

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Ливанов Дмитрий Викторович  
Должность: Ректор  
Дата подписания: 19.03.2026 12:26:34  
Уникальный программный ключ:  
c6d909c49c1d2034fa3a0156c4eaa51e7232a3a2

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

**ВЫПИСКА ИЗ ПРОТОКОЛА № 5**  
**заседания учебно-методического совета от 27 декабря 2023**

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 Программу профессиональной переподготовки «Fullstack-разработчик на Golang Pro»

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

Ректор

д-р физ.-мат. наук



Д. В. Ливанов

«27» декабря 2023 г.

**Дополнительная профессиональная программа**  
**Программа профессиональной переподготовки**  
**«Fullstack-разработчик на Golang Pro»**

Москва 2023

## Оглавление

1. Рабочая группа	2
2. Профессиональные стандарты, учтённые в содержании программы	2
3. Квалификационные требования, учтённые в содержании программы	2
4. Требования федерального государственного стандарта высшего образования, на основании которых была разработана программа	2
5. Характеристика новой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации.	2
6. Характеристика компетенций, подлежащих совершенствованию, и (или) перечень компетенций, формирующихся в результате освоения программы	5
7. Цель программы	5
8. Планируемые результаты обучения по программе	5
9. Учебный план	6
10. Календарный учебный график	7
12.1. Требования к квалификации педагогических кадров/представителей предприятий и организаций, обеспечивающих реализацию образовательного процесса	13
12.2. Требования к материально-техническим условиям	13
12.3. Требования к информационным и учебно-методическим условиям	14
13. Формы аттестации и оценочные материалы по программе	16

## **1. Рабочая группа**

С целью повышения качества дополнительной профессиональной программы (ДПП) с учетом требований профессиональных стандартов (ПС) в группу разработчиков входили:

**Райгородский Андрей Михайлович**

Доктор физико-математических наук, директор ФПМИ МФТИ

**Благодарный Евгений Владимирович**

заведующий учебно-методической лабораторией инноватики

**Иванова Анастасия Сергеевна**

руководитель проектов учебно-методической лаборатории инноватики

## **2. Профессиональные стандарты, учтённые в содержании программы**

Настоящая дополнительная профессиональная программа (ДПП) разработана с учётом соответствующих ей профессиональных стандартов (ПС) из национального реестра профессиональных стандартов. Профессиональные стандарты (ПС), связанные с настоящей ДПП: 06.001 Программист

## **3. Квалификационные требования, учтённые в содержании программы**

Квалификационные требования, учтённые в содержании программы, указаны в описании должностей единого квалификационного справочника должностей руководителей, специалистов и служащих. Причём учтены требования должностей, указанных в дополнительных характеристиках, отобранных для данной программы ОТФ (п. 5. настоящего документа), которые приведены в указанных в п. 2. настоящего документа профессиональных стандартах (ПС) в строке «ЕТКС или ЕКС». Квалификационные требования инкорпорированы в настоящий документ путём отсылки и являются его неотъемлемой частью.

## **4. Требования федерального государственного стандарта высшего образования, на основании которых была разработана программа**

Настоящей дополнительной профессиональной программе соответствует федеральный государственный стандарт высшего образования (ФГОС ВО): 01.04.02 «Прикладная математика и информатика (уровень магистратура)».

Настоящая дополнительная профессиональная программа разработана на основании требований указанного ФГОС ВО.

Полный перечень требований приведён в тексте указанного ФГОС ВО, инкорпорирован в настоящий документ путём отсылки и является его неотъемлемой частью.

## **5. Характеристика новой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации.**

«Программа профессиональной переподготовки направлена на получение компетенции, необходимой для выполнения нового вида профессиональной деятельности, приобретение новой квалификации (часть 5 статьи 76 Федерального закона N 273-ФЗ). Для их определения и может использоваться ПС. При этом необходимо учесть различия терминологии, используемой в образовании и в профессиональных стандартах, о которых говорилось выше. Вид

профессиональной деятельности, квалификация, упоминаемые в цитируемой статье закона, в ПС в большинстве случаев соответствуют ОТФ, иногда - ТФ.»<sup>1</sup>,

Новые квалификации (ОТФ/ТФ) и их уровни:

06.001 Программист

Обобщенные трудовые функции			Трудовые функции		
код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
А	Разработка и отладка программного кода	3	Формализация и алгоритмизация поставленных задач	А/01.3	3
			Написание программного кода с использованием языков программирования, определения и манипулирования данными	А/02.3	3
			Оформление программного кода в соответствии с установленными требованиями	А/03.3	3
			Работа с системой контроля версий	А/04.3	3
			Проверка и отладка программного кода	А/05.3	3
В	Проверка работоспособности и рефакторинг кода программного обеспечения	4	Разработка процедур проверки работоспособности и измерения характеристик программного обеспечения	В/01.4	4

<sup>1</sup> Письмо Министерства образования и науки РФ от 22 апреля 2015 г. N ВК-1032/06 "О направлении методических рекомендаций"

			Разработка тестовых наборов данных	V/02.4	4
			Проверка работоспособности программного обеспечения	V/03.4	4
			Рефакторинг и оптимизация программного кода	V/04.4	4
			Исправление дефектов, зафиксированных в базе данных дефектов	V/04.5	4
C	Интеграция программных модулей и компонент и проверка работоспособности выпусков программного продукта	5	Разработка процедур интеграции программных модулей	C/01.5	5
			Осуществление интеграции программных модулей и компонент и верификации выпусков программного продукта	C/02.5	5
D	Разработка требований и проектирование программного обеспечения	6	Анализ требований к программному обеспечению	D/01.6	6
			Разработка технических спецификаций на программные компоненты и их взаимодействие	D/02.6	6
			Проектирование программного обеспечения	D/03.6	6

Характеристики данных ОТФ/ТФ приведены в профессиональном стандарте (ПС), указанном в п. 2. настоящего документа. инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Связанные виды профессиональной деятельности (ВПД):

научно-исследовательская;

производственно-технологическая;

проектная.

Характеристика данных ВПД приведена в федеральном государственном стандарте высшего образования (ФГОС ВО), указанном в п. 4. настоящего документа, инкорпорирована в настоящий документ путём отсылки, и является его неотъемлемой частью. Указанные ВПД, освоение которых предусмотрено указанным ФГОС ВО, являются «сквозными» и служат основой овладения выбранной квалификацией (ОТФ/ТФ).

ТФ, связанные с указанными в настоящем пункте ОТФ (в случае их указания), а также их характеристики приведены в профессиональном стандарте (ПС), указанном в п. 2. настоящего документа. инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

#### **6. Характеристика компетенций, подлежащих совершенствованию, и (или) перечень компетенций, формирующихся в результате освоения программы**

Учитывая, что слушатели программы уже освоили или осваивают основную профессиональную образовательную программу, в том числе общие компетенции, соответствующего уровня профессионального образования, компетенции формирующиеся в результате освоения настоящей программы и компетенции, подлежащие совершенствованию, а также их характеристика, представлены в ФГОС, указанном в п. 4. настоящего документа, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Данные компетенции указанного ФГОС ВО служат основой определения результатов обучения по программе.

#### **7. Цель программы**

Реализация программы профессиональной переподготовки направлена на получение компетенции, необходимой для выполнения нового вида профессиональной деятельности, приобретение новой квалификации.

В соответствие с Письмом Министерства образования и науки РФ от 22 апреля 2015 г. N ВК-1032/06 «О направлении методических рекомендаций», цель представляет собой осознанное представление (предвосхищение) результата деятельности. Планируемые результаты отражены ниже.

#### **8. Планируемые результаты обучения по программе**

Планируемые результаты обучения соответствуют результатам освоения соответствующей основной профессиональной образовательной программы, а также направлены на приобретение новой квалификации, требующей изменение направленности (профиля) или специализации в рамках направления подготовки (специальности) полученного ранее профессионального образования, определены на основе профессиональных компетенций соответствующего федерального государственного образовательного стандарта, указанного в п. 4. настоящего документа.

Имеющаяся квалификация и (или) уровень образования (требования к слушателям): Высшее, незаконченное высшее образование или среднее профессиональное.

Новые виды профессиональной деятельности, и ПК, соответствующие им, которые планируются к получению в результате обучения по настоящей программе, а также их характеристики, представлены в ФГОС, указанном в п. 4. настоящего документа, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Выпускник также должен обладать прочими компетенциями в соответствие с указанным ФГОС, которые приведены и охарактеризованы в нём, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Новые квалификации (ОТФ/ТФ), планируемые к приобретению в результате обучения по настоящей программе, приведены и охарактеризованы в п. 5. настоящего документа.

Приведённая информация о цели и результатах обучения является основой для разработки рабочих программ, оценочных материалов и иных компонентов дополнительной профессиональной программы.

## 9. Учебный план

№ п/п*	Наименование Модулей, дисциплин	Всего, час.	В том числе:				
			Лекции	Практические занятия (семинары)	Самостоятельная работа	Контрольные задания	Форма аттестации
1	Модуль 1. Разработка на Golang	125	30	30	63	2	проект
2	Модуль 2. Технологии программирования и операционные системы	135	30	15	75	15	проект
3	Модуль 3. Fullstack-разработка	135	30	30	65	10	проект
4	Итоговая квалификационная работа	35			35		проект
	<b>ИТОГО:</b>	<b>430</b>	<b>90</b>	<b>75</b>	<b>238</b>	<b>27</b>	

\* Слушатели могут проходить модули в произвольном порядке.

Срок освоения настоящей программы профессиональной переподготовки превышает минимально допустимый срок освоения 250 часов (либо другой установленный актуальными нормативными документами соответствующего Министерства срок).

## 10. Календарный учебный график

### Модуль 1. Разработка на Golang

Обучение по установленной форме: очная, с применением электронного обучения и дистанционных образовательных технологий																
Количество академических часов: 125																
Порядковый № занятия	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Лекции	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
практические	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
самостоятельные	2	2	2	2	2	2	1	2	2	2	2	2	2	2	1	35
контрольные задания	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
Итого	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	35

### Модуль 2. Технологии программирования и операционные системы

Обучение по установленной форме: очная, с применением электронного обучения и дистанционных образовательных технологий																
Количество академических часов: 135																
Порядковый № занятия	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Лекции	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
Практические занятия	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
Самостоятельная работа	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	30
Контрольные задания	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
Итого	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	30

### Модуль 3. Fullstack-разработка

Обучение по установленной форме: очная, с применением электронного обучения и дистанционных образовательных технологий																
Количество академических часов: 135																
Порядковый № занятия	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Лекции	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
Практические занятия	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
Самостоятельная работа	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	35

Контрольные задания	0	0	0	0	0	3	0	0	0	0	3	0	2	0	2	0
Итого	6	6	6	6	6	9	6	6	6	6	9	6	8	6	8	35

### Итоговая квалификационная работа

	Обучение по установленной форме: очная, с применением электронного обучения и дистанционных образовательных технологий
	Количество академических часов: 35
самостоятельная работа	35
Итого	35

## 11. Рабочие программы (учебная программа)

### Модуль 1. Разработка на Golang

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1	Модуль 1. Основы разработки на Golang	42
1.1	Занятие 1. Введение в Go: история, особенности, применение	6
1.2	Занятие 2. Структура программы: объявления, переменные, новые типы, пакеты и файлы, область видимости	6
1.3	Занятие 3. Типы данных: числа, булевы, строки, константы, юникод	6
1.4	Занятие 4. Составные типы: массивы, maps, structs, slices, динамические списки	6
1.5	Занятие 5. Функции, обработка исключений: panic, recover, defer	6
1.6	Занятие 6. Методы: Объявление методов. Методы с указателями. Встраивание структур. Значения методов и выражения. Инкапсуляция.	6
1.7	Занятие 7. Интерфейсы: Интерфейсы как Контракты. Типы интерфейсов. Приведение к интерфейсу. Значения интерфейсов. Интерфейс ошибки. Утверждение типов. Переключения интерфейсов.	6

2.	<b>Модуль 2. Продвинутая разработка на Golang</b>	<b>48</b>
2.1	Занятие 1. Асинхронность: горутины и каналы. Горутины. Каналы. Параллельный цикл. Умножение с select. Прерывания.	6
2.2	Занятие 2. Асинхронность: общедоступные переменные. Состояние гонки. Взаимные исключения. Синхронизация памяти. Ленивая инициализация. Горутины и потоки.	6
2.3	Занятие 3. Пакеты и библиотеки: Пути импортов. Объявление пакета. Объявления импортов. Пустые импорты. Пакеты и названия.	6
2.4	Занятие 4. Тестирование: Тесты функций. Покрытие. Функции оценки производительности. Профилирование.	6
2.5	Занятие 5. Рефлексия: Reflect.Type и reflect.Value. Рекурсивный вывод типов. Установка переменных с помощью reflect.Value. Доступ к тэгам полей структур. Вывод методов типа.	6
2.6	Занятие 6. Пакет unsafe: Unsafe.Sizeof, Alignof, Offsetof. Unsafe.Pointer. Вызов кода на C в cgo.	6
2.7	Занятие 7. Сборщик мусора.	6
2.8	Занятие 8. Шаблоны проектирования.	6
3.	<b>Итоговая проектная работа</b>	<b>35</b>
	<b>Итого</b>	<b>125</b>

### Модуль 2. Технологии программирования и операционные системы

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак. час.
1.	<b>Семейство ОС Unix. Современные инструменты для работы в консоли</b>	21
	Теория:	6

	<ol style="list-style-type: none"> <li>1. Семейство ОС Unix, основные дистрибутивы и их отличия.</li> <li>2. Основные командные интерпретаторы, их отличия.</li> <li>3. Утилита sed, язык awk.</li> <li>4. Использование Python как заменителя shell.</li> <li>5. Jupyter и TmpNb как его модификация.</li> <li>6. Терминальные мультиплексоры, автоматизация работы с ними.</li> </ol>	
	Практика	3
	Самостоятельная работа	9
	Выполнение контрольных заданий	3
<b>2.</b>	<b>Системы контроля версий</b>	<b>21</b>
	Теория: <ol style="list-style-type: none"> <li>1. Системы контроля версий в современных проектах.</li> <li>2. CVS – первая система контроля версий.</li> <li>3. SVN.</li> <li>4. Современные VCS – Git, Mercurial и работа с ними.</li> <li>5. Автоматизация работы с Git.</li> </ol>	6
	Практика	3
	Самостоятельная работа	9
	Выполнение контрольных заданий	3
<b>3.</b>	<b>Виртуализация</b>	<b>21</b>
	Теория: <ol style="list-style-type: none"> <li>1. Виртуализация. Её виды (контейнерная, на уровне ОС).</li> <li>2. Современные платформы виртуализации и работа с ними.</li> <li>3. Автоматизация работы с виртуальными окружениями.</li> </ol>	6
	Практика	3
	Самостоятельная работа	9
	Выполнение контрольных заданий	3
<b>4.</b>	<b>Непрерывная интеграция</b>	<b>21</b>
	Теория: <ol style="list-style-type: none"> <li>1. Непрерывная интеграция (continuous integration) и её этапы.</li> <li>2. Основные инструменты CI и их связь с VCS.</li> </ol>	6
	Практика	3
	Самостоятельная работа	9

	Выполнение контрольных заданий	3
<b>5.</b>	<b>Оркестрация контейнеров. Kubernetes</b>	<b>21</b>
	Теория: 1. Компьютерные сети, модель OSI, её уровни. 2. Сетевые протоколы обмена информацией.	6
	Практика	3
	Самостоятельная работа	9
	Выполнение контрольных заданий	3
<b>6.</b>	<b>Итоговая проектная работа</b>	<b>30</b>
	<b>Итого</b>	<b>135</b>

### Модуль 3. Fullstack-разработка

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
<b>1.</b>	<b>Frontend-разработка</b>	<b>39</b>
	Теория: 1. HTML/CSS 2. JS 3. DOM 4. React 5. React hooks 6. Redux	12
	Практика: 1. Выбор проекта 2. Подготовка структуры проекта в figma 3. Подготовка главной страницы проекта 4. Frontend проекта	12
	Самостоятельная работа	12
	Выполнение контрольных заданий	3
<b>2.</b>	<b>Backend-разработка</b>	<b>33</b>
	Теория: 1. MVC модель 2. Наиболее популярные фреймворки для Backend-разработки 3. Работа с данными	10

	<ul style="list-style-type: none"> <li>4. Сериализация json</li> <li>5. Слияние React приложения с API</li> <li>6. Аутентификация</li> </ul>	
	<p>Практика:</p> <ul style="list-style-type: none"> <li>1. Модель данных Backend</li> <li>2. Авторизация</li> <li>3. Собрать вместе Frontend + Backend проекта</li> </ul>	10
	Самостоятельная работа	10
	Выполнение контрольных заданий	3
<b>3.</b>	<b>DevOps</b>	<b>14</b>
	<p>Теория:</p> <ul style="list-style-type: none"> <li>1. Облачные сервисы</li> <li>2. Настройка сервера</li> </ul>	2
	Практика: настройка сервера	2
	Самостоятельная работа	2
	Выполнение контрольных заданий	2
<b>4.</b>	<b>NoCode-технологии</b>	<b>14</b>
	<p>Теория:</p> <ul style="list-style-type: none"> <li>1. Что такое NoCode, LowCode, ZeroCode</li> <li>2. Где No Code заменил программистов</li> <li>3. Платформы для NoCode</li> <li>4. Затраты на NoCode</li> <li>5. Как выбрать No-code платформу для IT проекта</li> </ul>	2
	Практика: альтернатива проекту при помощи NoCode-технологий	2
	Самостоятельная работа	2
	Выполнение контрольных заданий	2
<b>5.</b>	<b>Итоговая проектная работа</b>	<b>35</b>
	<b>Итого</b>	<b>135</b>

<b>4.</b>	<b>Итоговая квалификационная работа</b>	<b>35</b>
	Самостоятельная работа.	35

<b>Итого</b>	<b>430</b>
--------------	------------

Содержание дополнительной профессиональной программы направлено на достижение результатов её целей (планируемых результатов).

## **12. Организационно-педагогические условия реализации дополнительной профессиональной программы**

### **12.1. Требования к квалификации педагогических кадров/представителей предприятий и организаций, обеспечивающих реализацию образовательного процесса**

Требования к образованию педагогических и иных работников, а также (при наличии) требования к освоению ими дополнительных профессиональных программ, опыту работы в области профессиональной деятельности, соответствующей направленности ДПП: высшее образование, трудовой и (или) педагогический стаж от полугода, наличие опыта практической работы в сфере деятельности, совпадающей с направлением преподавания.

### **12.2. Требования к материально-техническим условиям**

Перечень кабинетов, лабораторий, мастерских, тренажеров и др., обеспечивающих проведение всех предусмотренных программой видов занятий:

<b>Наименование специализированных аудиторий, кабинетов, лабораторий</b>	<b>Вид занятий</b>	<b>Наименование оборудования, программного обеспечения</b>
Система дистанционного обучения провайдера массовых открытых онлайн курсов	Лекции	<p>Слушателю необходимо наличие доступа в сеть интернет, компьютер.</p> <p>Преподавателю курса необходимо наличие доступа администратора курса на LMS-платформе к материалам курса.</p>
Информационно-коммуникационная платформа дистанционных семинаров	Практические занятия (дистанционные семинары)	<p>Слушателю необходимо наличие доступа в сеть интернет, компьютер.</p> <p>Преподавателю курса необходимо оборудование для проведения дистанционных семинаров (вебинаров), качественный отказоустойчивый доступ в сеть интернет.</p>

Система дистанционного обучения провайдера массовых открытых онлайн курсов	Самостоятельная работа	Наличие компьютера и доступа в сеть интернет.
Система дистанционного обучения провайдера массовых открытых онлайн курсов	Промежуточный, рубежный контроль, Итоговая аттестация	Наличие компьютера и доступа в сеть интернет.

### 12.3. Требования к информационным и учебно-методическим условиям

Список литературы:

#### МОДУЛЬ 1:

Основная литература

1. Свердлов, С. З. Языки программирования и методы трансляции / С. З. Свердлов. — 4-е изд., стер. — Санкт-Петербург : Лань, 2024. — 564 с. — ISBN 978-5-507-48776-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/362948> (дата обращения: 21.12.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Марк, С. Программирование на Go. Разработка приложений XXI века : учебное пособие / С. Марк ; перевод с английского А. Н. Киселёв. — Москва : ДМК Пресс, 2013. — 580 с. — ISBN 978-5-94074-854-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/69944> (дата обращения: 21.12.2023). — Режим доступа: для авториз. пользователей.

#### МОДУЛЬ 2:

Основная литература

1. Орещенков, И. С. Операционные системы. Bodhi Linux 6.0: установка, настройка, эксплуатация / И. С. Орещенков. — Санкт-Петербург : Лань, 2023. — 364 с. — ISBN 978-5-507-44988-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/276659> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
2. Сейерс, Э. Х. Docker на практике / Э. Х. Сейерс, А. Милл ; перевод с английского Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 516 с. — ISBN 978-5-97060-772-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131719> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
3. Васильева, М. А. Система контроля версий. Основы командной разработки / М. А. Васильева, К. М. Филипченко. — Санкт-Петербург : Лань, 2022. — 144 с. — ISBN 978-5-507-44630-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/261089> . — Режим доступа: для авториз. пользователей.

### Дополнительная литература

1. Моуэт, Э. Использование Docker / Э. Моуэт ; научный редактор А. А. Маркелов ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2017. — 354 с. — ISBN 978-5-97060-426-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/93576> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
2. «Операционные системы. Программное обеспечение : учебник / составитель Т. П. Куль. — Санкт-Петербург : Лань, 2020. — 248 с. — ISBN 978-5-8114-4290-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131045> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
3. Донцов, В. П. Linux на примерах : руководство / В. П. Донцов, И. В. Сафин. — Санкт-Петербург : Наука и Техника, 2017. — 352 с. — ISBN 978-5-94387-742-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/101550> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
4. Фишерман, Л. В. Git. Практическое руководство. Управление и контроль версий в разработке программного обеспечения : руководство / Л. В. Фишерман. — Санкт-Петербург : Наука и Техника, 2021. — 304 с. — ISBN 978-5-94387-547-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/191470> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

## МОДУЛЬ 3:

### Основная литература

1. Заяц, А. М. Проектирование и разработка WEB-приложений. Введение в frontend и backend разработку на JavaScript и node.js : учебное пособие для вузов / А. М. Заяц, Н. П. Васильев. — 3-е изд., стер. — Санкт-Петербург : Лань, 2021. — 120 с. — ISBN 978-5-8114-7042-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/154380> (дата обращения: 22.06.2023). — Режим доступа: для авториз. пользователей.
2. Диков, А. В. Клиентские технологии веб-программирования: JavaScript и DOM : учебное пособие / А. В. Диков. — Санкт-Петербург : Лань, 2020. — 124 с. — ISBN 978-5-8114-4074-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/126934> (дата обращения: 22.06.2023). — Режим доступа: для авториз. пользователей.
3. Государев, И. Б. Введение в веб-разработку на языке JavaScript : учебное пособие / И. Б. Государев. — Санкт-Петербург : Лань, 2022. — 144 с. — ISBN 978-5-8114-3539-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/206588> (дата обращения: 22.06.2023). — Режим доступа: для авториз. пользователей.

### Дополнительная литература

1. Диков, А. В. Клиентские технологии веб-дизайна. HTML5 и CSS3 / А. В. Диков. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 188 с. — ISBN 978-5-507-46740-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/318443> (дата обращения: 22.06.2023). — Режим доступа: для авториз. пользователей.
2. Сейерс, Э. Х. Docker на практике / Э. Х. Сейерс, А. Милл ; перевод с английского Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 516 с. — ISBN 978-5-97060-772-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

<https://e.lanbook.com/book/131719> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

3. Моуэт, Э. Использование Docker / Э. Моуэт ; научный редактор А. А. Маркелов ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2017. — 354 с. — ISBN 978-5-97060-426-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/93576> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

#### 12.4. Общие требования к организации образовательного процесса

Форма обучения – очная / заочная, с применением дистанционных технологий.

Режим обучения – 2 раз(а) в неделю по 2 академических часа.

Преподаватель проводит практические занятия дистанционно в форме вебинаров с использованием платформы ZOOM (или аналогичной).

Видеоматериалы и дополнительные материалы выкладываются на обучающую платформу Skillspace.

Самостоятельная работа выполняется слушателем в удобном для слушателя режиме.

В Таблице ниже описаны образовательные технологии.

№ п/п	Вид занятия	Форма проведения занятий	Цель
1	Лекция	Изложение материала посредством лекций, обсуждение общих вопросов по тематике курса.	Ознакомление слушателей с базовым материалом по тематике курса
2	Практическое занятие	Выполнение практических заданий, получение обратной связи от преподавателя.	Практическое освоение теоретических знаний, Разбор ситуаций, а также углубление знаний по курсу
3	Самостоятельная работа	Самостоятельное изучение дополнительных материалов и литературы. Выполнение тренировочных тестов и заданий.	Углубление знаний по курсу. Применение знаний к своей организации / проекту
4	Выполнение контрольных заданий	Выполнение тестов, проверочных заданий.	Практическое освоение теоретических знаний, контроль освоения материалов
5	Итоговая аттестация	Подготовка итоговой аттестационной работы.	Практическое освоение теоретических знаний, контроль освоения материалов. Получение практического результата в своих организациях / проектах

#### 13. Формы аттестации и оценочные материалы по программе

Оценка качества освоения программы проводится по двухбалльной системе: «зачтено», «не зачтено» по результатам промежуточного контроля (тестирование, проверочные задания на

взаимную оценку), контроля посещаемости практических занятий (вебинаров) и результатам итоговой аттестации.

Слушатель считается аттестованным в случае положительных результатов работы (не менее 70% баллов от итоговой оценки) в процессе обучения и успешной сдачи экзамена. При этом баллы за экзамены начисляются только при достижении 50% порога при прохождении каждого экзаменационного испытания. После аттестации слушатель получает оценку «отлично», если набрано не меньше 80% баллов от возможного максимума, «хорошо», если набрано не меньше 65%, «удовлетворительно», если набрано не меньше 40%, в противном случае слушатель курс не сдает.

Результат тестирования, решения проверочных заданий и написания кода проверяется автоматически системой на образовательной платформе. Итоговая аттестационная работа проверяется преподавателем

Составляющие процесса обучения, которые оцениваются в ходе обучения, и их вклад в итоговую оценку представлены в Таблице ниже.

Таблица – Составляющие процесса обучения

	Основные показатели оценки	Вклад в итоговую оценку
1	Основной курс обучения на образовательной платформе	80%
3	Итоговая аттестация	20%

### Оценочные материалы:

#### 1. Пример заданий текущего контроля к модулю 1 «Разработка на Golang»

- 1) В Unix есть утилита командной строки echo. Она выводит в стандартный вывод первый аргумент. Написать свой аналог.
- 2) В некотором сервисе, на некотором сервере основные данные самых активных пользователей лежат в файле, который обновляется с определённой периодичностью. В сервис часто поступают запросы на отсортированный список самых активных пользователей, живущих в определённом городе. Один из разработчиков получил задачу по написанию кэша для таких запросов. Для подготовки удобного формата ему необходима функция GroupAndSort, реализацию которой он поручил вам. Функция GroupAndSort на вход подаётся динамический массив объектов типа UserData. На выходе ожидаются эти объекты, сгруппированные по городам (поле City), причём внутри каждой группы они должны быть отсортированы по полю ID в лексикографическом порядке по возрастанию.
  - Напишите свой HTTP клиент, который будет поддерживать:
  - HTTP GET
  - Установку таймаута
  - Установку количества повторных попыток соединений при таймаутах
  - Логирование ошибок в STDOUT

#### 2. Пример заданий текущего контроля к модулю 2 «Технологии программирования и операционные системы»

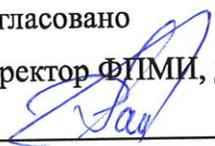
- 1) Что такое Grains и States в SaltStack? Для чего они применяются? Приведите примеры. 2. Может ли Ansible работать без мастера? А SaltStack?
- 2) В каких случаях нужно использовать Docker compose? Приведите примеры. Зачем нужен Kubernetes если есть Docker compose?
- 3) Для чего нужен Ingress? Почему не рекомендуется открывать порты из контейнеров на хост-машину?
- 4) Напишите скрипт, который создаст N независимых друг от друга Jupyter-окружений. Каждый Jupyter должен иметь свой порт. Пользователь ("админ") должен иметь возможность прервать или перезапустить один или несколько конкретных Jupyter-окружений. После перезапуска порт у Jupyter не должен изменяться. Использовать Tmux API.

### 3. Пример тестовых заданий к модулю 3 «Fullstack-разработка»

- 1) Подготовить структуру сайта в figma
- 2) Подготовить главную страницу сайта
- 3) Соединить Frontend и Backend сайта
- 4) Настроить сервер

Согласовано

Директор ФПМИ, д.ф.-м.н.

  
\_\_\_\_\_ А.М. Райгородский

## **Пояснительная записка к разработке и реализации Дополнительной программы профессиональной переподготовки «Fullstack-разработчик на Golang Pro»**

Целью программы профессиональной переподготовки является получение навыков Fullstack-разработки — разработка веб-приложения от начала до конца: и серверной части — бэкенд, и клиентской — фронтенд, а также развертывания, сборки и публикации приложения. Будут изучаться язык программирования Golang, основы frontend-разработки, популярные фреймворки для backend-разработки, основы DevOps, а также инструменты NoCode-разработки.

На практических занятиях будут отрабатываться навыки написания и тестирования программ на языке Golang, frontend-разработки на JavaScript, настройки сервера.

Задачами профессиональной переподготовки являются:

- приобретение студентами навыков Fullstack-разработки на языках Golang и JavaScript;
- овладение студентами современных практик Fullstack-разработки: использование IDE, оптимизация кода, unit-тестирование.

Целевая аудитория программы:

- ИТ-специалисты, желающие овладеть языком программирования Golang, а также современными инструментами Fullstack-разработки и DevOps;
- специалисты, желающие переквалифицироваться в Fullstack-разработчиков программного обеспечения на языках Golang и JavaScript.

Программа будет построена по очному принципу с применением дистанционных технологий (с применением электронного обучения и дистанционных образовательных технологий) в формате вебинаров, которые займут 165 ак.ч.

Преподавателями программы будут являться преподаватели МФТИ, имеющие опыт преподавания и консультирования по заявленным модулям, а также опыт практической работы в отечественных и/или зарубежных организациях в сфере деятельности, совпадающей с направлением преподавания.

### **Составители программы:**

**Райгородский Андрей Михайлович**

Доктор физико-математических наук, директор ФПМИ МФТИ

**Благодарный Евгений Владимирович**

заведующий учебно-методической лабораторией инноватики

**Ивченко Олег Николаевич**

старший преподаватель кафедры АТП ФПМИ

**Иванова Анастасия Сергеевна**

руководитель проектов учебно-методической лаборатории инноватики ФПМИ

Директор ФПМИ, д.ф.-м.н.



А.М. Райгородский

**КВАЛИФИКАЦИЯ И ОПЫТ ПРИВЛЕКАЕМОГО ПРЕПОДАВАТЕЛЬНОГО СОСТАВА ДЛЯ РЕАЛИЗАЦИИ ДОПОЛНИТЕЛЬНОЙ ПРОГРАММЫ  
ПРОФЕССИОНАЛЬНОЙ ПЕРЕПОДГОТОВКИ «Fullstack-разработчик на Golang Pro»**

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 наличии званий и ученых степеней) и т.д.	Место работы, занимаемая должность в настоящий момент, общий трудовой стаж, педагогический стаж	Опыт преподавания и консультирования по предмету, согласующемуся с направлением логотипа (перечислить), преподавательский стаж	Наличие опыта практической работы в отечественных и зарубежных организациях в сфере деятельности, совпадающей с направлением преподавания
Ивченко Олег Николаевич	<p>Высшее образование - специалист, магистратура, специальность: Прикладная математика и информатика, квалификация: Магистр, ФГБОУ ВПО «Московский физико-технический институт (национальный исследовательский университет)»</p> <p>Дополнительное профессиональное образование, курс: Математическое моделирование для анализа данных, ФГБОУ ВПО «Московский физико-технический институт (национальный исследовательский университет)»</p>	<p>Должность: Старший преподаватель Общий стаж: 4 года Педагогический стаж: 4 года.</p>	<p>Преподавательский стаж: 4 года</p>	<p>Разработчик системы Judge - системы автоматизированного тестирования Набор-приложений, системный администратор инфраструктуры больших данных в группе Яндекс-CERN</p>
Честнов Никита Николаевич	<p>Магистрант МФТИ, кафедра распознавания изображений и обработки текста.</p>	<p>Преподаватель курсов «Практикум Python» и «Технологии программирования» в МФТИ. Общий трудовой и педагогический стаж 4 года.</p>	<p>Преподаватель курсов «Практикум Python» и «Технологии программирования» в МФТИ.</p>	<p>Сотрудник лаборатории ABBYY Lab в течение 2 лет.</p>

			<p>Преподавательский стаж 1,5 года.</p>	<p>Руководитель направления разработок в VK. Опыт разработки Web-сайтов: 7 лет.</p>
<p>Нагайко Иван Игоревич, 1995</p>	<p>Руководитель направления разработок в VK. Преподаватель ФПМИ МФТИ, Кафедра алгоритмов и технологий программирования, курс «Fullstack-разработка». Общий трудовой стаж: 9 лет. Общий педагогический стаж: 9 лет.</p>	<p>Руководитель направления разработок в VK. Преподаватель ФПМИ МФТИ, Кафедра алгоритмов и технологий программирования, курс «Fullstack-разработка». Общий трудовой стаж: 9 лет. Общий педагогический стаж: 9 лет.</p>	<p>Преподаватель ВФГПШ МФТИ, ФПМИ МФТИ. Общий педагогический стаж: 9 лет.</p>	<p>Руководитель направления разработок в VK. Опыт разработки Web-сайтов: 7 лет.</p>
<p>Косицын Павел Андреевич, 1999</p>	<p>Высшее образование - бакалавриат, Прикладная математика и информатика, «Московский физико-технический институт (национальный исследовательский университет)», 2021</p>	<p>Яндекс, руководитель службы. Общий трудовой и педагогический стаж 6 лет.</p>	<p>Преподаватель курсов «Алгоритмы и структуры данных», «Архитектура компьютеров и операционных систем», «Разработка на языке GoLang» ФПМИ МФТИ. Педагогический стаж 6 лет.</p>	<p>Яндекс, руководитель службы. Опыт разработки 4,5 года.</p>