Assessing and Improving the Quality of Docker Artifacts

«The quality assessment of Docker artifacts and approaches to solving associated issues could provide cutting-edge support for developers to produce high-quality Dockerfiles and, consequently, more reliable and performing Docker images»

Step 1. Improving the Quality of Dockerfiles

Goal

- Dockerfile **smells** are **common**, but not all of them are **relevant** to be addressed by developers. Smells are used in literature to measure Dockerfile quality
- Empirical evaluation of **smell survivability**
- Evaluation of what smells developers are **willing to address**

Execution Plan

Dockerfile changes over time → Analyze literature and smell-fixing commits to identify **relevant Dockerfile smells** → Provide **fix recommendations** using a rule-based refactoring tool → Evaluating fixes via **pull requests** with an observation window of **three months**

Status

- Registered Report paper submitted and accepted to ICSME ‘22
- A preliminary version of the refactoring tool provides fixes for 6 of the 8 most common smells from the literature

Step 2. Quality Features that Influence Docker Images Adoption

Goal

- It is **not clear** how developers **measure** the **quality** of a Docker image. Evaluating smells is not enough
- Empirical evaluation of the **internally** and **externally** observable quality aspects of Docker images

Execution Plan

Evaluation of the quality features from the **literature** → Definition of **metrics** that **measure** quality features → Mining of the **most used** Docker images → Empirical analysis of the **relation** between quality features and adoptions

Status

- Some of the features related to image size and security negatively impact the adoption
- The “official image” label has a positive impact on the adoption of Docker images
- These are preliminary results as the work is currently under review

Step 3. Quality-Aware Generation of Docker Artifacts

Goal

- Lack of **advanced tools** to support developers during development
- Automatic generation of Dockerfiles matching the **quality model** resulting from the previous steps
- The generated artifacts must match **developers’ preferences**

Execution Plan

Natural language requirements and Dockerfile pairs → Approach to generate Dockerfiles → Evaluation of the generated artifacts

Status

- A preliminary experiment using deep-learning was conducted
- The results are promising, but more work is required to achieve our goal

Giovanni Rosa
STAKE Lab - University of Molise, Italy
giovanni.rosa@unimol.it
https://giovannirosa.com

Advisors: **Rocco Oliveto** and **Simone Scalabrino**

38th IEEE international Conference on Software Maintenance and Evolution
Doctoral Symposium

2-7 October, 2022 - Limassol, Cyprus

ICSME 2022