DM848: Microservice Programming

Sandboxed Interpreter for Interactive Code Execution

Anders Busch, anbus12@student.sdu.dk

June 23, 2016

Contents

1 Introduction

When learning a new programming language, it is often useful to have a environment where the learner can execute code, quick and easy without having to worry about breaking the execution environment or installing new software locally.

The goal of this project was to create a cloud server for executing JOLIE programs through the browser; and to embed the cloud server in a virtual environment provided by the virtualization library DOCKER.

To achieve this goal, we had to solve the following problems:

1. Create an DOCKER image for running JOLIE code in a DOCKER container. This image helps automating the process of creating new containers and is available on-line[1] for everyone interested in using DOCKER in conjunction with JOLIE.

2. Produce a service for handling the setup of the DOCKER container. As our execution environment uses JOLIE to communicate between services, we opted to create a service that handles the communication with the DOCKER CLI.

3. Make a service that can evaluate JOLIE within the DOCKER container. For our evaluator to work, we have decided to use a JOLIE service to evaluate sent JOLIE code.

4. Assemble a front-end service for presenting and handling of the user interface. Here we opted for a simple JOLIE-based HTTP server that serves our user interface.

2 Preliminaries

3 Technical Description

4 Related Work and Discussion

Although we partly achieved what we set out to create; there are some limitation in the current implementation that can be expanded upon in future revision of this execution

[1] https://hub.docker.com/r/ezbob/jolie/

tool, namely the support for inter-service communications, better error explanation and dynamic linting[2] of the program code.

[2]https://en.wikipedia.org/wiki/Lint_%28software%29