SSV - Security Smells Visualizer

Andrea Malnati
Roberto Negro

March 28, 2024

Contents

1 Introduction

SSV (Security Smells Visualizer) is a software crafted with the objective of providing a graphical interface for displaying the results of security smells analysis on microservices applications. We have opted to adopt the Unified Process (UP), breaking down the phases into multiple iterations.

2 Inception

During the inception phase, which spanned a single iteration of three days, we analyzed the most critical functional and non-functional requirements, examined use cases, and produced both the Glossary and the Use Case Diagram.

2.1 Analysis

Requirements

Functional requirements

- The user can upload the analysis results from KubeHound in .txt format.
- The user have to specify the relevance of the microservices
- The user can choose an analysis from his previous analysisis.
- The user can visualize the security smells detected with an urgency code.
- The user can visualize the respective proposed refactoring for each smell.
- The user can add manually a security smell.

Non functional requirements

- The system's data must be saved persistently in a local database.
- At the start of the application, a local server must be available at local host 8080.
- The system must support .txt format for input files.

Glossary

Triage:

Refactoring:

Use Cases

Use Case UC1: New analysis upload

Primary actor: User

Table 1: Main scenario

Step	Action
1	User selects ”New analysis” button.
2	User uploads a txt file that contains analysis results.
(3)	*Continue in Use Case UC2: Insert microservices’s information*

Table 2: Alternative scenarios

Step	Action
2.1	User uploads a non-supported format for analysis. The system shows an error message.

Use Case UC2: Insert microservices’s information

Primary actor: User

Table 3: Main scenario

Step	Action
1	User inserts microservices’s informations.
(1)	*This step is repeated for each microservice*
2	User confirms his inputs.

3 Elaboration

4 Construction

5 Transition