static final constant or non-public field."					Closed 100%		
java:S1104: "Make LIGHT a static final constant or non-public field."					Closed 100%		

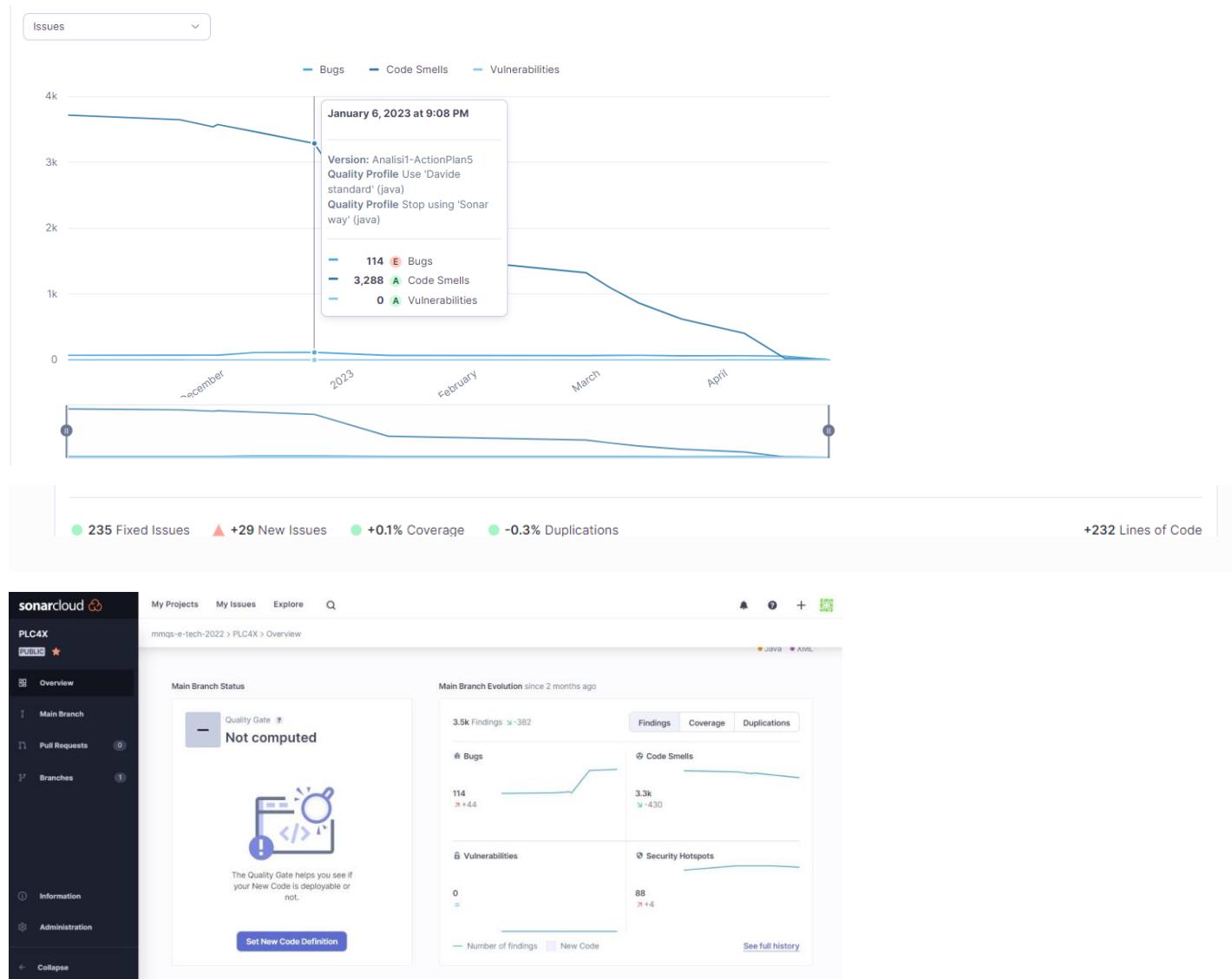


Grafico dell'actionPlan5



ActionPlan6

Al termine dell'ActionPlan5, rieseguendo l'analisi di SonarCloud, i bugs sono aumentati di 2, però i Code Smells sono stati risolti. In questa fase il codice totale del progetto è stato dimezzato a cause di alcune scelte progettuali al fine della consegna del caso di studio. L'operazione è stata approvata e convalidata dal docente per tramite del suo assistente.



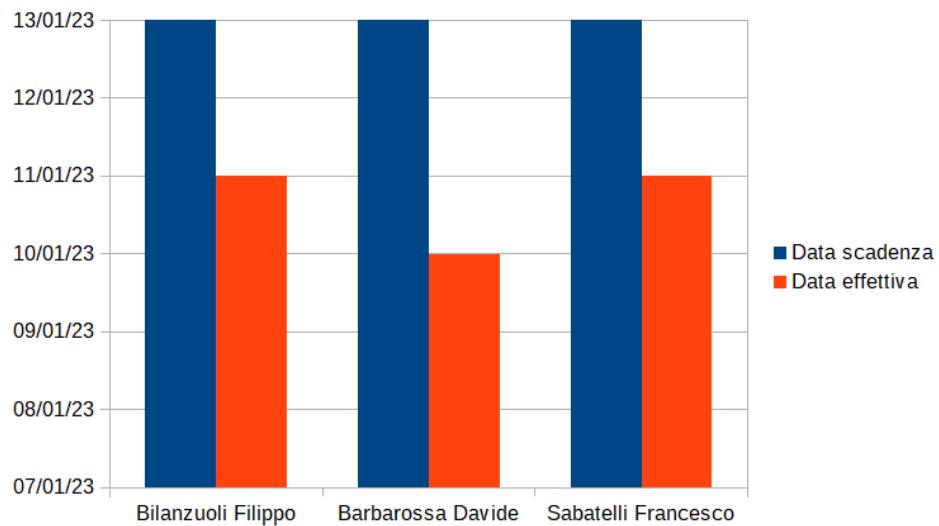
Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 code smells di diverso tipo che trovavamo nel file Analisi6.csv.

Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan6.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 a testa.

L'estimated time di questo ActionPlan è di 45.50 ore; invece, lo spent time è di 25.00 ore.

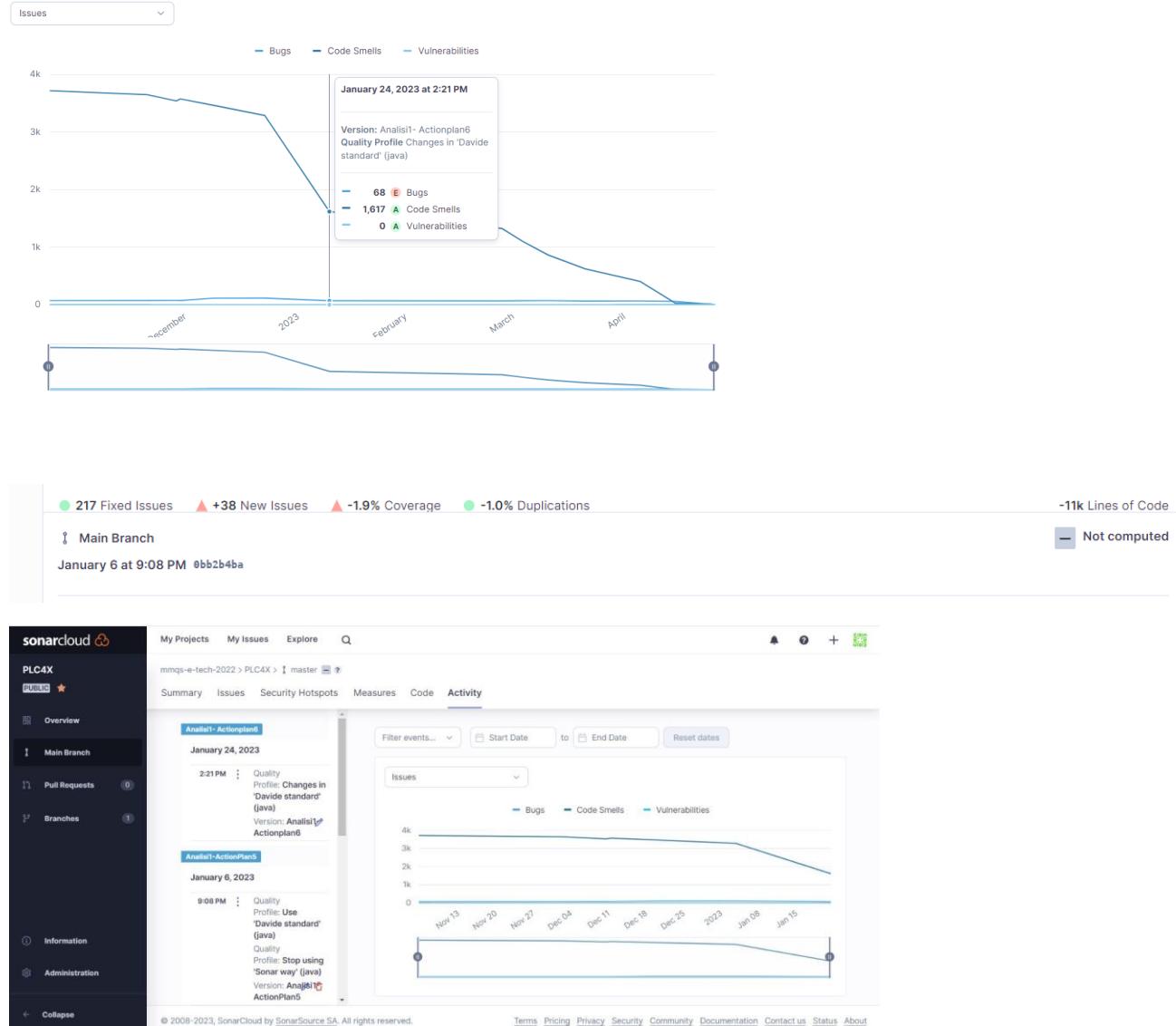
Questo ActionPlan aveva come data di partenza il 07/01/2023 e come data di scadenza il 13/01/2023, ed è stato chiuso il 11/01/2023, con due giorni di anticipo.

Nelle pagine seguenti riportiamo il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan7

Al termine dell'ActionPlan6, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più Issues di quelle svolte durante l'ActionPlan, con però altre 38 che si erano generate.



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 code smells di diverso tipo che trovavamo nel file Analisi7.csv.

Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan7.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 a testa.

L'estimated time di questo ActionPlan è di circa 17.30 ore; invece, lo spent time è di circa 10.00 ore.

Questo ActionPlan aveva come data di partenza il 16/01/2023 e come data di scadenza il 22/01/2023, ed è stato chiuso il 24/01/2023, in ritardo di due giorni.

Gantt di progetto:

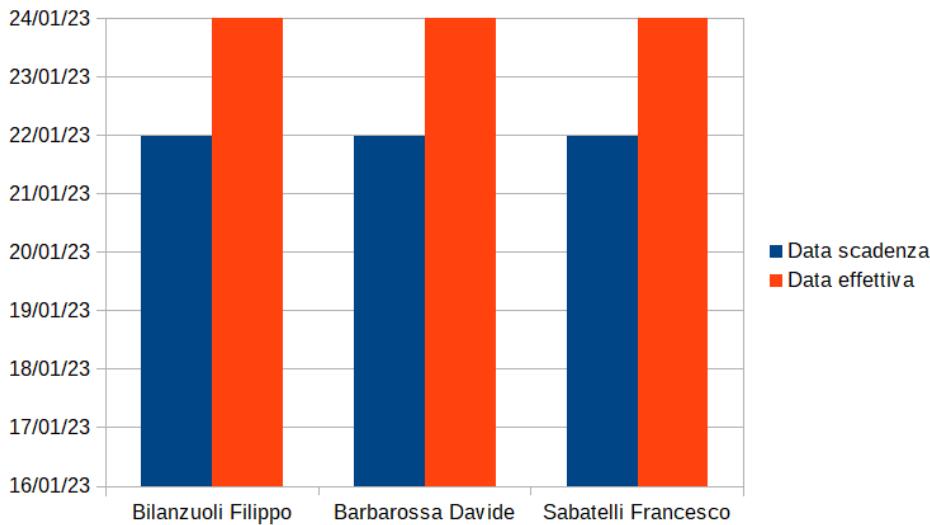


java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment
java:S1135: "Complete the task associated to this TODO comment

Gantt esecutivo:

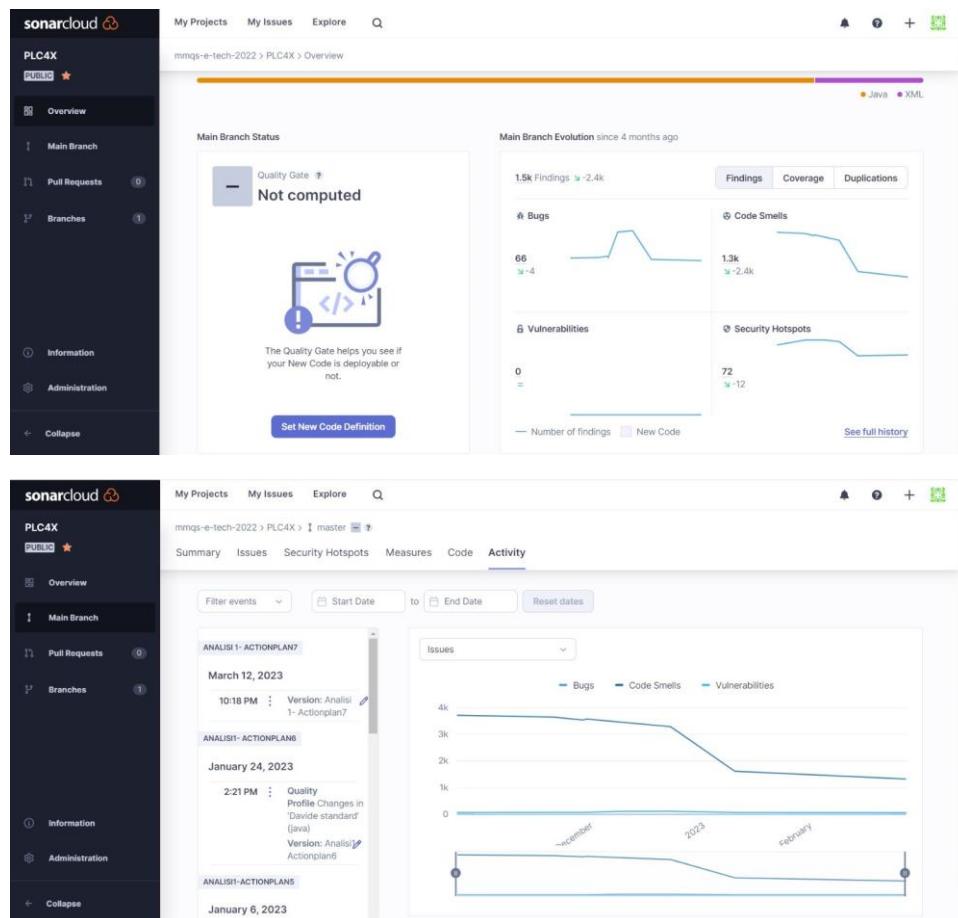


Nelle pagine seguenti abbiamo inserito il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan8

Al termine dell'ActionPlan7, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più Issues di quelle svolte durante l'ActionPlan, con però altre 38 che si erano generate.



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 code smells che trovavamo nel file Analisi8.csv.

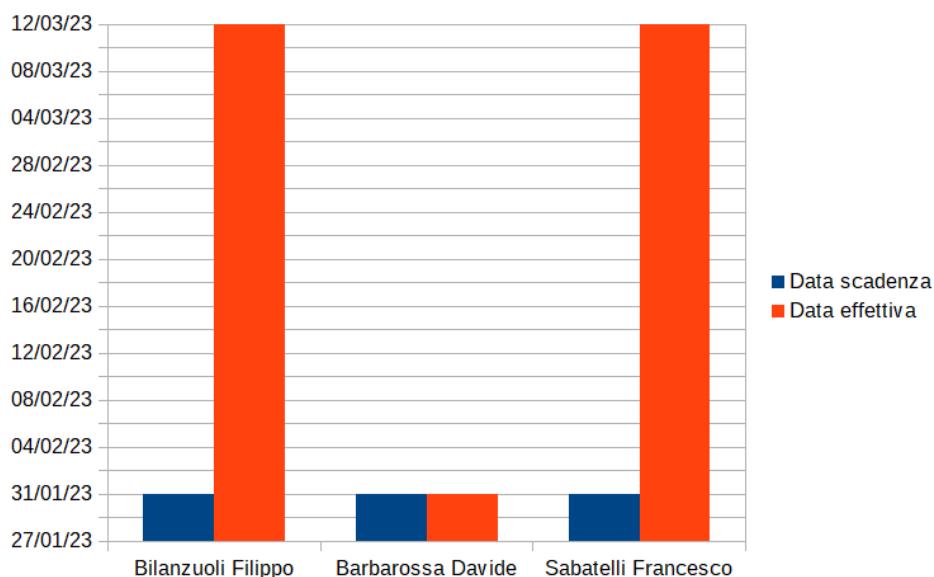
Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan8.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 a testa.

L'estimated time di questo ActionPlan è di circa 31.44 ore; invece, lo spent time è di circa 22.30 ore.

Questo ActionPlan aveva come data di partenza il 27/01/2023 e come data di scadenza il 31/01/2023, ed è stato chiuso il 12/03/2023, ci è stato un ritardo di due mesi, per diversi motivi a livello universitario.

[Il gantt completo dell'Action Plan 8 si trova qui](#)

Nella pagina seguente abbiamo riportato il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan9

Al termine dell'ActionPlan8, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più Issues di quelle svolte durante l'ActionPlan, con però altre 26 che si erano generate.

SonarCloud Project Overview

PLC4X (PUBLIC)

My Projects, **My Issues**, **Explore**, **Activity**

ANALISI1-ACTIONPLAN8 (March 18, 2023)

- Profile Changes in 'Davide standard' (java)
- Version: Analisi1-ACTIONPLAN8

ANALISI1-ACTIONPLAN7 (March 12, 2023)

- Version: Analisi1-ACTIONPLAN7

ANALISI1-ACTIONPLAN6

Issues: Bugs (blue), Code Smells (green), Vulnerabilities (red)

© 2008-2023, SonarCloud by SonarSource SA. All rights reserved.

SonarCloud Project Overview

PLC4X (PUBLIC)

Main Branch Status: Not computed

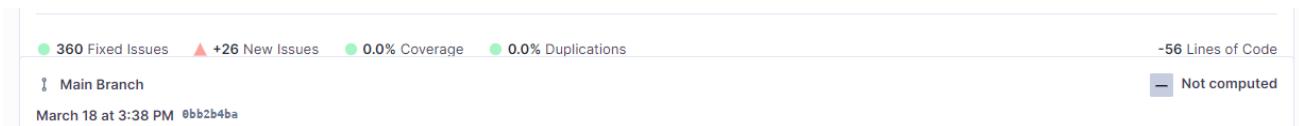
The Quality Gate helps you see if your New Code is deployable or not.

Main Branch Evolution since 4 months ago

Category	Findings	Coverage	Duplications
Bugs	69	-1	1.2k
Code Smells	1.1k	-2.6k	1.2k
Vulnerabilities	0	=	0
Security Hotspots	72	-12	0

Number of findings vs. New Code

See full history



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 code smells che trovavamo nel file Analisi9.csv.

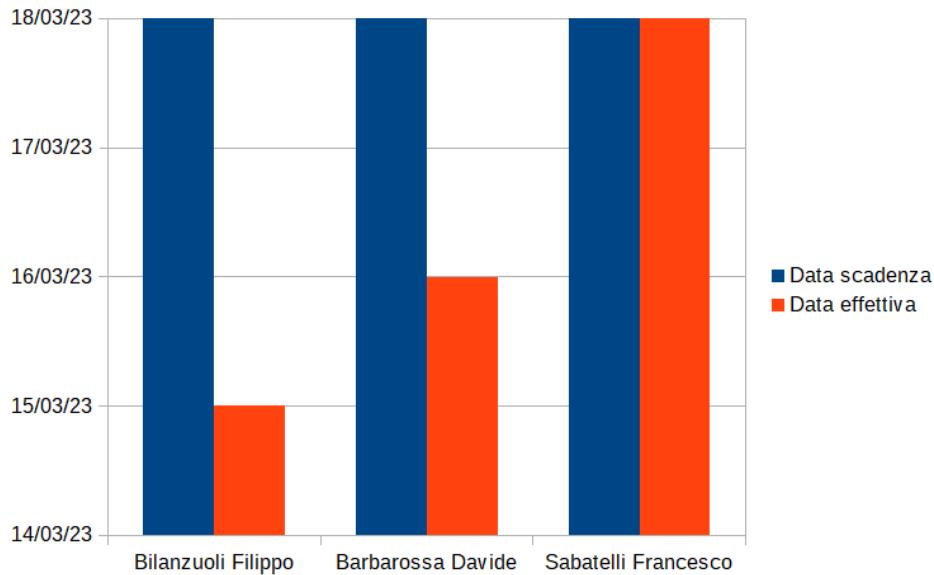
Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan9.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 a testa.

L'estimated time di questo ActionPlan è di 31.44 ore; invece, lo spent time è di 22.20 ore.

Questo ActionPlan aveva come data di partenza il 14/03/2023 e come data di scadenza il 18/03/2023, ed è stato chiuso il 18/03/2023, rispettando la data di scadenza.

[Il gantt completo dell'Action Plan 9 si trova qui](#)

Nelle pagine seguenti ci sono gli screen delle versioni del Gantt di Progetto e del Gantt Esecutivo ed in coda un grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan10

Al termine dell'ActionPlan9, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più issues di quelle svolte durante l'ActionPlan, con però altre 31 che si erano generate.



PLC4X

Issues

March 25, 2023 at 2:05 PM

Version: Analisi1-ActionPlan9
Quality Profile Changes in 'Davide standard' (java)

- 70 Bugs
- 865 Code Smells
- 0 Vulnerabilities

March 18, 2023 at 3:38 PM

Quality Profile Changes in 'Davide standard' (java)

PLC4X

Main Branch Status

Not computed

The Quality Gate helps you see if your New Code is deployable or not.

Main Branch Evolution since 5 months ago

1k Findings -2.9k

Category	Value	Change
Bugs	70	=
Code Smells	865	-2.9k
Vulnerabilities	0	=
Security Hotspots	72	-12

383 Fixed Issues +31 New Issues 0.0% Coverage -0.1% Duplications +32 Lines of Code

Main Branch Not computed

March 25 at 2:05 PM 0bb2b4ba

Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 Code smells che trovavamo nel file Analisi10.csv.

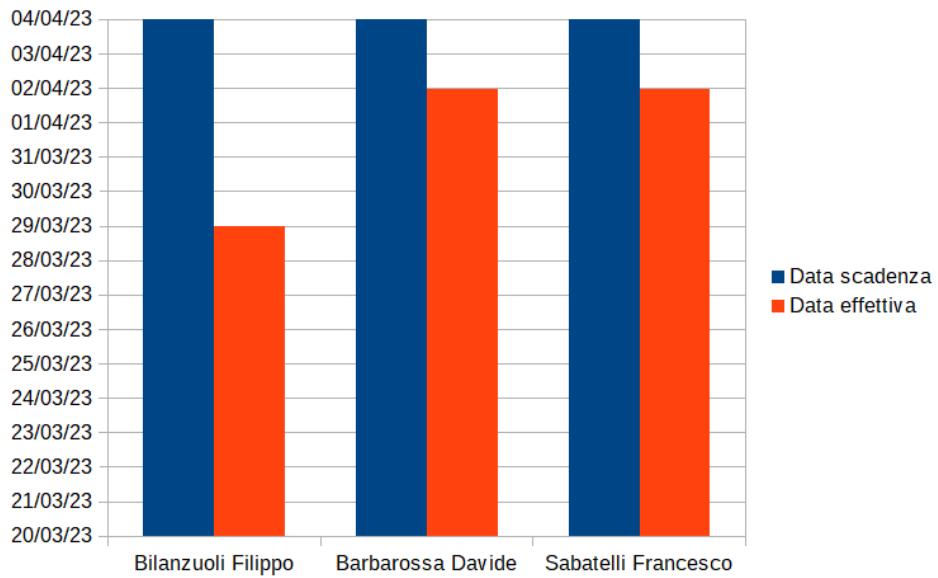
Dopo aver selezionato i Code smells e copiati in un altro file chiamato ActionPlan10.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 a testa.

L'estimated time di questo ActionPlan è di circa 45.47 ore; invece, lo spent time è di 15.24 ore.

Questo ActionPlan aveva come data di partenza il 20/03/2023 e come data di scadenza il 04/04/2023, ed è stato chiuso il 04/04/2023, è stata rispettata la data di scadenza.

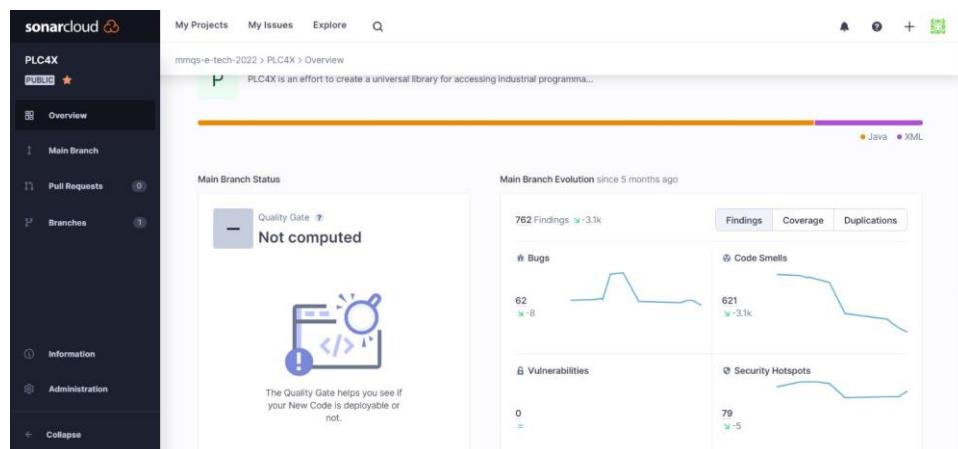
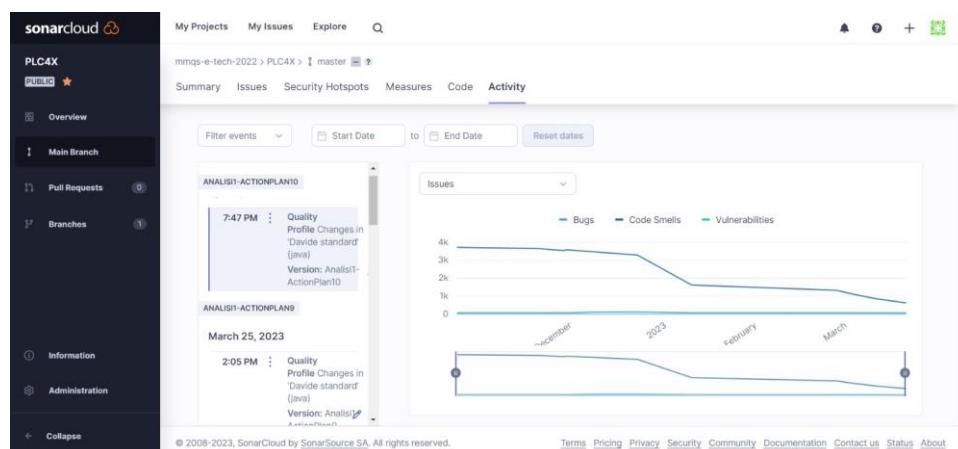
Il gantt completo dell'Action Plan 10 si trova [qui](#).

Nelle pagine seguenti ci sono gli screen delle versioni del Gantt di Progetto e del Gantt Esecutivo ed in coda un grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan11

Al termine dell'ActionPlan10, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più issues di quelle svolte durante l'ActionPlan, con però altre 65 che si erano generate.





Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i primi 300 code Smells che trovavamo nel file Analisi11.csv.

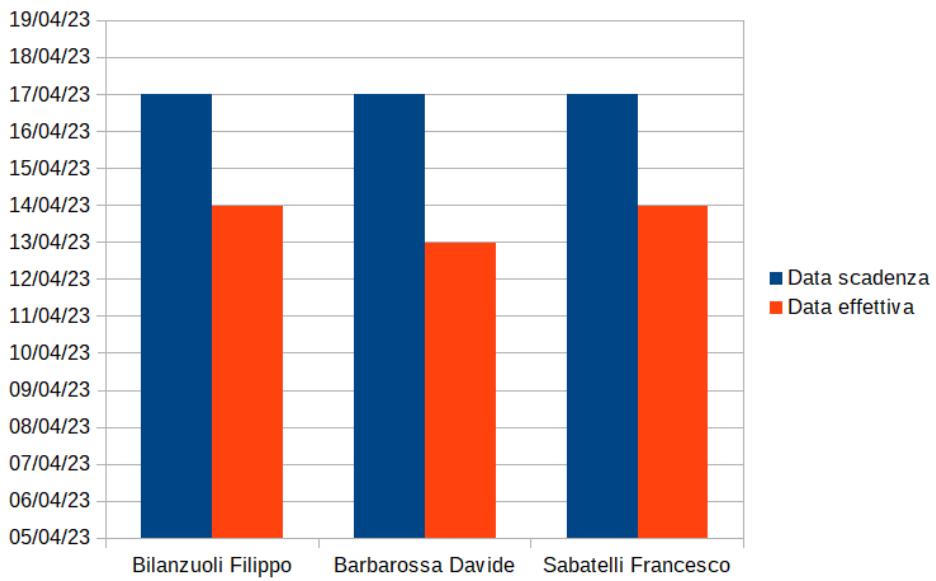
Dopo aver selezionato i Code Smells e copiati in un altro file chiamato ActionPlan11.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 100 ad ogni membro.

L'estimated time di questo ActionPlan è di circa 70.00 ore; invece, lo spent time è di circa 20.30 ore.

Questo ActionPlan aveva come data di partenza il 05/04/2023 e come data di scadenza il 17/04/2023, ed è stato chiuso il 19/04/2023, con due giorni di ritardo rispetto alla data di scadenza.

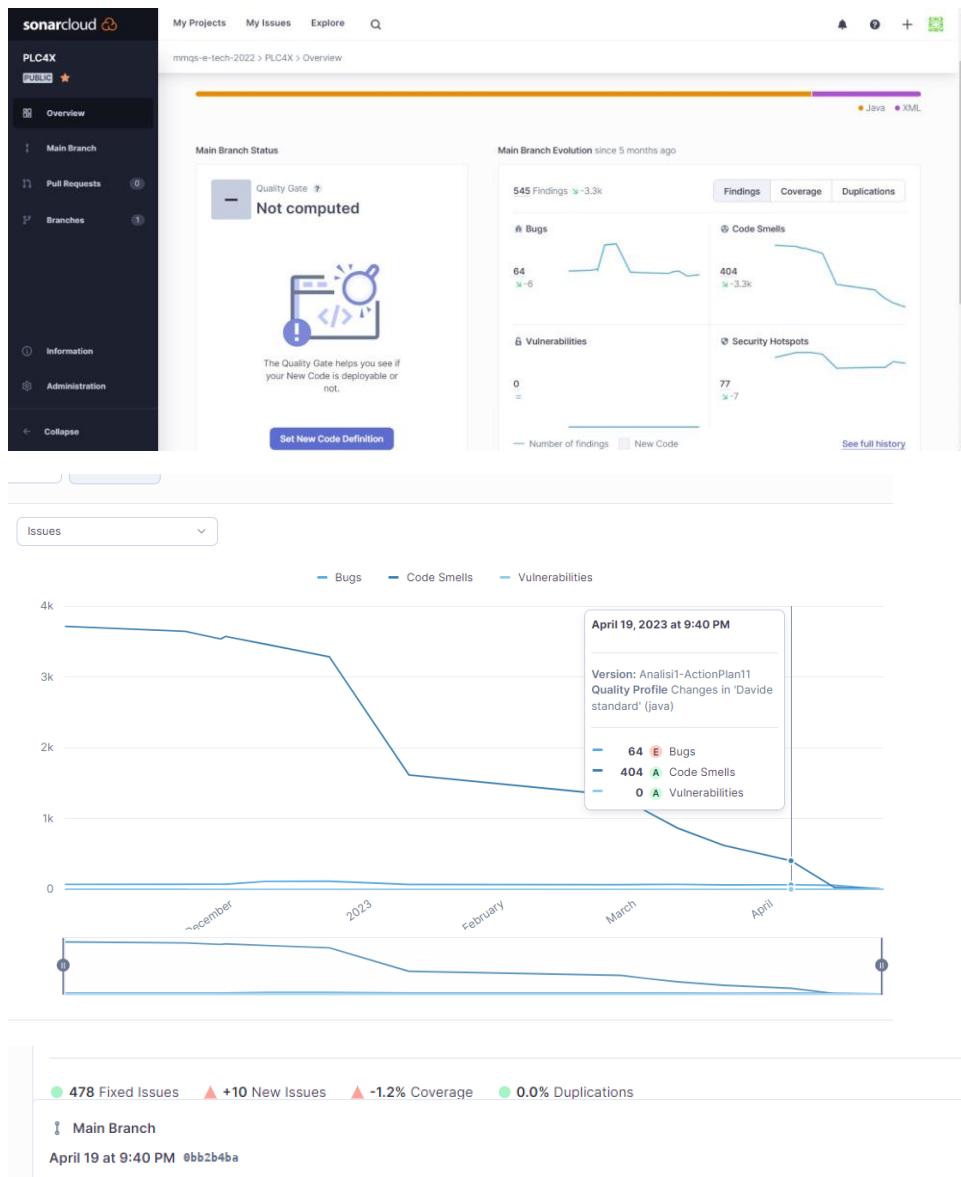
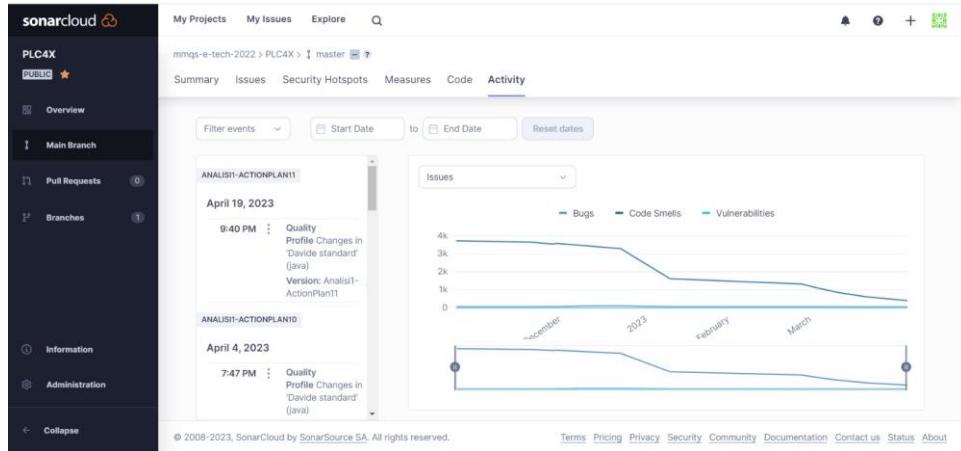
Il gantt completo dell'Action Plan 11 si trova [qui](#)

Nella pagina seguente abbiamo inserito il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan12

Al termine dell'ActionPlan11, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte più issues di quelle svolte durante, ma si erano generate altre 10.



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i restanti 405 code smells che trovavamo nel file Analisi12.csv.

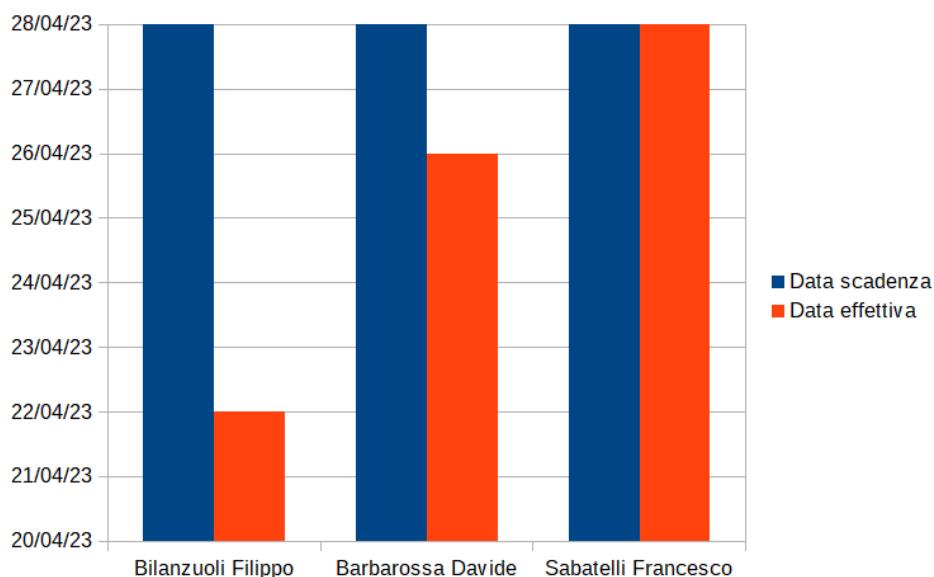
Dopo aver selezionato i Code Smells e copiati in un altro file chiamato ActionPlan12.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 135 per ogni membro.

L'estimated time di questo ActionPlan è di circa 25.48 ore; invece, lo spent time è di circa 10.48 ore.

Questo ActionPlan aveva come data di partenza il 20/04/2023 e come data di scadenza il 28/04/2023, ed è stato chiuso il 28/04/2023, in tempo con la data di scadenza.

Il gantt completo dell'Action Plan 12 si trova [qui](#)

Nella pagina seguente abbiamo inserito il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan13

Al termine dell'ActionPlan12, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte tutte le issues svolte durante l'ActionPlan ed anche i 7 bugs, ma se ne erano generate altre 8.



sonarcloud

PLC4X PUBLIC

Overview Main Branch Pull Requests 0 Branches 1 Information Administration

My Projects My Issues Explore Q

mmosi-e-tech-2022 > PLC4X > master

Summary Issues Security Hotspots Measures Code Activity

Filter events Start Date to End Date Reset dates

ANALYSIS1-ACTIONPLAN12 2:38 PM

Quality Profile Stop using 'Sonar way' (xml)
Quality Profile Changes in 'Davide standard' (java)
Quality Profile Use 'Davide standard' (xml)
Version: Analysis1-ActionPlan12

ANALYSIS1-ACTIONPLAN11

ANALYSIS1-ACTIONPLAN12

Issues

Bugs Code Smells Vulnerabilities

April 29, 2023

Date	Bugs	Code Smells	Vulnerabilities
December	~3500	~100	~10
January	~3200	~100	~10
February	~1800	~100	~10
March	~1200	~100	~10
April	~50	~10	~10

sonarcloud

PLC4X PUBLIC

Overview Main Branch Pull Requests 0 Branches 1 Information Administration

My Projects My Issues Explore Q

mmosi-e-tech-2022 > PLC4X > Overview

Main Branch Status

Quality Gate Not computed

The Quality Gate helps you see if your New Code is deployable or not.

Set New Code Definition

Main Branch Evolution since 6 months ago

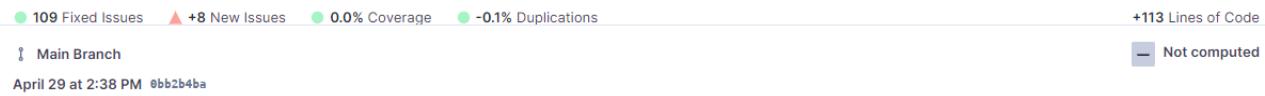
158 Findings ~3.7k

Findings Coverage Duplications

Bugs: 57 (~13) Code Smells: 25 (~3.7k)

Vulnerabilities: 0 Security Hotspots: 78 (~8)

Number of findings New Code See full history



Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i 57 bugs, i 25 Code Smells e gli 80 Security Hotspots che trovavamo nel file Analisi13.csv, concludendo la risoluzione di questo tipo di issues.

Dopo aver selezionato i bugs, i Code Smells e i Security Hotspots e copiati in tre file diversi chiamati ActionPlan13.csv, ActionPlan13(Bugs).csv e ActionPlan13(Security Hotspots).csv. il Project Manager li ha importati su Redmine e ne ha assegnati 54 per ogni membro.

L'estimated time di questo ActionPlan è di circa 18.56 ore; invece, lo spent time è di 4.15 ore.

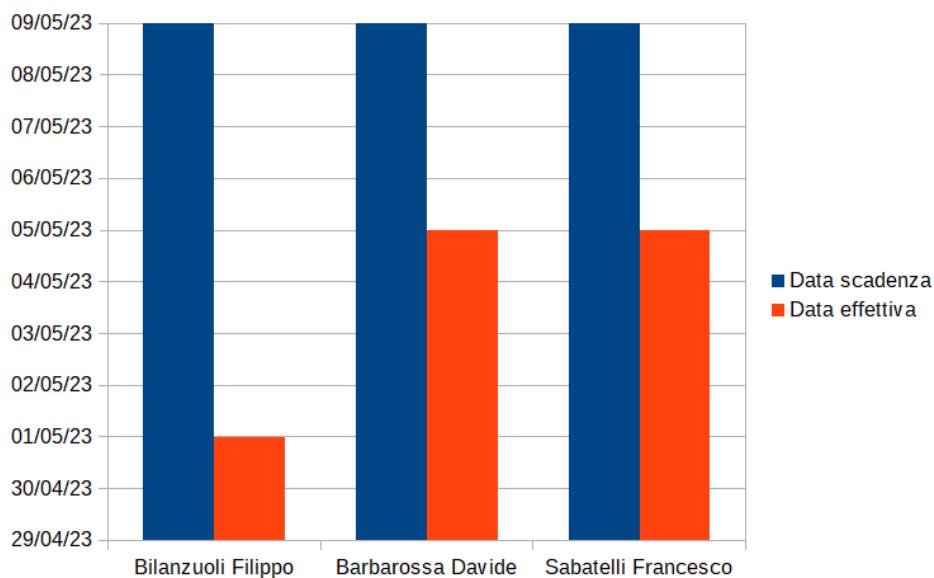
Questo ActionPlan aveva come data di partenza il 29/04/2023 e come data di scadenza il 09/05/2023, ed è stato chiuso il 05/05/2023, in anticipo di 4 giorni.

[Il gantt completo dell'Action Plan 13 \(Code smells\) si trova qui](#)

[Il gantt completo dell'Action Plan 13 \(Security Hotspot\) si trova qui](#)

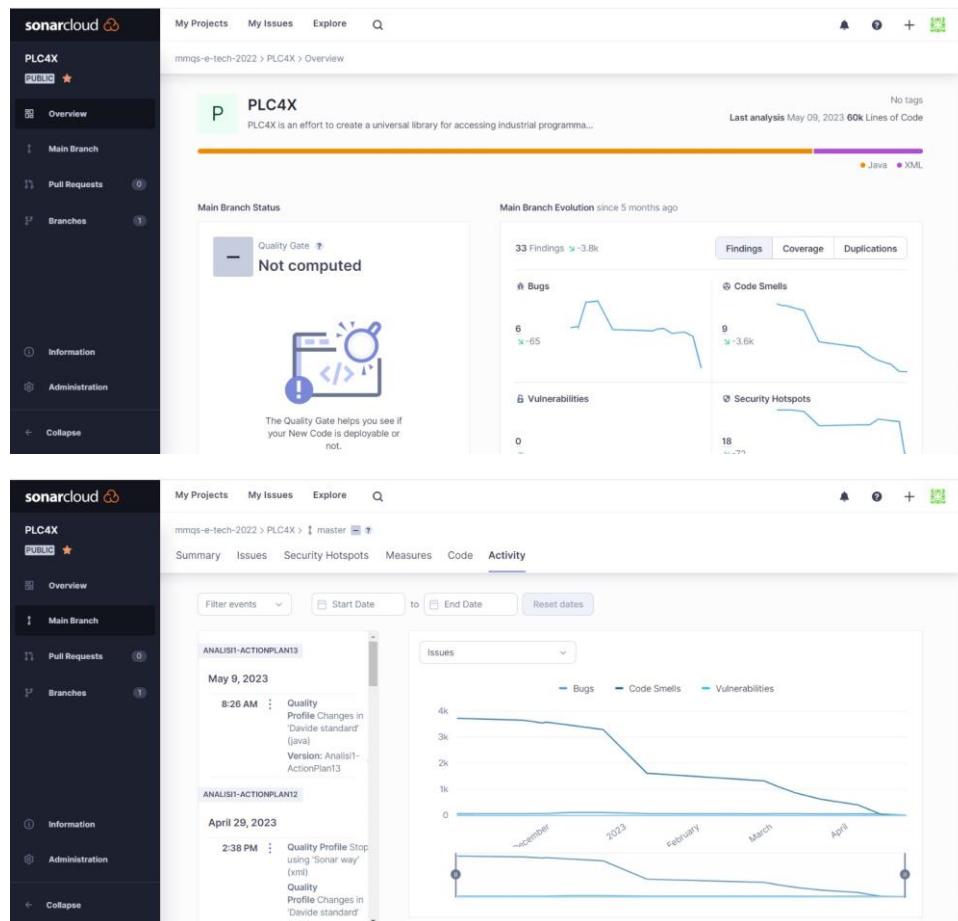
[Il gantt completo dell'Action Plan 13 \(Bugs\) si trova qui](#)

Nella pagina seguente abbiamo inserito il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:



ActionPlan14

Al termine dell'ActionPlan13, rieseguendo l'analisi di SonarCloud, abbiamo notato che si erano risolte tutte le issues svolte durante l'ActionPlan e se ne erano generate altre 8.





Il target che ci siamo fissati per questo ActionPlan è stato quello di risolvere i 9 code smells e i 6 bugs che trovavamo nel file Analisi14.csv, concludendo la risoluzione di tutte le Issues.

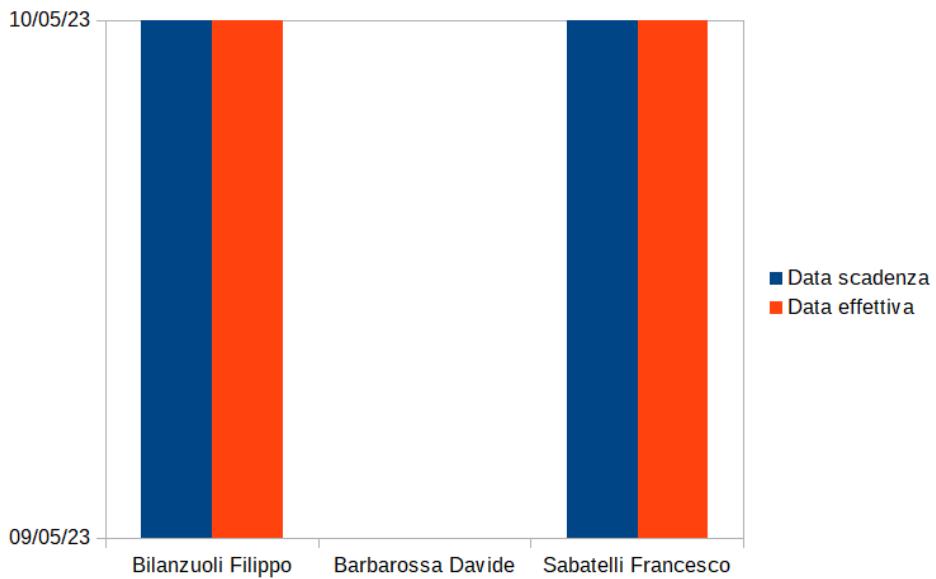
Dopo aver selezionato i bugs e i Code Smells e copiati in un altro file chiamato ActionPlan14.csv, il Project Manager li ha importati su Redmine e ne ha assegnati 7 per un membro e 8 l'altro membro. Il terzo membro ha verificato solamente il funzionamento del progetto e si è occupato dell'analisi finale.

L'estimated time di questo ActionPlan è di circa 2.17 ore; invece, lo spent time è di circa 1.47 ore.

Questo ActionPlan aveva come data di partenza il 09/05/2023 e come data di scadenza il 10/05/2023, ed è stato chiuso il 10/05/2023, in tempo con la data di scadenza.

Il gantt completo dell'Action Plan 14 si trova [qui](#)

Nella pagina seguente abbiamo inserito il grafico raffigurante la data di scadenza dell'ActionPlan e la data di fine svolgimento per ogni componente:

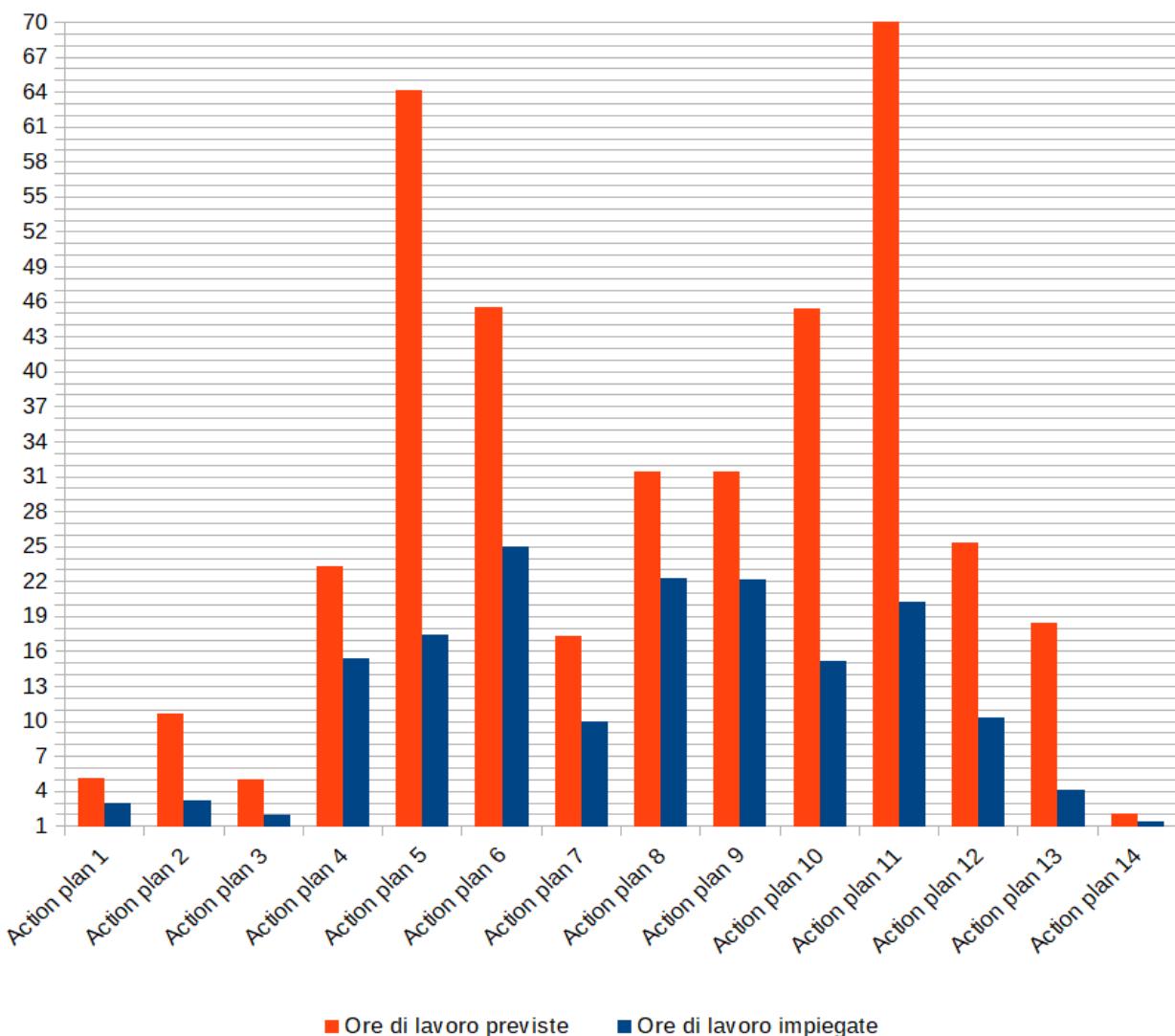


Parte Finale SonarCloud

Al termine dell'ActionPlan14, rieseguendo l'analisi di SonarCloud, abbiamo notato che tutte le issue svolte si erano risolte il che ha voluto dire che abbiamo raggiunto il target prefissato per l'analisi di SonarCloud.

Così è come appare attualmente la dashboard di SonarCloud del nostro progetto:

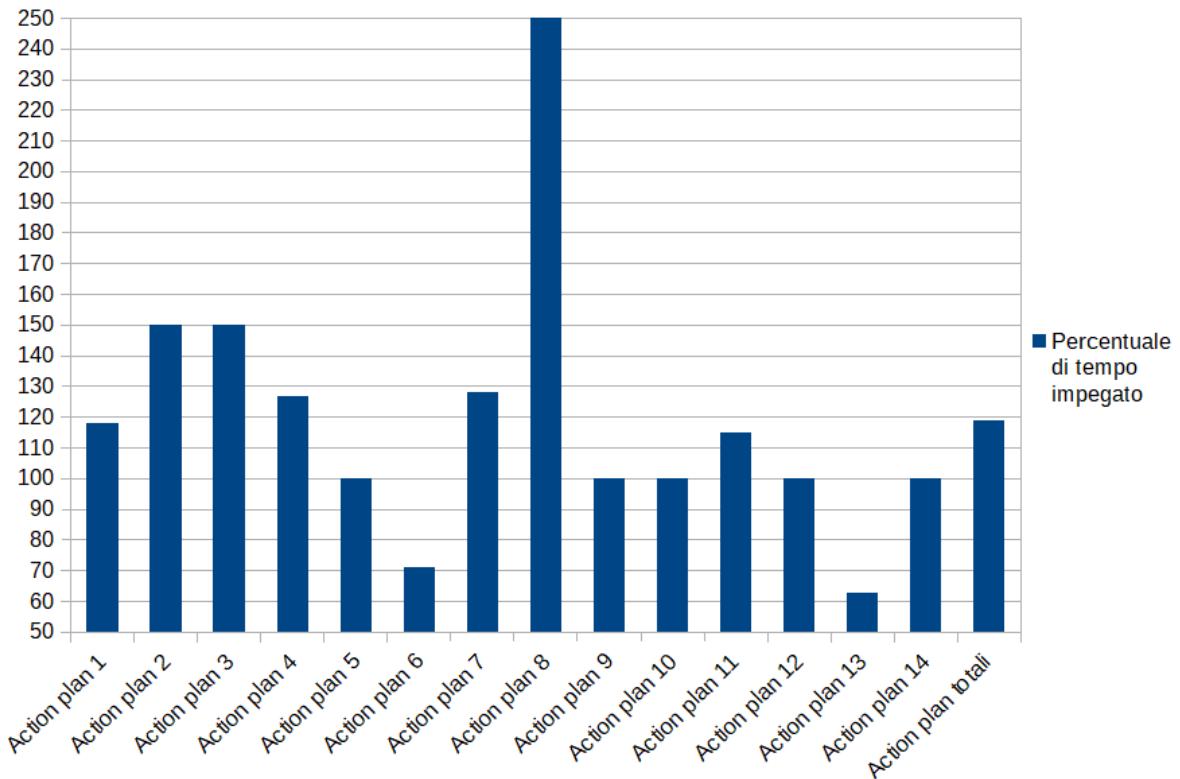
Alla fine della parte di SonarCloud possiamo stimare quanto tempo abbiamo impiegato realmente per svolgere i vari ActionPlan rispetto al tempo stimato (questo non comprende i Security Hotspots, perché non si ha un tempo stimato).



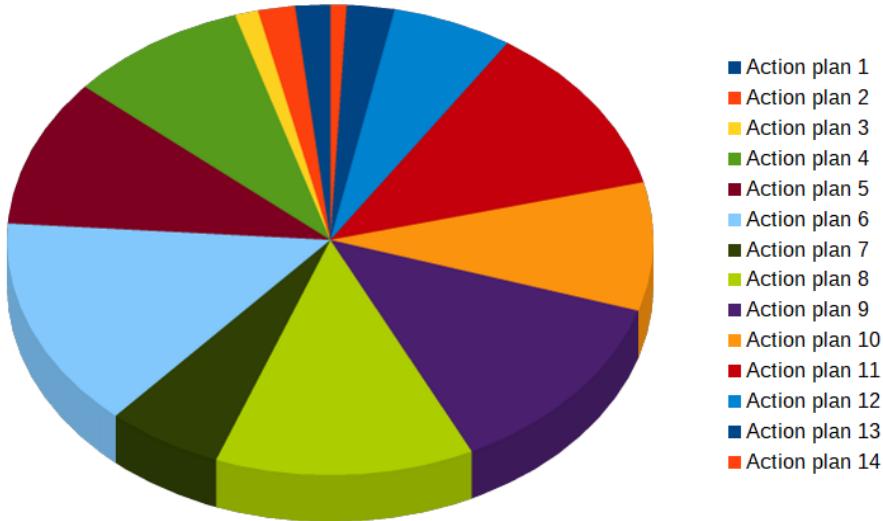
Il tempo stimato per le issue di questi ActionPlan in totale è stato di 364.26 ore, mentre quello impiegato è di 160.39 ore.

Leggenda:

- 100% ==> (data di consegna prevista – data di creazione)
- Altezza > 100% = completato in ritardo
- Altezza < 100% = completato in anticipo



Dal grafico precedente si può notare la presenza di alcuni action plan che superano la data di consegna prevista. La maggior parte dei ritardi derivano da problemi con la piattaforma redmine o eventuali incidenti di percorso.



ActionPlan1_Fortify

A causa del disservizio di fortify, per raggiungere il nostro target, abbiamo basato il nostro lavoro sulla prima scansione effettuata. Di seguito riportiamo i dati della scansione:

- 16.000 LOW
- 78 HIGH
- 0 MEDIUM
- 0 CRITICAL

Category	Primary Location	Analysis Type	Criticality	Tagged
Poor Style: Value Never Read	BACnetConstructedDataFileAll.java: 82	SCA	Low	
Poor Style: Value Never Read	ModbusAsciiADU.java: 105	SCA	Low	
J2EE Bad Practices: Sockets	TcpHexDumper.java: 63	SCA	Low	
Poor Style: Value Never Read	AdsReadDeviceInfoResponse.java: 159	SCA	Low	
Poor Style: Value Never Read	DoubleComplexNumberType.java: 87	SCA	Low	
Poor Style: Value Never Read	BACnetPriorityValueTime.java: 94	SCA	Low	
Poor Style: Value Never Read	BACnetConstructedDataAccessUserAll.java: 68	SCA	Low	

A causa del disservizio, non abbiamo potuto verificare se al seguito del lavoro svolto con la piattaforma di SonarCloud, le issus fossero state risolte dai precedenti actionPlan.

Quindi in questa fase ci siamo dedicati solamente a convalidare le issus segnalate nella precedente scansione, che in realtà sono state risolte. Una volta raggiunto il numero di issus previste dal target, ovvero:

- 0 CRITICAL

- 0 MEDIUM
- 53 HIGH
- 16K - 250 LOW

Il tempo impiegato per lo svolgimento di questo actionPlan è stato di circa 10 ore.

Note finali

Standard ignorati ritenuti inutili o poco pertinenti al miglioramento effettivo del codice (leggibilità, efficacia, efficienza, manutenibilità):

- java S110
- java S107
- java S1452
- java S2326
- java S1128
- java S1874
- java S3740
- java S4276
- java S6397
- java S100
- java S1854
- java S138
- java S135
- java S1604
- java S2093
- java S2864
- java S6355
- java S3824
- java S5976
- java S128
- java S3457
- java S3252
- java S1612
- java S3824
- java S1872
- java S2201
- java S5443
- java S2925
- java S3958
- java S5998

Di seguito riportiamo i nomi delle cartelle di progetto che abbiamo dovuto tagliare:

- bacnet
- c-bus
- opocua
- S7

Conclusioni

A progetto ultimato possiamo quindi dire che l'esecuzione dello stesso è stata tutto sommato agevole per quasi tutte le Issues svolte, le quali, infatti, sono state quasi sempre risolte con largo anticipo rispetto al tempo stimato.

Il lavoro di gruppo non ha riscontrato particolari inconvenienti anche grazie alla facilità d'uso dei software messi a disposizione, salvo qualche "malfunzionamento" temporaneo che ha ostacolato il lavoro.

In conclusione, il progetto svolto ci ha permesso di approfondire la conoscenza delle metodologie e degli strumenti utilizzati per garantire la qualità del software. Ci siamo trovati davanti a diverse problematiche legate alla qualità del codice e di conseguenza applicato nuovi metodi per migliorarne la manutenibilità e la leggibilità.

Il progetto ci ha insegnato l'importanza di una costante attenzione alla qualità del codice in tutte le fasi dello sviluppo e siamo sicuri che quest'esperienza ci sarà utile nella nostra carriera professionale e ci aiuterà a creare software di qualità sempre migliore.