

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ КОНТЕЙНЕРИЗАЦИИ ДЛЯ ДИСТАНЦИОННОГО ОБУЧЕНИЯ ПО ИТ СПЕЦИАЛЬНОСТЯМ

Калоша С.О., Нестеренков С.Н.

*Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Беларусь,
swetofdark@gmail.com*

Abstract. Distance learning is being actively used in IT education. Although it has a lot of advantages, there are some issues of organizing practice tasks. Considering some similarities with software development world, the article shortly reviews containerization technology, suggests a solution based on its usage and demonstrates it in the examples.

Дистанционное обучение по ИТ специальностям получило большое распространение за последнее десятилетие, в частности, на территории стран постсоветского пространства. Рост престижа ИТ сферы оказывает влияние не только на абитуриентов, определяющихся с какой специальностью связать свое будущее, но также увеличивает количество специалистов, желающих сменить свой основной род деятельности. Для второй категории обучающихся дистанционная модель образования зачастую является более предпочтительной.

Среди недостатков дистанционного обучения выделяют слабое формирование практических навыков у обучаемого [1]. Действительно, для некоторых специальностей невозможно сформировать практические навыки без работы с лабораторным оборудованием. Однако для большинства ИТ специальностей таким оборудованием является персональный компьютер, который и так входит в необходимый минимум для дистанционного обучения. Таким образом, для ИТ специальностей данный недостаток не является проблемой столь острой, как, например, формирование коммуникативных навыков [2].

При более детальном рассмотрении организации практических работ в условиях дистанционного образования для таких специальностей можно выделить ряд других проблем, связанных с этапами выдачи, выполнения и проверки заданий. Например, проблемы, связанные с необходимостью создания единой среды для выполнения и проверки заданий:

- описание практической работы должно включать в себя не только условия самого задания, но и инструкцию по воссозданию окружения для его выполнения;

- время, затраченное на подготовку окружения, может быть больше, чем время на выполнение основного задания;

- аналогично, при проверке заданий, преподаватель тратит время на воссоздание окружения, в котором задания были выполнены;

- различия в настройках персональных компьютеров преподавателя и обучаемого способны исказить результаты выполненной работы или существенно замедлить ее проверку.

Исходя из вышесказанного, наличие инструмента, который уменьшит затраты времени на подготовку окружения для выполнения и проверки практических заданий, а также сведет вероятность ошибки при воссоздании этого окружения к минимуму, способно улучшить процесс дистанционного выполнения практических работ.

Проблема воспроизведения одинаковых окружений относится не только к вопросам обучения студентов ИТ специальностей или дистанционного обучения в целом. Так, например, данная необходимость существует и при разработке программного обеспечения распределенными командами [3]. Одним из решений этой проблемы в мире разработки ПО является использование технологий контейнеризации, в частности технологии Docker.

Все технологии контейнеризации базируются на понятии контейнера. Под контейнером (Container) подразумевают средство инкапсуляции программных приложений вместе с их зависимостями.

В целом контейнеры похожи на виртуальные машины (VM), но обладают рядом преимуществ, таких как:

- малое время запуска и останова контейнеров и простота их конфигурирования;

- переносимость – один и тот же контейнер может выполняться на различных компьютерах, с разными операционными системами, при этом гарантируя одинаковое поведение самого контейнера;

- простая структура контейнера позволяет запускать одновременно десятки контейнеров, что позволяет имитировать реальные промышленные системы [4].

Не углубляясь в особенности реализации технологий контейнеризации, рассмотрим основные их понятия и компоненты на примере технологии Docker [5]. На рисунке 1 схематически изображены основные ее компоненты.

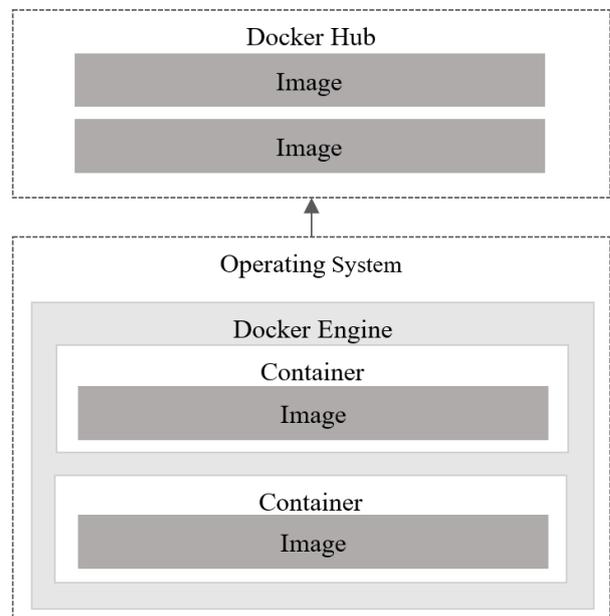


Рисунок 1 – Основные компоненты технологии Docker

Понятие контейнера (Container) в Docker не отличается от понятия контейнера, данного ранее. Контейнер представляет собой выполняющееся приложение и его зависимости. Например, контейнер может содержать сервер баз данных, веб-сервер или самое простое консольное приложение. Для описания контейнеров и их распространения используются их образы (Image). То есть, если контейнер является сервером баз данных, то образ содержит в себе его конфигурацию, например, логин и пароль его администратора или имя служебной базы данных. За выполнение контейнеров на локальной машине пользователя отвечает Docker Engine – единственный модуль, необходимый для установки на целевых вычислительных машинах. Для обмена и хранения образов приложений можно использовать любое средство обмена файлами, однако существует облачный интернет-ресурс Docker Hub, на котором представлены официальные образы большинства приложений и сервисов, используемых в разработке ПО. Далее рассмотрим два примера организации процесса дистанционного выполнения и проверки практических работ.

На рисунке 2 показан пример выдачи и проверки практической работы, где обучаемый должен реализовать приложение для работы с базами данных. Выбор СУБД не ограничен конкретными технологиями.

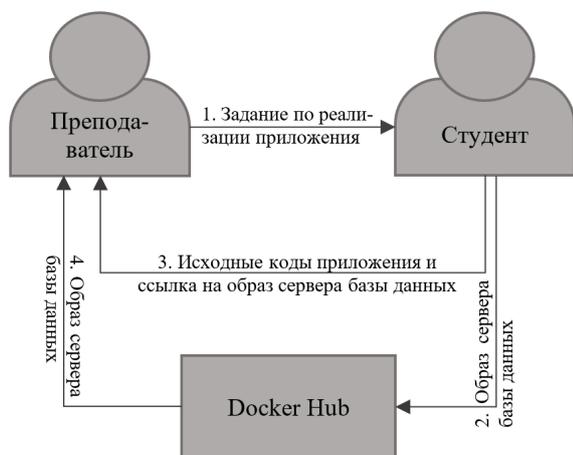


Рисунок 2 – Пример взаимодействия преподавателя и студента с использованием Docker для проверки результатов практической работы

В данном примере преподаватель высылает только требования к приложению (шаг 1). Далее обучаемый выполняет задание используя контейнер сервера базы данных, а его образ отправляет в хранилище образов (шаг 2). На проверку отправляется исходный код приложения и ссылка на образ контейнера, которую также можно встроить в исходный код приложения (шаг 3). К слову, для обмена исходными кодами уже активно используется другая технология, а именно системы контроля версий, такие как git и svn. В результате, для подготовки к проверке задания, преподавателю достаточно запустить контейнер сервера баз данных (шаг 4), вместо его полноценной установки и конфигурации, что в свою очередь сохранит время на проверку самого приложения.

Второй пример проиллюстрирован на рисунке 3.

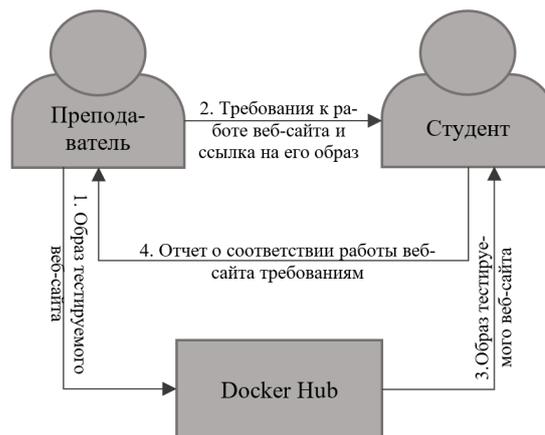


Рисунок 3 – Пример взаимодействия преподавателя и студента с использованием Docker для выдачи задания практической работы

Здесь обучаемый должен протестировать некоторый веб-сайт на соответствие требованиям. На этапе подготовки задания преподаватель «упаковывает» тестируемый веб-сайт в контейнер и отправляет его в хранилище (шаг 1). Далее обучаемому высылается образ веб-сайта и требования к его работе (шаг 2). Обучаемый получает и запускает образ веб-сайта на локальной машине (шаг 3). Вместо полноценной настройки веб-сайта можно сосредоточиться непосредственно на его тестировании (шаг 4).

В рассмотренных нами примерах использование технологии контейнеризации позволило уменьшить затраты времени как на выполнение задания, так и на его проверку. Также удалось избежать проблем с разными персональными компьютерами и операционными системами. Таким образом, применение технологий контейнеризации, в частности Docker, способно повысить эффективность дистанционного выполнения практических работ студентами IT специальностей, что должно положительно сказаться на качестве полученных ими практических навыков. Однако стоит отметить, что ознакомление с самой технологией потребует некоторого времени как у преподавателя, так и у обучаемого.

Литература

1. Кислухина, А. Использование дистанционных образовательных технологий в системе высшего образования: проблемы и перспективы / И. А. Кислухина. – Управление экономическими системами : электронный научный журнал. – 2017. – №9. – С.103.
2. Dabaj, F. Analysis of Communication Barriers to Distance Education : A Review Study / F. Dabaj, A. Yetkin – Online Journal of Communication and Media Technologies. – 2011. – №1. – С.1-15.
3. Morris, K. Infrastructure as Code: Managing Servers in the Cloud / K. Morris. – O'Reilly Media Inc. – 2016.
4. Mouat, A. Using Docker. Developing and Deploying Software with Containers / A. Mouat. – O'Reilly Media Inc. – 2015.
5. Docker Overview [Электронный ресурс] – Режим доступа: <https://docs.docker.com/engine/docker-overview/>.