Scelte fatte per l’analisi statica e dinamica

Durante lo Sprint 2, ci siamo concentrati sull'analisi delle possibili soluzioni per l'analisi statica e dinamica del codice. Per tracciare e organizzare il lavoro, abbiamo aperto due issue su GitHub, rispettivamente l’Issue #13 e l’Issue #14.

A supporto di queste attività, abbiamo inoltre prodotto due documenti dettagliati: uno dedicato all'**analisi statica del codice**, in cui vengono approfondite le metodologie e gli strumenti per individuare potenziali problemi senza eseguire il programma, e un altro sull'**analisi dinamica del codice**, che invece esamina il comportamento del software durante l'esecuzione.

A seguito dello studio fatto, si è deciso di implementare le seguenti soluzioni per il contesto del progetto MS3:

- Per l’**analisi statica del codice**
  - SonarCloud
  - SpotBugs
  - PMD
  - ESLint
  - Lighthouse
- Per l’**analisi dinamica del codice**
  - OWASPZap

L’obiettivo dello **Sprint 3** era quindi quello di integrare i tool di analisi precedentemente individuati. Tuttavia, questa fase si è rivelata particolarmente impegnativa per il team, che ha dovuto affrontare diverse difficoltà tecniche.

Per quanto riguarda l’implementazione di **SonarCloud**, la principale difficoltà è stata quella di integrare il codice del frontend nell’analisi. Questo è stato dovuto a diversi fattori, tra cui la **limitata esperienza del team** nell’integrazione di questo strumento e la **scarsa modularità** del codice che portava alla necessità della scrittura di script specifici per riuscire a far partire l’analisi del codice frontend posto in una cartella separata.
Per questi motivi, ed anche perché il focus del progetto non è quello di impare ad utilizzare SonarCloud, si è deciso di cambiare strumento di analisi statica del codice, adottando **Codacy**. Questo strumento infatti non necessita di alcuno script per riconoscere i file nei quali è presente il codice, analizzandolo quindi tutto quanto il progetto. Inoltre, al suo interno integra già tutti quanti gli altri tool di analisi statica del codice. In particolare, i code patterns a cui il codice di MS3 è sottoposto ad analisi sono:

- Checkstyle (Java)
- ESLint (Javascript)
- Jackson Linter (JSON)
- markdownlint
- PMD (Java & Javascript)
- ShellCheck
- SpotBugs (Java)
- SQLint (SQL)

Per l’analisi dinamica invece si è provato ad integrare questa all’interno delle GitHub Actions. Tuttavia questo compito si è rivelato esser molto più complesso del previsto a causa della complessità del progetto e non avendo un sito di deploy sempre attivo su cui poter lanciare l’analisi. Per questa ragione la soluzione adottata è stata quella di condurre l’analisi dinamica tramite il tool scaricabile di OWASP Zap, ed utilizzandolo manualmente.