

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ливанов Дмитрий Викторович

Должность: Ректор

Дата подписания: 17.06.2025 16:34:03

Уникальный программный ключ:

c6d909c49c1d2034fa3a0156c4eaa51e723263a2

Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»

ВЫПИСКА ИЗ ПРОТОКОЛА № 4

заседания учебно-методического совета от «29» ноября 2021 г.

ПОВЕСТКА:

Рассмотрение дополнительных общеобразовательных и профессиональных программ.

Проректор по учебной работе А. А. Воронов

СЛУШАЛИ: Директора Физтех-школы прикладной математики и информатики А.М. Райгородского.

ПОСТАНОВИЛИ:

Рекомендовать к утверждению в установленном порядке дополнительную общеобразовательную программу «Docker и Gitlab CI для начинающих»

Решение принято единогласно заочным голосованием.

Форма проведения заседания: заочная

Председатель УМС МФТИ

А.А. Воронов

Ученый секретарь УМС МФТИ

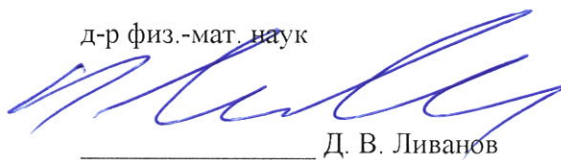
М.В. Березникова

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»

УТВЕРЖДАЮ

Исполняющий обязанности ректора

д-р физ.-мат. наук



Д. В. Ливанов

«29» ноября 2021 г.

Программа дополнительного образования

«Docker и Gitlab CI для начинающих»

Москва 2021

1. Общая характеристика программы

1.1. Целью реализации дополнительной общеобразовательной программы «Docker и Gitlab CI для начинающих» (далее – программа) является приобретение и совершенствование профессиональных, общепрофессиональных, универсальных и общекультурных компетенций слушателей для работы с инструментом контейнеризации Docker и использованию Gitlab CI.

1.2. Категории слушателей, на обучение которых рассчитана программа: начинающие разработчики, менеджеры проектов, DevOps инженеры.

1.3. Нормативный срок освоения программы – 40 академических часов.

1.4. Форма обучения – очная (с применением дистанционных образовательных технологий).

1.5. Режим обучения: дистанционно – 4 дня (10 ак. часов в день).

2. Планируемые результаты обучения

В результате освоения программы слушатель должен:

Владеть навыками:

- настройки Docker на любой современной ОС;
- настройки CI/CD на примере GitLab (gitlab-ci)

знать:

- основы работы с инструментом контейнеризации Docker;
- базовый инструмент Gitlab CI;

уметь:

- настраивать Docker на любой современной ОС
- создавать, подключать и удалять контейнеры
- работать с файловыми системами в Docker - UnionFS и OverlayFS
- работать с данными в Docker
- работать с репозиториями docker-контейнеров
- писать конфигурации CI
- настраивать pipelines, jobs, stages
- создавать собственный runner

3. Структура программы

Программа предусматривает изучение следующих модулей:

Модуль 1. Основы Docker (16 ак. ч)

Модуль 2. Gitlab CI (24 ак. ч)

Таблица 1

№ п/п	Модуль	Тема	Количе- ство ча- сов	В том числе	
				ауд. за- нятия	самост. работа
1.	Основы Docker	Виртуализация и контейнеризация. Docker engine	0,5	0,5	
2.		Docker image и Docker container	0,5	0,5	
3.		Файловые системы в Docker. Un- ionFS и OverlayFS	3	1	2
4.		Работа с данными: добавление файлов, монтирование внешних директорий. Обмен данными с уже развернутым контейнером	3	1	2
5.		Рекомендации по работе с Docker. Как писать DockerFile, .dockrignore. Базовая безопас- ность в docker	3	1	2
6.		Репозитории docker-контейнеров и работа с ними	3	1	2
7.		Docker compose	3	1	2
8.	Gitlab CI	Настройка pipelines, jobs, stages. Написание конфигураций CI	11	4	7
9.		Создание собственного runner'a. Executors: shell, docker	13	5	8
ИТОГО			40	15	25

4. Содержание программы

4.1. Учебно-тематический план программы

Таблица 2

№ п/п	Наименование модуля, разделов и тем	Содержание обучения, наименование и тематика практических занятий (семинаров), самостоятельной работы	Всего, час
1	Основы Docker		16
1.1	Виртуализация и контейнеризация. Docker engine		0,5
1.1.1	Лекция	Виртуализация и контейнеризация. Docker engine	0,5
1.2	Docker image и Docker container		0,5
1.2.1	Лекция	Docker image и Docker container	0,5
1.3	Файловые системы в Docker. UnionFS и OverlayFS		3
1.3.1	Лекция	Файловые системы в Docker. UnionFS и OverlayFS	0,5

1.3.2	Практическое занятие		0,5
1.3.3	Самостоятельная работа. Изучение дополнительных материалов		2
1.4	Работа с данными: добавление файлов, монтирование внеш-них директорий. Обмен дан-ными с уже развернутым кон-тейнером		3
1.4.1	Практическое занятие	Работа с данными: добавление файлов, монтирование внеш-них директорий. Обмен дан-ными с уже развернутым кон-тейнером	1
1.4.2	Самостоятельная работа. Работа с данными		2
1.5.	Рекомендации по работе с Docker. Как писать DockerFile, .dockrtignore. Базовая безопас-ность в docker		3
1.5.1	Практическое занятие	Работа с Docker. Как писать DockerFile, .dockrtignore. Базовая безопасность в docker	1
1.5.2	Самостоятельная работа. Пишем DockerFile, .dockrtignore		2
1.6.	Репозитории docker-контейнеров и работа с ними		3
1.6.1	Лекция	Репозитории docker-контейнеров	0,5
1.6.2	Практическое занятие	Работа с репозиториями docker-контейнеров	0,5
1.6.3	Самостоятельная работа.		2
1.7.	Docker compose		3
1.7.1	Лекция	Docker compose	0,5
1.7.2	Практическое занятие	Работа с инструментом Docker compose	0,5
1.7.3	Самостоятельная работа. Работа с инструментом Docker compose		2
2.	Gitlab CI		24
2.1.	Настройка pipelines, jobs, stages. Написание конфигураций CI		11
2.1.1	Лекция	Pipelines, jobs, stages. Конфигурации CI	1
2.1.2	Практическое занятие	Настройка pipelines, jobs, stages. Написание конфигураций CI	3
2.1.3	Самостоятельная работа. Настройка pipelines, jobs, stages. Написание конфигураций CI		7

2.2.	Создание runner'a. Executors: shell, docker		13
2.2.1	Лекция	Создание runner'a. Executors: shell, docker	2
2.2.2		Создание собственного runner'a. Executors: shell, docker	3
2.2.3	Самостоятельная работа. Создание собственного runner'a. Executors: shell, docker		8
Итого:			40

4.2. Учебная программа по модулям

Таблица 3

№ п/п	Наименование модуля, разделов и тем	Содержание обучения и тематика практических занятий (семинаров), самостоятельной работы, используемых образовательных технологий
1.	Основы Docker	Виртуализация и контейнеризация. Docker engine. Docker image и Docker container. Файловые системы в Docker. UnionFS и OverlayFS. Работа с данными: добавление файлов, монтирование внешних директорий. Обмен данными с уже развернутым контейнером. Рекомендации по работе с Docker. Как писать DockerFile, .dockrignore. Базовая безопасность в docker. Репозитории docker-контейнеров и работа с ними. Docker compose
2.	Gitlab CI	Настройка pipelines, jobs, stages. Написание конфигураций CI . Создание собственного runner'a. Executors: shell, docker

4.3. Список рекомендуемой литературы

1. Эдриен Моуэт. Использование Docker. Разработка и внедрение программного обеспечения при помощи технологии контейнеров, 2017
2. Парминдер Сингх Кочер. Микросервисы и контейнеры Docker, 2019
3. Gabriel N. Schenker. Learn Docker — Fundamentals of Docker 19.x: Build, test, ship, and run containers with Docker and Kubernetes, 2020
4. Иван Портянкин. Программирование Cloud Native. Микросервисы, Docker и Kubernetes, 2020
5. Docker: Enterprise Container Platform [Электронный ресурс]. — www.docker.com
6. Docker Hub [Электронный ресурс]. — hub.docker.com
7. Continuous Integration для новичков [Электронный ресурс]. — <https://habr.com/post/352282/>
8. CI/CD: принципы, внедрение, инструменты [Электронный ресурс]. — <https://medium.com/southbridge/ci-cd-принципы-внедрение-инструменты-f0626b9994c8>
9. Непрерывная интеграция [Электронный ресурс]. — https://ru.wikipedia.org/wiki/Непрерывная_

10. Что такое непрерывная интеграция [Электронный ресурс]. –
<https://ru.stackoverflow.com/questions/470453/Что-такое-непрерывнаяинтеграция/471034>

5. Материально-технические условия реализации программы

Таблица 4

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Информационно-коммуникационная платформа дистанционных лекций и семинаров	Лекции, практические занятия (дистанционные семинары)	Слушателю необходимо наличие доступа в сеть интернет, компьютер. Преподавателю курса необходимо оборудование для проведения дистанционных лекций, семинаров (вебинаров), качественный отказоустойчивый доступ в сеть интернет.
Информационно-коммуникационная платформа	Самостоятельная работа	Наличие компьютера и доступа в сеть интернет.
Информационно-коммуникационная платформа дистанционных лекций и семинаров	Промежуточный контроль	Наличие компьютера и доступа в сеть интернет.

6. Оценка качества освоения программы

Оценка качества освоения программы осуществляется в процессе промежуточной аттестации. Формы и методы промежуточного контроля представлены в таблице 5.

Таблица 5

Наименование модулей	Основные показатели оценки	Формы и методы контроля и оценки
Основы Docker	Количество правильных ответов	Тестирование
Gitlab CI	Количество правильных ответов	Тестирование

7. Составители программы

Райгородский Андрей Михайлович

Доктор физико-математических наук, директор ФПМИ МФТИ

Благодарный Евгений Владимирович

заведующий учебно-методической лабораторией инноватики ФПМИ

Ивченко Олег Николаевич

старший преподаватель кафедры АТП

Иванова Анастасия Сергеевна

руководитель проектов учебно-методической лаборатории инноватики ФПМИ

Согласовано

Директор ФПМИ, д.ф.-м.н.

 А.М. Райгородский

« ____ » _____ 2021 г.

Пояснительная записка
к разработке и реализации дополнительной общеобразовательной программы
«Docker и Gitlab CI для начинающих»

Данная программа дает слушателям основные понятия - что такое Docker, контейнеры и как использовать их в той или иной среде, а также, как работает непрерывная интеграция GitLab. Docker – это открытая платформа для разработки, доставки и эксплуатации приложений. Применение Docker существенно уменьшает время между написанием и запуском кода. В своем ядре Docker позволяет запускать практически любое приложение, безопасно изолированное в контейнере. Безопасная изоляция позволяет запускать на одном хосте много контейнеров одновременно без дополнительной нагрузки гипервизора.

В ходе обучения слушателей научат устанавливать Docker, настраивать для запуска на локальной системе, клонировать и обрабатывать образы, реализовывать контейнеры, а также управлять образами и контейнерами. По завершении слушатели смогут применять контейнеры в своих проектах и средах с полным пониманием того, в каких случаях стоит или не стоит прибегать к этой технологии. Слушатели также получают практический опыт настройки pipelines, jobs, stages, смогут создавать собственный runner.

Задачи курса «Docker и Gitlab CI для начинающих»

- 1- научить слушателей устанавливать и настраивать платформу Docker, применять контейнеры в своих проектах в различных окружениях.
- 2- освоить на практике, с помощью Gitlab CI, путь от создания самого простого пайплайна до настройки сложных вариантов CI/CD

Программа рассчитана на 40 академических часа. Занятия состоят из лекций и семинаров, для поддержки обучающихся будут созданы чаты с преподавателями. В качестве контроля знаний слушатели пройдут тестирование.

После успешного прохождения программы слушатели получают сертификат о прохождении программы.

Составители программы:

Райгородский Андрей Михайлович

Доктор физико-математических наук, директор ФПМИ МФТИ

Благодарный Евгений Владимирович

заведующий учебно-методической лабораторией инноватики ФПМИ

Ивченко Олег Николаевич

старший преподаватель кафедры АТП

Иванова Анастасия Сергеевна

руководитель проектов учебно-методической лаборатории инноватики ФПМИ

Директор ФПМИ,

Зав. кафедрой дискретной математики

А.М. Райгородский д.ф.-м.н.

**КВАЛИФИКАЦИЯ И ОПЫТ ПРИВЛЕКАЕМОГО ПРЕПОДАВАТЕЛЬСКОГО СОСТАВА ДЛЯ РЕАЛИЗАЦИИ ПРОГРАММЫ
ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ «DOCKER И GITLAB CI ДЛЯ НАЧИНАЮЩИХ»**

Ф.И.О. лектора, год рождения	Информация об образовании, полученном в соответствии с образовательными программами высшего профессионального образования, дополнительного профессионального образования (в т.ч. о наличие званий и ученых степеней) и т.д.	Место работы, занимаемая должность в настоящий момент, общий трудовой стаж	Опыт преподавания и консультирования по предмету, согласующемуся с направлением лота (перечислить), педагогический стаж	Наличие опыта практической работы в отечественных и зарубежных организациях в сфере деятельности, совпадающей с направлением преподавания
Ивченко Олег Николаевич,	Высшее образование - специалитет, магистратура, специальность: Прикладная математика и информатика, квалификация: Магистр, ФГБОУ ВПО "Московский физико-технический институт (национальный исследовательский университет)"	Должность: Старший преподаватель Общий стаж: 4 года	Педагогический стаж: 4 года. Преподаватель курсов по обработке больших данных в МФТИ	Разработчик системы NJudge - системы автоматизированного тестирования Nadoor- приложений, системный администратор инфраструктуры больших данных в группе Яндекс-CERN