

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АВТОМАТИЗАЦИИ РАЗВЕРТЫВАНИЯ, ЗАПУСКА И ПРОВЕРКИ ПРИЛОЖЕНИЙ В ОБЛАЧНОЙ ИНФРАСТРУКТУРЕ НА БАЗЕ КОНТЕЙНЕРОВ DOCKER.

Дуж Д. В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Сурков К. А. – ассистент кафедры ПОИТ

Облачные технологии являются самым современным способом развертывания приложений. Они позволяют снизить издержки по поддержанию приложений в рабочем состоянии, а также позволяют повысить их доступность за счет использования специализированного программного обеспечения и оборудования, а также различных техник, как например разделение интернет-запросов по регионам.

Требование масштабируемости приложений, развертываемых в облачной инфраструктуре, привело к появлению технологии под названием LXC [1]. Она представляет из себя способ виртуализации на уровне операционной системы. Это позволяет значительно повысить эффективность использования ресурсов, так как исключается необходимость виртуализации ядра операционной системы.

Программный продукт Docker [2] предоставляет экосистему, построенную вокруг LXC [1]. Он позволяет пользователям упаковывать в один модуль все файлы, необходимые для запуска приложения, а также создавать изолированные системы на базе этого модуля и запускать в них приложения. Эти возможности являются ключевыми при разработке облачной инфраструктуры. Они позволяют гарантировать пользователям, что приложение будет работать в облаке ровно так же, как у самих пользователей в локальных системах.

Разработанное программное средство позволяет, взаимодействуя с Docker-ом [1], разворачивать консольные, а также веб-приложения, в облачной инфраструктуре с последующим отображением интерфейса этих приложений используя веб-технологии.

Для упрощения взаимодействия с приложением, пользователю в большинстве случаев нет необходимости знать о том, как именно его приложение запускается в облаке. С этой целью в веб-клиенте программного средства реализована функция создания приложений из шаблонов, заранее сконфигурированных для запуска в облаке.

Шаблоны представляют из себя git-репозитории, размещенные в GitHub. Пользователь, авторизуясь на одном из хостингов, например GitHub, GitLab, BitBucket, может создать репозиторий из шаблона, затем в этом репозитории создать приложение, которое будет работать в облаке.

В данном программном средстве реализовано взаимодействие между студентами и преподавателем с учетом описанных выше возможностей. Преподаватель создает задачи, которые студенты должны решать. После того как студенты предоставляют решения на проверку, преподаватель может без надобности устанавливать программное обеспечение, необходимое для запуска решений студентов, используя лишь браузер, проверить корректность их работы.

Функция создания задач преподавателем обязывает предоставлять редактор текста. Для этого используется библиотека Monaco Editor [3], которая является одной из самых продвинутых реализаций текстового редактора, к тому же она доступна для использования в браузере. Описание заданий представлено в виде Markdown-разметки, для которой в Monaco Editor [3] существует встроенная поддержка.

Программное средство представляет из себя клиент-серверное приложение на монолитной архитектуре. На стороне сервера реализован REST API на технологии ASP.NET Core [4]. В качестве СУБД для хранения всей пользовательской информации используется PostgreSQL [5]. Веб-приложение, в свою очередь, реализовано с использованием технологий React [6] и Reactronic [7].

Список использованных источников:

1. LXC [Электронный ресурс]. – Режим доступа <https://linuxcontainers.org/>. – Дата доступа: 07.04.2022.
2. Docker [Электронный ресурс]. – Режим доступа <https://www.docker.com/>. – Дата доступа: 07.04.2022.
3. Monaco Editor [Электронный ресурс]. – Режим доступа <https://microsoft.github.io/monaco-editor/>. – Дата доступа: 07.04.2022.
4. ASP.NET Core [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-6.0/>. – Дата доступа: 07.04.2022.
5. PostgreSQL [Электронный ресурс]. – Режим доступа: <https://postgresql.org/>. – Дата доступа: 07.04.2022.
6. Руководство по React [Электронный ресурс]. – Режим доступа: <https://ru.reactjs.org/>. – Дата доступа: 07.04.2022.

7. *Reactronic* [Электронный ресурс]. – Режим доступа: <https://github.com/nezaboodka/reactronic/>. – Дата доступа: 07.04.2022.