GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Bachelor

submitted in partial fulfillment of the
requirements for the course “”

The Automation of implementation of Docker Container in XNAT

Tanae Bousfiha

Institute of Computer Science

Bachelor’s and Master’s Theses
of the Center for Computational Sciences
at the Georg-August-Universität Göttingen

xx. July 2025

Georg-August-Universität Göttingen
Institute of Computer Science

Goldschmidtstraße 7
37077 Göttingen
Germany

☎ +49 (551) 39-172000
FAX +49 (551) 39-14403
✉ office@informatik.uni-goettingen.de
🌍 www.informatik.uni-goettingen.de

First Supervisor:	Prof.Dr. Roman Grothausmann
Second Supervisor:	Philip Zaschke, MSc.

I hereby declare that I have written this thesis independently without any help from others and without the use of documents or aids other than those stated. I have mentioned all used sources and cited them correctly according to established academic citation rules.

Göttingen, xx. July 2025

Abstract

The area of the automation of implementation of Docker Container in xnat (eXtensible Neuroimaging Archive Toolkit) is attracting considerable interest in the Somnolink Project. This paper provides an overview of the automation of implementation of Docker Container in xnat. It was hypothesized that a Container could be implemented in a Project in xnat and to rework all the files out and to reupload the result files back to their places. The approach is partly based on using a Python Script that worked with REST API and create a Dockerfile and askes the user for an external/ Container Script and extract all the files from the Open source Xnat and upon the conclusion of the experience, the Docker Container in xnat should be in 'Complete' turned and the result files should be uploaded. Experimental application of the methods demonstrated that the container could not receive the extracted files via REST API, which led to a not feasible neither processing nor uploading files. Results revealed a significant correlation between the workflow data of xnat and the Mounting of data in the Docker container. These findings demonstrate that the extraction of all the files from all the levels of xnat leads to a problem Mounting in the xnat host before arriving to the container. And the fact that the Container has no access to Data Bank of xnat conducts that the effectiveness of the method proved to be context dependent and thus limited.

Contents

Chapter 1

Introduction

The automatisation of the implementation of the Docker container serves as a core foundation enabling scalability, security, ´reproducibility and rapid innovation. In the Context of the project like Somnolink provides not only an efficient service operation but also provides the foundation for evidence-based development, regulatory compliance, and effective research collaboration.
Docker streamlines the development lifecycle by allowing developers to work in the standardized environments using local containers which provide your applications and services. Container are great for continuous delivery workflow. This work contributes meaningfully to the field of medical informatics, through this approach, it can be used in automated patient data processing, medical AI Workflow, and secure data integration. It also provides a fast and efficient way to create new operations or updates, which can have positively a huge impact of the patient diagnosis or on the patient care in general.
Automation is the application of technology, programs, robotics or processes to achieve outcomes with minimal human input, and in the case of the implementation of the Docker container it describes the process of the Deployment, the Configuration and the actualisation of the Software use and their dependencies with the help of the Docker Container and REST APIS based technologies with reduced human assistance. while a large scale of previous studies has used the methods of the Docker Container the in the field of neuro-imaging,telemedicine applications and in the electronic health record systems.[1].One considerable study in this context we can recite the study of the integration of the artificial intelligence tools in the Clinical Research Setting: The Ovarian Cancer Use Case by Lorena Escudero Sanchez[2]. In their 2023 study they proposed using a integration of the AI tool and one of the methods used were the Docker-Xnat pipeline. the developed tool

[1]Collabnix, 5 benefits of Docker for the healthcare industry, 09.10.2023, https://collabnix.com/5-benefits-of-docker-for-the-healthcare-industry (Zugriff am 17.07.2025)

[2]Lorena Escudero Sanchez et al., Integrating Artificial Intelligence Tools in the Clinical Research Setting: The Ovarian Cancer Use Case, in: Diagnostics, Vol. 13(17), 2023, Article 2813, https://doi.org/10.3390/diagnostics13172813 (Zugriff am 17.07.2025)

were very powerful and were able to find tumors and precily. In addition by Satrajit Chakrabarty [3] who implemented a AI based framework that can automate a MRI scan tumor segmentation and characterization using the Docker Containers in xnat.
The Docker Container has a huge rule for the devolopment and integration of virtuell imaging platforms[4] they allow the the abstraction, installation, and configuration of environments so that software can be both distributed in self-contained images and used repeatably by tool consumers.[5] in addition XNAT is an open source imaging informatics platform developed by the Neuro-informatics Research Group at Washington University. [6] and despite the flexibility of xnat of integrating external tool (AI, pipelines or plugins) the process of integrating the Docker Containers remains a challenge and requires often steps and manual scripting. Several studies has been using the implementation of Docker Containerization but few of them have addressed the idea to automatize the idea of the implementation on xnat, in order to address this gaps limits the current study propose a Docker-Container-based framework integrated in xnat that enable the fully automation of the Docker implementation in xnat.

[3]et al., Deep learning-based end-to-end scan-type classification, pre-processing, and segmentation of clinical neuro-oncology studies, in: Proceedings of SPIE, Vol.12469 (2023), Article 124690N, https://doi.org/10.1117/12.2647656 (Zugriff am 17.07.2025)

[4]Satrajit Chakrabarty et al., Container applications for the development and integration of virtual imaging trials, in: Medical Physics, 2025, https://doi.org/10.1002/mp.17777 (Zugriff am 17.07.2025)

[5]SatrajitChakrabarty et al., Container applications for the development and integration of virtual imaging trials, in: Medical Physics, 2025, https://doi.org/10.1002/mp.17777 (Zugriff am17.07.2025)

[6]XNAT, About XNAT, https://www.xnat.org/about/ (Zugriff am 17.07.2025).

Chapter 2

Project Description

The aim of this project is to provide a automation for the process of the implementation of the docker container in xnat. therefore in order to achieve this goal a significant workload remains. As an open-source imaging informatics software platform (xnat) dedicated to improve the imaging-based research, xnat provide a lot of fonction in the medical field, under this functions we can list A full DI-COM Integration and Anonymization, a secure access and permission control and a the power of high-performance computing on the data data.[1] Most of the cases if we want to implement a docker container in xnat we proceed by writing a external script that the container will then contain, assuming that we are working on xnat with the Container Service plugin already installed. Afterward we start writting a dockerfile.[2] Before deploying the container in xnat a JSON Command must be written. After this the container can be without any problem in xnat deployed launched. The primary goal of the the automation is to to summarize all of these steps into a single script which receives an external script from the user for each creation of a container.

2.1 The Dockerfile :

A dockerfile is a text-based document that contains all the commands a user could call on the command line to create an image. It provides instructions to the image builder on the commands to run, files to copy, startup command, and more.[3], Beforehand writing the docker-file we need to make sure that the docker engine is installed and opened.The Docker Engine is an open source containerization technology for building and containerizing the applications.[4]. A Dockerfile always startes with FROM<IMAGE> this specifies the base image that the build will elongate.Then we have WORKDIR <path> and it specifies the "working directory"or the procedure in the image

[1]https://www.xnat.org/
[2]https://docs.docker.com/reference/dockerfile/
[3]https://docs.docker.com/get-started/docker-concepts/building-images/writing-a-dockerfile/
[4]https://docs.docker.com/engine/

where files will be copied and commands will be executed. then we have COPY <host-path> <image-path> this instruction tells the builder to copy files from the host and put them into the container image. The RUN <command> tells the builder to run the specified command. After writing and specifying the instruction in the Docker file and the Docker Engine is installed and turned on, then we can proceed with building the image. In order to build the Docker image a basic command can be executed : docker build . . [5]

2.2 Building and Tagging the image:

With the purpose of building a image for the docker container a basic command can be used: "docker build . ", the name of the image can be specifies between the word build and the point and the final point in the command refers to the path to the build context. Given that xnat is configured to recognize only docker images that are publicly available on Docker Hub, it is required after building the image to push and tag the image. The process begin with Tagging the image. T

[5]https://docs.docker.com/get-started/docker-concepts/building-images/build-tag-and-publish-an-image/

Chapter 3

System

In this chapter, the analysis of ...

Chapter 4

Realisation

In this chapter, the design of ...

Chapter 5

Testing and results

In this chapter, the implementation of ...

Chapter 6

Discussion

In this chapter, the conclusion of ...

Bibliography