SVLC: Service virtualization into Linux Containers

Daniel Carvalho, A49419@alunos.isel.pt, 913764987

Supervisors:
Luis Osório, lo@isel.ipl.pt
Mário Pinheiro, mario.pinheiro@isel.pt

March 18, 2024

1   Introduction

The Linux container concept has gained an increasing level of adoption in the last decade, facilitated by the Docker product. The goal for this project is to develop a set of functionalities that facilitate the instantiation, management and deployment of applications or parts of applications (informatics systems of systems) in Linux containers and, as a more advanced objective, in the Kubernetes network. Due to the importance of open standards, the project will adopt open source initiatives, aligned with the Linux Foundation's Open Container Initiative (OCI).

2 Background Knowledge

In order to understand the reasons and objectives of this project, it is first necessary to know a little more about the themes that are covered in this project.

2.1 Linux Containers

A Linux container is an isolated set of 1 or more processes. All the necessary files are "provided from a distinct image, meaning that Linux containers are portable and consistent" in all stages of development and production. This also implies a quicker usage when compared with "development pipelines that rely on replicating traditional testing environments"[1].

2.2 Informatics System of Systems (ISoS)

The ISoS is a framework [2] that proposes an "integration model to establish a multi-supplier or multi-vendor technology landscape" [3], that works independently from the environment language. An ISoS is composed of one or more *ISystems* that comprises one or more *CES* (Cooperative Enabled Services), which in turn, contains one or more *Services*.

3 Conceptualization of the project

This project follows the idea of using ISoS as a starting point to facilitate the development and deployment of Isystems images. However, the objective is also to create features that make this process as autonomous as possible. This is achieved by implementing CI/CD (continuous integration and continuous delivery) practices that automate much or all of the manual human intervention traditionally needed to get new code from a commit into production, encompassing the build, test (including integration tests, unit tests, and regression tests), and deploy phases, as well as infrastructure provisioning"[4].

4 Technologies

For the development of this project we will be using technologies such as:

1. Podman/Podman-desktop: To manage and deploy containers[5];

2. Quarkus JIB or Eclipse JKube: To programmatically build optimized images for Java applications[6][7];

3. Kubernets: For distributed deployment, scaling, and management of containerized applications (container orchestrator)[8].

4. Visual Paradigm: To create conceptual models in sysml.

5. Vagrant: For creation and configuration of lightweight, reproducible, and portable development environments[9].
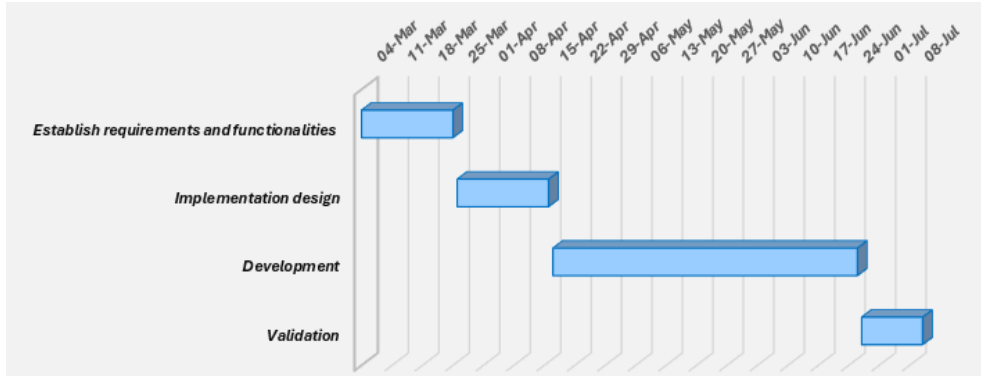
5 Schedule

Figure 1: Timeline of each phase of the project (in weeks).

References

[1] Red Hat. What's a linux container? `https://www.redhat.com/en/topics/containers/whats-a-linux-container`, 2022. [Online; accessed 14-March-2024].

[2] A. Osório, A. Belloum, Hamideh Afsarmanesh, and Luis Camarinha-Matos. Agnostic informatics system of systems: The open isos services framework. pages 407–420, 09 2017.

[3] Carlos Gonçalves, Tiago Dias, A. Osório, and Luis Camarinha-Matos. *A Multi-supplier Collaborative Monitoring Framework for Informatics System of Systems*, pages 44–53. 09 2022.

[4] GitLab. What is ci/cd? `https://about.gitlab.com/topics/ci-cd/`. [Online; accessed 17-March-2024].

[5] Sphinx Technologies. What is podman? `https://docs.podman.io/en/latest/`, 2019. [Online; accessed 14-March-2024].

[6] Quarkus. Container images. `https://quarkus.io/guides/container-image`. [Online; accessed 14-March-2024].

[7] Eclipse Foundation. Cloud-native java applications without a hassle. `https://eclipse.dev/jkube/`. [Online; accessed 14-March-2024].

[8] VmWare. What is a kubernetes cluster? `https://www.vmware.com/topics/glossary/content/kubernetes-cluster.html`. [Online; accessed 14-March-2024].

[9] Hashicorp. Introduction to vagrant. `https://developer.hashicorp.com/vagrant/intro`. [Online; accessed 17-March-2024].