

УДК 004.42

РАЗРАБОТКА ПОДСИСТЕМЫ ИСПОЛНЕНИЯ ПОЛЬЗОВАТЕЛЬСКОГО КОДА НА ПЛАТФОРМЕ АВТОМАТИЗИРОВАННОЙ ПРОВЕРКИ ВЫПОЛНЕНИЯ ТЕСТОВЫХ ЗАДАНИЙ ПО ПРОГРАММИРОВАНИЮ

Коков Н.А. (Университет ИТМО)

Научный руководитель – доцент, кандидат технических наук Перл И.А.
(Университет ИТМО)

Введение. Каждый год сотни, если не тысячи, студентов ИТМО в процессе обучения выполняют различные работы, суть которых заключается в написании программного кода для решения определенной задачи. Университет не предоставляет доступа к какой-либо системе для организации выполнения такого рода работ, поэтому преподавателям приходится искать различные способы организации проверки студенческих программ. Некоторые преподаватели предпочитают ручную проверку, другие – пытаются использовать такие сервисы как Contester, HackerRank и др. У подобных сервисов есть различные функциональные особенности и недостатки, которые ограничивают возможности педагогов. Также некоторые сервисы предоставляют только платный доступ к своему функционалу.

Одно из основных функциональных требований при разработке автоматизированной системы проверки выполнения тестовых заданий по программированию является возможность исполнения пользовательского кода с получением результатов его работы.

Данная работа посвящена разработке подсистемы, в основные задачи которой входит:

- 1) безопасное исполнение пользовательского кода в изолированной среде;
- 2) сохранение и выдача выходных результатов работы программного кода пользователя;
- 3) выдача подробной информации при возникновении ошибок в пользовательском коде во время его исполнения или сборки.

Основная часть. Существуют различные способы организации исполнения пользовательского кода:

- исполнение на локальной машине – код загружается на локальную машину, компилируется и выполняется в той же среде, что и сама подсистема. Также можно использовать удаленный сервер вместо локальной машины;
- исполнение в специально организованных для этого песочницах;
- исполнение в контейнерах.

В разрабатываемой подсистеме был выбран метод контейнеризации пользовательских программ. Данный метод хорошо поддается расширению, так как позволяет добавить поддержку нового языка программирования за счет загрузки подходящего базового образа, а также позволяет организовать безопасное исполнение кода в изолированной среде.

Также была учтена простота реализации, так как: во-первых, платформа Docker имеет очень понятный и простой REST API интерфейс для взаимодействия; во-вторых, сама подсистема будет выполняться в Docker контейнере, из которого можно получить доступ к платформе Docker хоста, примонтировав файл unix-сокета докера.

Чтобы добиться максимальной эффективности при исполнении студенческого кода, для каждого языка программирования в каждом задании заранее подготавливается докер-образ со всеми наборами входных данных. Для исполнения пользовательского решения создается контейнер, в который загружается исходный код решения. Далее запускается контейнер. Точкой входа является команда исполнения специального bash-скрипта, который запускает студенческое решение с каждым набором входных данных отдельно. Если язык программирования является компилируемым, то перед запуском происходит сборка решения.

Выводы. Разработанная подсистема позволяет организовать взаимодействие с платформой Docker для контейнеризации студенческих программ и их исполнения, а также получить

результат его работы с созданным учителем набором входных данных для последующей оценки работы алгоритма.

Коков Н.А. (автор)

Подпись

Перл И.А. (научный руководитель)

Подпись