

# Relazione Ingegneria del Software 2

## *Machine Learning for Software Engineering*

### Deliverable 1

Camilli Michela 0286047

January 27, 2022

## Introduzione

Descrizione del problema

## Progettazione

Reperimento ticket da Jira

Commit Date

Calcolo #FixedTickets

## Risultati ottenuti

Process Control Chart

Link GitHub e SonarCloud

- Scopo del progetto è stato quello di realizzare una prima analisi di un sistema Software, tramite l'ausilio di sistemi di Version Control ed Issue Tracking.
- Il progetto preso in esame è stato *Apache C++ Standard Library*, i cui ticket sono stati esaminati grazie all'utilizzo di *JIRA*.
- Dalla repository github di tale progetto, tramite comandi *git*, sono state estratte informazioni utili all'analisi del sistema. Grazie a queste è stato elaborato il relativo *Process Control Chart*.

Sono stati, dapprima, ricavati tutti i *fixed tickets* del progetto, tramite query JIRA, andando a filtrare su quei ticket (o issue) aventi status "closed" o "resolved".

```
{
  "expand": "schema,names",
  "startAt": 0,
  "maxResults": 50,
  "total": 606,
  "issues": [
    {
      "expand": "operations,versionedRepresentations,editmeta,changelog,renderedFields",
      "id": "12611158",
      "self": "https://issues.apache.org/jira/rest/api/2/issue/12611158",
      "key": "STDCXX-1073"
    },
    {
      "expand": "operations,versionedRepresentations,editmeta,changelog,renderedFields",
      "id": "12609529",
      "self": "https://issues.apache.org/jira/rest/api/2/issue/12609529",
      "key": "STDCXX-1072"
    }
  ]
}
```

- Sono stati considerati i commit relativi a questi ticket, andando a ricavare la committer date di ciascuno di essi, tramite comando *git log -grep*.
- Per ciascun ticket è stata considerata la committer date più recente, essendo la data di risoluzione effettiva del ticket.

Si è quindi proceduto con il calcolo del numero di Fixed Tickets. Tale computazione è stata effettuata all'interno di uno specifico range temporale, che va dal primo all'ultimo mese di commit sui ticket del progetto considerato.

Sono stati, quindi, sommati i Fixed Ticket relativi ad un mese nell'intervallo. I risultati ottenuti sono stati del tipo:

| Date     | NumFixed |
|----------|----------|
| 2006 Feb | 8        |
| 2006 Mar | 7        |
| 2006 Apr | 2        |
| 2006 May | 2        |

Da cui è stato, infine, graficato il relativo Process Control Chart.

Il Process Control Chart permette di valutare la stabilità degli eventi nel tempo e, quindi, di andare a constatare se un certo processo si trovi o meno in uno stato di controllo. A seguito dell'individuazione di specifici limiti, se il grafico supera in determinati punti queste soglie, si possono, in particolare, fare sia analisi retrospettive in merito agli errori riscontrati, sia considerazioni predittive sui possibili esiti del processo.

Per il calcolo dell'*Upper* e *Lower Control Limit* sono state utilizzate le formule:

$$\text{Upper limit} = \text{mean} + 3 * \text{STDV}$$

$$\text{Lower limit} = \text{mean} - 3 * \text{STDV}$$

a seguito della computazione della media e della deviazione standard. Per quanto concerne, invece, i dati sugli assi, come si evince dalla Figura 1, sulla x stato è posto il tempo in termini di *Mesi*, mentre, sulla y il *numero di Fixed Tickets* complessivo.



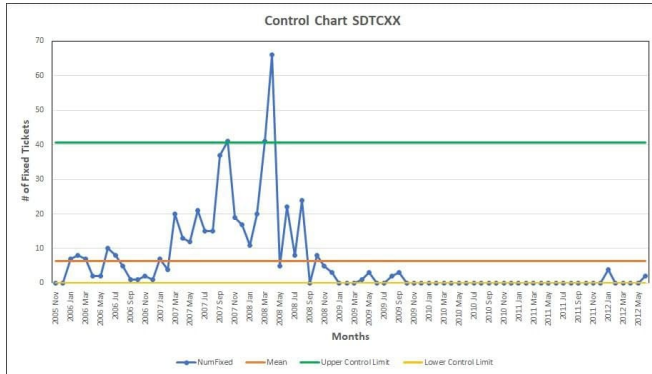


Figure: Process Control Chart di SDTCXX

- Nella realizzazione del grafico mostrato, si è notato che il *Lower Control Limit*, poiché la deviazione standard risulta maggiore della media, assume valori negativi. Dato che, però, risulta illogico un tale comportamento dei fixed tickets, si è scelto di porre tale limite a 0.
- Dall'analisi del grafico si può, quindi, notare come vi siano punti in cui il numero di Fixed Tickets supera la soglia massima consentita. Questo evidenzia la presenza di problemi nella gestione dei ticket in questi periodi.
- Si può notare che, nei mesi finali dello sviluppo del progetto, vi è una netta diminuzione del numero di Fixed Tickets.

Di seguito sono riportati i link della repository Github del progetto e dell'analisi dello stesso su SonarCloud.

- Repository GitHub
- Analisi Sonar