

# Chapter 1

## Introduction

This project is an attempt to solve problems commonly surface when learning programming languages or performing programming as a leisure activity.

### 1.1 Learning programming

When learning a programming language or programming itself – either via organized education or self-teaching – it is necessary to occasionally test one’s acquired theoretical knowledge in action and measure their progression. This may be done for many reasons, including general practicing, self-evaluation or grading students of a course. However, in practice testing may be difficult in all of these situations due to different inconveniences.

An autodidact may find it difficult to come up with problems matching their level of skills, and even evaluating their solution isn’t as easy as it may seem. A lot of computational tasks require a verified, correctly working second program to compare results or a list of expected results for specific inputs, both of which require a second party to create them first. It is also easy to fall into the trap of believing in the correctness of your solution based on a limited set of test cases you have created on your own.

As for communal education – formal and non-formal alike –, while the educator may be proficient in coming up with good exercises, it usually is very challenging for them to evaluate and/or grade submitted works due to the number of students. This problem is present not just in schools and universities, but boot camps, online courses and company training as well.

Thus, some sort of automated testing system becomes necessary. Many institutions, especially universities and online learning platforms already provide solutions for this problem, but they are often insecure, ad hoc in nature, and require the superfluous development of a relatively large, mostly analogous overhead by all par-

ticipants.

## **1.2 Recreational programming**

The other important target subject of the project is the recreational programming scene. By its nature, this area can strongly overlap with the aforementioned educational programming, as free time programming can be viewed as a way of skill development, but it also covers purely leisure activities. Several communities exist in the programming world that publish programming challenges of some sorts, many of them mainly design games for which coders can write bots, either competing with each other or simply solving puzzles. But generally, open solutions are not commonplace and existing frameworks and websites usually don't provide the ability to publish custom games.

## **1.3 Goal of the project**

The goal of the project is to provide a safe runtime for untrusted programs that interact with each other on top of the JVM platform. This framework shall also allow the development of games/challenges and their bots in an easy-to-integrate and environment-familiar way. These games should be able to effortlessly provide meaningful results which can then be processed by a user of the runtime.

The framework must allow the development of both single-, and two-player games so that it can satisfy the needs of both educational-, and recreational programming.

The runtime must be integrateable into different projects and configured for diverse environments, but should also contain a default management interface for game and bot control.