

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Ордена Трудового Красного Знамени федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский технический университет связи и информатики» (МТУСИ)

Рабочая программа дисциплины
ПРОЕКТНЫЙ ПРАКТИКУМ

Направление подготовки
09.03.04 «Программная инженерия»

Направленность (профиль) программы
«ТОП-ИТ: Разработка и сопровождение программного обеспечения»

Квалификация выпускника
Бакалавр

Форма обучения
Очная

Москва, 2025 г.

Рабочая программа составлена с учетом требований федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.04 «Программная инженерия», утвержденного Приказом Министерства образования и науки РФ от 19.09.2017 № 920, и на основании учебного плана, утвержденного Ученым советом вуза 02.10.2025, протокол №2.

Разработчик(и) программы:

И.о. заведующего кафедрой ПИ, к.т.н.

М.С. Мосева

Рабочая программа утверждена на заседании кафедры ПИ.

И.о. заведующего кафедрой ПИ, к.т.н.

М.С. Мосева

Рабочая программа актуализируется (обновляется) ежегодно, в том числе в части программного обеспечения, материально-технического обеспечения, литературы.

Рабочая программа хранится на кафедре ПИ (Программная инженерия) и в деканате факультета ИТ (Информационные технологии).

1. Цели и задачи дисциплины (модуля)

Целью освоения дисциплины «Проектный практикум» является приобретение практических навыков промышленной разработки программного обеспечения, а именно использование паттернов проектирования, в том числе микросервисной архитектуры, принципов CI/CD, инструментов контейниризации, оркестрации, мониторинга и логирования. Данные навыки дадут возможность проектировать, разворачивать и управлять контейнерными средами, в том числе при разработке On-Premise решений. Совокупность полученных обучающимся навыков позволит сократить время на его онбординг в компании. Немаловажным является практическое развитие навыков работы в команде, методологии Agile и межличностной коммуникации.

Задачами освоения дисциплины «Проектный практикум» является изучение на практических кейсах инструментов промышленной разработки:

1. Принципы Continuous Integration and Continuous Delivery (CI/CD), в том числе инструментов GitLab CI (Git Flic), Jenkins, GitHub Actions (или аналогичные российские продукты).
2. Инструменты контейнеризации и оркестрации такие как Docker, Kubernetes.
3. Инструменты мониторинга на примере Prometheus, Grafana, Elasticsearch, Kibana.
4. Принципы чистого кода.
5. Инструменты тестирования программных средств: модульное, интеграционное и нагружочное тестирование.
6. Архитектурные паттерны разработки микросервисных приложений.
7. Принципы и подходы оптимизации образов контейнеров: Docker, Podman, Nexus, Trivy.
8. Методы формулирования и описания архитектурных требований к программной системе.
9. Нормативные документы, используемые для разработки программных продуктов и автоматизированных систем.

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Проектный практикум» включена в часть, формируемую участниками образовательных отношений блока дисциплин учебного плана (Б1.В.09). Дисциплина реализуется в соответствии с требованиями ФГОС ВО, ОПОП ВО и Учебного плана по направлению подготовки 09.03.04 «Программная инженерия», профиль «ТП-ИТ: Разработка и сопровождение программного обеспечения».

Знания и умения, необходимые для успешного усвоения программы дисциплины во 2 семестре, формируются у обучающихся в результате изучения дисциплин «Введение в информационные технологии» и «Архитектура информационных систем».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 3 семестре, формируются у обучающихся в результате изучения дисциплин «Введение в информационные технологии», «Математические основы баз данных» и «Основы DevOps».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 4 семестре, формируются у обучающихся в результате изучения дисциплин «Информационные технологии и программирование», «Основы DevOps» и «Методология Agile и проектное управление».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 5 семестре, формируются у обучающихся в результате изучения дисциплин «Методология Agile и проектное управление», «Информационные технологии и программирование», «Основы DevOps» и «Структуры и алгоритмы обработки данных».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 6 семестре, формируются у обучающихся в результате изучения дисциплин «Сетевые технологии», «Структуры и алгоритмы обработки данных», «Операционные системы» и «Full-stack разработка».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 7 семестре, формируются у обучающихся в результате изучения дисциплин «Технологии и инструменты систем управления данными», «Функциональное программирование», «Full-stack разработка» и «Программная инженерия».

Знания и умения, необходимые для успешного усвоения программы дисциплины в 8 семестре, формируются у обучающихся в результате изучения дисциплин «Тестирование и отладка программного обеспечения», «Системы искусственного интеллекта» и «Управление и администрирование информационных систем».

Материалы дисциплины используются при изучении курсов «Рефакторинг баз данных и приложений», «Высоконагруженные приложения», а также при прохождении всех видов практик и выполнении выпускной квалификационной работы.

Рабочая программа дисциплины для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

3. Перечень планируемых результатов освоения дисциплины, соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной дисциплины направлено на формирование у обучающихся компетенций, представленных в таблице 1.

4. Структура и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 19 зачетных единиц (684 часа). Процесс изучения дисциплины реализуется при очной форме обучения во 2-8 семестрах. Промежуточная аттестация предусматривает зачет во 2-8 семестрах.

4.1. Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 19 зачетных единиц (684 часа), их распределение по видам работ семестрам представлено в таблице 2.

Требования к результатам освоения дисциплины (модуля)

Таблица 1

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индекс индикатора достижения компетенции	Содержание индикатора достижения компетенции	Результаты освоения индикатора достижения компетенции
1.	ПК-6	Способен участвовать в промышленной разработке программного обеспечения	ПК-6.2	Использует инструменты промышленной разработки	<p>Знает:</p> <ul style="list-style-type: none"> - принципы Continuous Integration and Continuous Delivery (CI/CD); - системы мониторинга и логирования в продуктивной среде <p>Умеет:</p> <ul style="list-style-type: none"> - настраивать потоки работ CI/CD; - работать с контейнеризацией и оркестрацией; - настраивать мониторинг в продуктивной среде <p>Владеет:</p> <ul style="list-style-type: none"> - инструментами контейнеризации и оркестрации такими как Docker, Kubernetes; - инструментами CI/CD: GitLab CI (Git Flic), Jenkins, GitHub Actions (или аналогичные российские продукты)
			ПК-6.3	Разрабатывает масштабируемый и поддерживаемый код	<p>Знает:</p> <ul style="list-style-type: none"> - принципы чистого кода, SOLID, DRY, KISS и др.; - паттерны проектирования и антипаттерны <p>Умеет:</p> <ul style="list-style-type: none"> - разрабатывать модульный и тестируемый программный код; - выполнять модульное, интеграционное и нагружочное тестирование

					<p><i>Владеет:</i></p> <ul style="list-style-type: none"> - архитектурными паттернами разработки микросервисных приложений; - системами мониторинга Prometheus, Grafana, Elasticsearch, Kibana
2.	ПК-9	Способен проектировать, развертывать и управлять контейнерными средами	ПК-9.1	Создает и оптимизирует образы контейнеров, обеспечивает безопасность и эффективность контейнеризации	<p><i>Знает:</i></p> <ul style="list-style-type: none"> - архитектуру контейнеров, принципы их реализации, безопасность контейнеров; - лучшие практики написания Dockerfile (многоступенчатая сборка, минимизация слоев) <p><i>Умеет:</i></p> <ul style="list-style-type: none"> - создавать, тестировать и публиковать образы контейнеров в реестрах (Docker Hub, GitLab Registry) <p><i>Владеет:</i></p> <ul style="list-style-type: none"> - технологиями Docker, Podman, Nexus, Trivy
3.	ПК-11	Осуществляет управление архитектурой единой информационной среды	ПК-11.1	Выявляет и согласовывает требования к архитектуре единой информационной среды	<p><i>Знает:</i></p> <ul style="list-style-type: none"> - методы проектирования архитектуры программной системы <p><i>Умеет:</i></p> <ul style="list-style-type: none"> - формулировать и описывать требования к программной системе с точки зрения архитектуры; - проверять требования на соответствие архитектуре программной системы <p><i>Владеет:</i></p> <ul style="list-style-type: none"> - нормативными документами в том числе ГОСТ Р 56939-2016 "Требования к программным продуктам", ГОСТ 34.601-90 "Автоматизированные системы. Требования к разработке и эксплуатации"

**Распределение трудоёмкости дисциплины (модуля) по видам работ
по семестрам**

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 2

Вид учебной работы	час.	Трудоёмкость							Из них практическая подготовка	
		В т.ч. по семестрам						8		
		2	3	4	5	6	7			
Общая трудоёмкость дисциплины по учебному плану	684	72	108	72	108	108	108	108	77	
1. Контактная работа:	119	19	19	17	19	17	17	11		
Аудиторная работа всего, в том числе:	112	18	18	16	18	16	16	10		
<i>лекции (Л)</i>										
<i>практические занятия (ПЗ)</i>	112	18	18	16	18	16	16	10	77	
<i>лабораторные работы (ЛР)</i>										
<i>Иная контактная работа на сессии (ИКР)</i>	7	1	1	1	1	1	1	1		
<i>Контактная работа в сессию(КРС)</i>										
2. Самостоятельная работа (СР)	565	53	89	55	89	91	91	97		
Вид промежуточного контроля:		Зачет	Зачет	Зачет	Зачет	Зачет	Зачет	Зачет		

4.2. Содержание дисциплины

Тематический план дисциплины ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 3

Наименование разделов и тем дисциплины	Всего	Аудиторная работа (по видам учебных занятий)			Самостоятельная работа (СР), ИКР, КРС, Контроль	Форма текущего контроля успеваемости/форма промежуточной аттестации
		Л	ПЗ	ЛР		
2 семестр						
Раздел 1. Планирование и анализ требований	72		4		14	Практические задания
Раздел 2. Техническое проектирование проекта			4		14	
Раздел 3. Разработка и настройка компонентов системы			6		14	
Раздел 4. Тестирование, разворачивание и защита проекта			4		12	
Всего за 2 семестр	72		18		54	
3 семестр						
Раздел 1. Планирование и анализ требований	108		4		12	Практические задания
Раздел 2. Техническое проектирование проекта			4		28	
Раздел 3. Разработка и настройка компонентов системы			6		30	
Раздел 4. Тестирование, разворачивание и защита проекта			4		20	
Всего за 3 семестр	108		18		90	
4 семестр						
Раздел 1. Планирование и анализ требований	72		2		14	Практические задания
Раздел 2. Техническое проектирование проекта			4		14	
Раздел 3. Разработка и настройка компонентов системы			6		16	
Раздел 4. Тестирование, разворачивание и защита проекта			4		12	
Всего за 4 семестр	72		16		56	
5 семестр						
Раздел 1. Планирование и анализ требований	108		4		12	Практические задания
Раздел 2. Техническое проектирование проекта			4		28	
Раздел 3. Разработка и настройка компонентов системы			6		30	
Раздел 4. Тестирование, разворачивание и защита проекта			4		20	
Всего за 5 семестр	108		18		90	
6 семестр						

Наименование разделов и тем дисциплины	Всего	Аудиторная работа (по видам учебных занятий)			Самостоятельная работа (СР), ИКР, КРС, Контроль	Форма текущего контроля успеваемости/форма промежуточной аттестации
		Л	ПЗ	ЛР		
Раздел 1. Планирование и анализ требований	108		2		12	Практические задания
Раздел 2. Техническое проектирование проекта			4		28	
Раздел 3. Разработка и настройка компонентов системы			6		32	
Раздел 4. Тестирование, разворачивание и защита проекта			4		20	
Всего за 6 семестр	108		16		92	
7 семестр						
Раздел 1. Планирование и анализ требований	108		2		12	Практические задания
Раздел 2. Техническое проектирование проекта			4		28	
Раздел 3. Разработка и настройка компонентов системы			6		32	
Раздел 4. Тестирование, разворачивание и защита проекта			4		20	
Всего за 7 семестр	108		16		92	
8 семестр						
Раздел 1. Планирование и анализ требований	108		2		12	
Раздел 2. Техническое проектирование проекта			2		32	
Раздел 3. Разработка и настройка компонентов системы			4		34	
Раздел 4. Тестирование, разворачивание и защита проекта			2		20	
Всего за 8 семестр	108		10		98	
Объем дисциплины (в академических часах)	684				Зачет	
Объем дисциплины (в зачетных единицах)	19					

4.3. Лекции/лабораторные/практические занятия

Содержание лекций/лабораторного практикума/практических занятий

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 4

№ п/п	Название раздела, № и название лекций/ лабораторных/ практических занятий	Код(ы) формируемых индикаторов компетенций
1.	Раздел 1. Планирование и анализ требований Практическое занятие 1. Сбор бизнес-требований и описание назначение продукта. Формулирование критериев оценки успешности проекта.	PK-11.1

№ п/п	Название раздела, № и название лекций/ лабораторных/ практических занятий	Код(ы) формируемых индикаторов компетенций
	Практическое занятие 2. Выбор методологии работы и описание пользовательских сценариев.	ПК-11.1
	Практическое занятие 2. Формирование документации с информацией о функциональных и нефункциональных требований.	ПК-6.3, ПК-11.1
2.	Раздел 2. Техническое проектирование проекта	
	Практическое занятие 3. Проектирование основных компонентов. Определение принципов взаимодействия компонентов между собой. Определение зависимостей от внешних программных средств.	ПК-6.2
	Практическое занятие 4. Разработка прототипа пользовательского интерфейса. Настройка среды и инфраструктуры разработки. Продумывание деталей будущих классов, методов или функций.	ПК-6.3
3.	Раздел 3. Разработка и настройка компонентов системы	
	Практическое занятие 5. Разработка ядра продукта. Разработка с использованием стека Kubernetes (после 3 семестра), Docker, CI/CD (начиная с 3 семестра).	ПК-6.2, ПК-6.3, ПК-9.1
	Практическое занятие 6. Работа с данными. Взаимодействие с API и внешними системами. Проектирование и реализация оптимальных для разрабатываемого проекта алгоритмы, структуры данных и логики работы внутри каждого модуля.	ПК-6.2, ПК-6.3, ПК-9.1
	Практическое занятие 7. Разработка инфраструктуры, проектирование и разработка оптимального в рамках разрабатываемого проекта хранилища данных и интеграция.	ПК-6.2, ПК-9.1
4.	Раздел 4. Тестирование, разворачивание и защита проекта	
	Практическое занятие 8. Тестирование и отладка. Деплой и документация.	ПК-9.1
	Практическое занятие 9. Защита проекта.	ПК-6.2, ПК-6.3, ПК-9.1, ПК-11.1

5. Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся направлена на углубленное изучение разделов и тем рабочей программы и предполагает изучение литературных источников, выполнение домашних заданий и проведение исследований разного характера. Работа основывается на анализе литературных источников и материалов, публикуемых в интернете, а также реальных речевых и языковых фактов, личных наблюдений. Также самостоятельная работа включает подготовку и анализ материалов по темам пропущенных занятий.

Самостоятельная работа по дисциплине включает следующие виды деятельности:

- поиск (подбор) и обзор литературы, электронных источников информации по индивидуально заданной проблеме курса, написание доклада, исследовательской работы по заданной проблеме;
- выполнение задания по пропущенной или плохо усвоенной теме;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашней контрольной работы (решение задач, выполнение упражнений);
- изучение материала, вынесенного на самостоятельную проработку (отдельные темы, параграфы);
- подготовка к практическим занятиям;
- подготовка к зачету.

Самостоятельная работа обучающихся над усвоением материала по дисциплине может выполняться в помещении для самостоятельной работы МТУСИ, посредством использования электронной библиотеки и ЭИОС.

5.1. Контрольные вопросы и задания (для самостоятельного изучения)

1. Объясните основные принципы Continuous Integration, Delivery и Continuous Deployment. В чем их ключевые отличия и бизнес-выгоды?
2. Опишите, как выстроить ступенчатый CI/CD-конвейер в GitLab CI (или Git Flic). Какие блоки («стадии») при этом необходимы и для чего?
3. Расскажите, как вы создавали и настраивали Jenkins Pipeline. Какими плагинами пользовались, как описывали стадии (Declarative vs Scripted)?
4. Приведите пример workflow в GitHub Actions (или аналогичном российском продукте), включающего сборку, тестирование и деплой контейнера.
5. Как вы организуете сборку Docker-образа и пуш в реестр в рамках CI-процесса? Опишите Dockerfile, инструкции CI-конфига и работу с кэшированием.
6. Опишите ваши шаги при деплее контейнеризованного приложения в Kubernetes: от создания манифестов (Deployment, Service) до rollout'a.
7. Расскажите о механизмах «канареичного» (canary) или «синего-зеленого» (blue-green) деплоя в Kubernetes. Как вы их реализовывали?
8. Как вы настраиваете горизонтальное автоскейлингование (HPA) в Kubernetes и какие метрики для этого используете?
9. Опишите, какие инструменты мониторинга (Prometheus, Zabbix, Nagios, Statping и т. п.) вы применяли в продакшн-среде.
10. Как вы строите систему логирования и агрегации логов (ELK/EFK, Loki, Graylog)? Какие шаги по парсингу, индексации и хранению логов выполняете?
11. Какие настройки readiness и liveness probes вы используете в Kubernetes и как они влияют на стабильность приложения?
12. Как вы реализовали оповещения (alerting) на основе метрик и логов? Опишите настройку Alertmanager, оповещения в Slack/Telegram/email.

13. Как вы храните и защищаете секреты (пароли, токены) в CI/CD-конвейерах и в Kubernetes (Secrets, Vault, SOPS)?
14. Опишите процесс отката (rollback) в случае неудачного деплоя: какие механизмы CI/CD-системы и Kubernetes для этого применяете?
15. Расскажите основные принципы чистого кода. Какие приёмы именования, структурирования функций и классов вы применяете на практике?
16. Перечислите и поясните пять принципов SOLID. Приведите пример реальной задачи, где нарушение одного из принципов привело к проблемам в коде.
17. Объясните принципы DRY, KISS и YAGNI. Как вы понимаете баланс между «не повторяться» и «не усложнять»?
18. Назовите три наиболее часто используемых вами паттерна проектирования (например, Factory, Strategy, Observer). В каком случае вы бы выбрали именно их и почему?
19. Приведите примеры трёх распространённых антипаттернов (God Object, Spaghetti Code, Golden Hammer и др.). Как вы их обнаруживаете и устраняете?
20. Какие приёмы/архитектурные решения вы используете, чтобы код оставался модульным и легко тестируемым? Расскажите про Dependency Injection и Inversion of Control.
21. Опишите ваш подход к написанию модульных тестов: выбор фреймворка, организация тестов, работа с mock-объектами.
22. Как вы строите интеграционные тесты? Чем они отличаются от модульных, какие инструменты и среды (контейнеры, тестовые БД) применяете?
23. Расскажите про нагрузочное тестирование: какие инструменты (JMeter, Gatling, Locust и т. п.) вы использовали, как моделировали сценарии и анализировали результаты?
24. Назовите основные архитектурные паттерны в микросервисных приложениях: API Gateway, Circuit Breaker, Saga, CQRS, Event Sourcing и др. В каком случае вы бы применили каждый из них?
25. Как вы проектируете взаимодействие микросервисов: синхронное (REST/gRPC) и асинхронное (Kafka/RabbitMQ)? Как это влияет на тестирование и отладку?
26. Опишите, как вы настраиваете сбор метрик с приложений для Prometheus: какие метрики важны, как экспортить, какие библиотеки используете.
27. Как вы строите дашборды в Grafana на основе метрик Prometheus? Какие ключевые панели вы рекомендуете для микросервисной платформы?
28. Расскажите про систему логирования с Elasticsearch и Kibana: как формируются индексы, настраиваются шаблоны, какие запросы и визуализации вы создаёте.
29. Какие алERTы вы настраиваете в Alertmanager (Prometheus)? Приведите пример правила и сценария оповещения при деградации сервиса.
30. Опишите архитектуру контейнера на уровне Linux: какие роли играют namespaces, groups и union-файловые системы (OverlayFS)?
31. В чем принципиальное отличие Docker и Podman по архитектуре запуска контейнеров и обеспечению rootless-режима?
32. Объясните, как формируются слои контейнерного образа, как Docker хранит и кэширует эти слои при сборке.
33. Какие механизмы изоляции и ограничений ресурсов реализуют cgroups и namespaces? Приведите примеры настроек лимитов CPU, памяти и сети.
34. Перечислите основные практики оптимизации Dockerfile: многоступенчатая сборка, минимизация числа слоев, использование лёгких базовых образов.
35. Как выстроить порядок инструкций в Dockerfile, чтобы максимально эффективно использовать кеш сборки? Как работает файл .dockerignore?
36. Приведите пример многоступенчатого Dockerfile для приложения: отдельные стадии сборки и исполнения, копирование бинарника или артефактов.
37. Расскажите о наиболее распространённых векторах уязвимостей в контейнерных образах и как их обнаружить с помощью Trivy. Как выстраивать политику блокировки по уровню CVE?

38. Какие приёмы вы используете для повышения безопасности контейнера: запуск от непrivилегированного пользователя, сброс CAP-прав, read-only root-fs, Seccomp-профили?
39. Как вы настраивали приватный реестр контейнеров на Nexus: создание репозиториев, управление правами доступа, проксижение Docker Hub?
40. Опишите процесс локального тестирования образа: проверка работоспособности через docker run (или podman run), использование healthcheck, проверка монтирования томов и сетевого взаимодействия.
41. Как вы автоматизируете сборку и публикацию образов в Docker Hub или GitLab Container Registry: теги, аутентификация, CI-задачи?
42. В каких сценариях вы предпочли бы Podman вместо Docker и как настраиваете alias-команды (docker → podman)? Какие преимущества даёт podman-socket?
43. Как интегрировать Trivy в ваш CI-конвейер: стадии сканирования, отчёты о уязвимостях, условие провала сборки?
44. Опишите процесс подписания и верификации контейнерных образов (Docker Content Trust, Notary или cosign). Какие угрозы закрывает этот механизм?
45. Опишите процесс сбора и формулирования требований к системе с архитектурной точки. Как вы разделяете функциональные, нефункциональные требования и ограничения?
46. Какие методы и нотации вы используете для документирования архитектурных требований (UML, SysML, C4, ArchiMate и др.)? Приведите пример.
47. Как вы определяете качественные атрибуты (производительность, масштабируемость, отказоустойчивость и т.п.) и превращаете их в конкретные архитектурные сценарии?
48. Расскажите о способе обеспечения тестируемости требований: какие критерии (SMART, приемочные тесты) вы применяете?
49. Как вы проверяете требования на непротиворечивость между собой и с существующей архитектурой? Описывайте приёмы трассировки и валидации.
50. Какие шаги вы предпринимаете при анализе влияния изменений требований на архитектуру? Как организуете управление этими изменениями?
51. Перечислите и кратко опишите методы проектирования архитектуры программной системы: Attribute-Driven Design (ADD), ATAM, CBAM, Quality Attribute Workshop и др.
52. Расскажите, как проходит аттестация архитектуры по методу ATAM: основные фазы, участники, артефакты.
53. Опишите этапы ADD-метода: как вы выбираете архитектурные паттерны и принимаете решения на основе качественных атрибутов?
54. Какие нормативные документы и стандарты вы применяете при оформлении требований и архитектурной документации (IEEE 830/29148, ISO/IEC/IEEE 42010, ISO 25010 и др.)?
55. Как вы формируете архитектурные взгляды (viewpoints) и представления (views) в соответствии со стандартом ISO/IEC/IEEE 42010?
56. Приведите пример матрицы трассировки, связывающей требования, архитектурные решения и тестовые сценарии.
57. Как вы проверяете соответствие архитектуры нормативам безопасности и защите персональных данных (например, ISO/IEC 27001, GDPR) на этапе выработки требований?

6. Оценочные материалы для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Оценочные материалы (оценочные средства) для проведения текущего контроля и промежуточной аттестации по дисциплине «Проектный практикум» прилагаются.

6.1. Перечень видов оценочных средств

1. Теоретические вопросы к промежуточному контролю.
2. Практические задания и задачи к промежуточному контролю.

7. Учебно-методическое и информационное обеспечение дисциплины (модуля)

7.1. Основная литература

1. Кознов, Д. В. Введение в программную инженерию : учебное пособие / Д. В. Кознов. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 305 с. — ISBN 978-5-4497-2385-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133932.html> (дата обращения: 19.06.2025). — Режим доступа: для авторизир. пользователей

7.2. Дополнительная литература

1. Рошин, П. Г. Командная разработка программного обеспечения с помощью системы контроля версий GIT: конспект лекций : учебное пособие / П. Г. Рошин. — Москва : Национальный исследовательский ядерный университет «МИФИ», 2022. — 106 с. — ISBN 978-5-7262-2846-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/132682.html> (дата обращения: 19.06.2025). — Режим доступа: для авторизир. пользователей

8. Требования к условиям реализации дисциплины (модуля)

8.1. Общесистемные требования

Электронная информационно-образовательная среда ФГБОУ ВО «МТУСИ»

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде МТУСИ из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет», как на территории МТУСИ, так и вне ее:

<https://mtuci.ru/> – адрес официального сайта университета;

<https://mtuci.ru/education/eios/> – электронная информационно-образовательная среда МТУСИ;

<http://elib.mtuci.ru/catalogue/> – каталог электронной библиотеки МТУСИ.

Электронно-библиотечные системы (электронные библиотеки)

№ п/п	Ссылка на информационный ресурс	Наименование образовательного ресурса	Доступность
1	http://iprbookshop.ru/	ЭБС IPRSmart	Индивидуальный неограниченный доступ из любой точки, в которой имеется доступ к сети Интернет
2	https://e.lanbook.com/	ЭБС ЛАНЬ	
3	https://znamium.com/	ЭБС ZNANIUM	
4	http://book.ru/	ЭБС BOOK.RU	
5	https://urait.ru/	Образовательная платформа Юрайт	
6	https://www.elibrary.ru/defaultx.asp	Научная электронная библиотека	

8.2. Материально-техническое и учебно-методическое обеспечение дисциплины (модуля)

МТУСИ располагает материально-техническим обеспечением образовательной деятельности (помещениями и оборудованием) для реализации программы дисциплины (модуля).

Для осуществления образовательного процесса по дисциплине используются учебные аудитории для проведения учебных занятий, оснащенные оборудованием и техническими средствами обучения:

1. Учебная аудитория для проведения практических занятий, укомплектованная специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории: наборами демонстрационного оборудования и учебно-наглядных пособий, обеспечивающими тематические иллюстрации, соответствующие рабочей программе дисциплины, оснащенная компьютерной техникой.

2. Учебная аудитория для проведения консультаций, текущего контроля и промежуточной аттестации, оснащенная компьютерной техникой.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду МТУСИ.

8.3. Необходимый комплект лицензионного программного обеспечения

МТУСИ обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства:

№	Наименование	Лицензия	Вид лицензии
1	PostgreSQL		
2	pgAdmin		
3	Git		
4	PyCharm		
5	VSCode		
6	Docker		
7	Kubernetes		
8	Podman		
9	Nexus		
10	Trivy		
11	Prometheus		
12	Grafana		
13	Elasticsearch		
14	Kibana		
15	Git Flic		

8.4. Перечень профессиональных баз данных и информационных справочных систем

Современные профессиональные базы данных:

1. Федеральный портал «Российское образование»: [Электронный ресурс] – Режим доступа: <http://www.edu.ru/> (открытый доступ)
2. Федеральный центр информационно-образовательных ресурсов: [Электронный ресурс] – Режим доступа: <http://fcior.edu.ru/> (открытый доступ)

Информационные справочные системы:

1. Портал Федеральных государственных образовательных стандартов высшего образования: <https://fgosvo.ru>
2. Справочно-правовая система Консультант – Режим доступа: <https://www.consultant.ru/>
3. Справочно-правовая система Гарант – Режим доступа: <https://www.garant.ru/>

9. Методические рекомендации для участников образовательного процесса, определяющие особенности освоения учебной дисциплины обучающимся с ограниченными возможностями здоровья (ОВЗ) в условиях инклюзивного образования

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе проведения учебных занятий преподавателю следует стремиться к созданию гибкой и вариативной организационно-методической системы обучения, адекватной образовательным потребностям данной категории обучающихся, которая позволит не только обеспечить преемственность систем общего (инклюзивного) и высшего образования, но и будет способствовать формированию у них компетенций, предусмотренных ФГОС ВО, ускорит темпы профессионального становления, а также будет способствовать их социальной адаптации.

В процессе преподавания учебной дисциплины необходимо способствовать созданию на каждом занятии толерантной социокультурной среды, необходимой для формирования у всех обучающихся гражданской, правовой и профессиональной позиции соучастия, готовности к полноценному общению, сотрудничеству, способности толерантно воспринимать социальные, личностные и культурные различия, в том числе и характерные для обучающихся с ОВЗ.

Посредством совместной, индивидуальной и групповой работы необходимо способствовать формированию у всех обучающихся активной жизненной позиции и развитию способности жить в мире разных людей и идей, а также обеспечить соблюдение обучающимися их прав и свобод и признание права другого человека, в том числе и обучающихся с ОВЗ на такие же права.

В процессе обучения студентов с ОВЗ в обязательном порядке необходимо учитывать рекомендации службы медико-социальной экспертизы или психолого-медицинско-педагогической комиссии, обусловленные различными стартовыми возможностями данной категории обучающихся (структурой, тяжестью, сложностью дефектов развития).

В процессе овладения обучающимися с ОВЗ компетенциями, предусмотренными рабочей программой дисциплины (РПД), преподавателю следует неукоснительно руководствоваться следующими принципами построения инклюзивного образовательного пространства:

- принцип индивидуального подхода, предполагающий выбор форм, технологий, методов и средств обучения и воспитания с учетом индивидуальных образовательных потребностей каждого из обучающихся с ОВЗ, учитывающий различные стартовые возможности данной категории обучающихся (структуру, тяжесть, сложность дефектов развития);

- принцип вариативной развивающей среды, который предполагает наличие в процессе проведения учебных занятий и самостоятельной работы обучающихся необходимых развивающих и дидактических пособий, средств обучения, а также организацию безбарьерной среды, с учетом структуры нарушения в развитии (нарушения опорно-двигательного аппарата, зрения, слуха и др.);

- принцип вариативной методической базы, предполагающий возможность и способность использования преподавателем в процессе овладения обучающимися с ОВЗ данной учебной дисциплиной, технологий, методов и средств работы из смежных областей, применение методик и приемов тифло-, сурдо-, олигофренопедагогики, логопедии;

- принцип модульной организации основной образовательной программы, подразумевающий включение в основную образовательную программу модулей из специальных коррекционных программ, способствующих коррекции и реабилитации обучающихся с ОВЗ, а также необходимости учета преподавателем конкретной учебной дисциплины их роли в повышении качества профессиональной подготовки данной категории обучающихся;

- принцип самостоятельной активности обучающихся с ОВЗ, предполагающий обеспечение самостоятельной познавательной активности данной категории обучающихся посредством дополнения раздела РПД «Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)» заданиями, учитывающими различные стартовые возможности данной категории обучающихся (структуру, тяжесть, сложность дефектов развития).

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе проведения учебных занятий преподавателю необходимо осуществлять учет наиболее типичных проявлений психоэмоционального развития, поведенческих и характерологических особенностей, свойственных обучающимся с ОВЗ: повышенной утомляемости, лабильности или инертности эмоциональных реакций, нарушений психомоторной сферы, недостаточное развитие вербальных и невербальных форм коммуникации. В отдельных случаях следует учитывать их склонность к перепадам настроения, аффективность поведения, повышенный уровень тревожности, склонность к проявлениям агрессии, негативизма и т.д.

С целью коррекции и компенсации вышеперечисленных типичных проявлений психоэмоционального развития, поведенческих и характерологических особенностей, свойственных обучающимся с ОВЗ, преподавателю в ходе проведения учебных занятий следует использовать здоровьесберегающие технологии по отношению к данной категории обучающихся, в соответствии с рекомендациями службы медико-социальной экспертизы или психолого-медицинско-педагогической комиссии.

В группах, в состав которых входят обучающиеся с ОВЗ различной нозологии, при проведении учебных занятий преподавателю следует обратить особое внимание на следующее:

- при обучении студентов с дефектами слуха: на создание безбарьерной среды общения, которая определяется наличием у обучающихся данной категории индивидуальных слуховых аппаратов (или кохлеарных имплантатов), наличие технических средств, обеспечивающих передачу информации на зрительной основе (средств статической и динамической проекции, видеотехники, лазерных дисков, адаптированных компьютеров и т.д.);

- присутствие на занятиях тыютора, владеющего основами разговорной, дактильной и калькирующей жестовой речи;

- при обучении студентов с дефектами зрения: на наличие повышенной освещенности (не менее 1000 люкс) или локального освещения не менее 400-500 люкс, а также наличие оптических средств (лупы, специальных устройств для использования компьютера, телевизионных увеличителей, аудио оборудования для прослушивания «говорящих книг»), наличие комплекта письменных принадлежностей (бумага для письма рельефно-точечным шрифтом Брайля), учебных материалов с использованием шрифта Брайля, звукоусиливающей аппаратуры индивидуального пользования;

- при обучении студентов с нарушениями опорно-двигательной функции: предусматривается применение специальной компьютерной техники с соответствующим программным обеспечением, в том числе специальные возможности операционных систем, таких как экранная клавиатура, альтернативные устройства ввода информации, а также обеспечение безбарьерной архитектурной среды, обеспечивающей доступность маломобильным группам обучающихся с ОВЗ.

В группах, в состав которых входят обучающиеся с ОВЗ, с целью реализации индивидуального подхода, а также принципа индивидуализации и дифференциации, преподавателю следует использовать технологию нелинейной конструкции учебных занятий, предусматривающую одновременное сочетание фронтальных, групповых и индивидуальных форм работы с различными категориями обучающихся, в т.ч. и имеющими ОВЗ.

В процессе учебных занятий в группах, в состав которых входят обучающиеся с ОВЗ, преподавателю желательно использовать технологии, направленные на решение дидактических, коммуникативных и компенсаторных задач, посредством использования информационно-коммуникативных технологий дистанционного и online обучения:

- стандартные технологии – например, компьютеры, имеющие встроенные функции настройки для лиц с ограниченными возможностями здоровья;

- доступные форматы данных, известные также как альтернативные форматы – например, доступный HTML, говорящие книги системы DAISY (Digital Accessible Information System – электронная доступная информационная система); а также «низко технологичные» форматы, такие, как система Брайля;

– вспомогательные технологии (ВТ) – это устройства, продукты, оборудование, программное обеспечение или услуги, направленные на усиление, поддержку или улучшение функциональных возможностей обучающихся с ОВЗ, к ним относятся аппараты, устройства для чтения с экрана, клавиатуры со специальными возможностями и т.д.;

– дистанционные образовательные технологии обучения студентов с ОВЗ предоставляют возможность индивидуализации траектории обучения данной категории обучающихся, что подразумевает индивидуализацию содержания, методов, темпа учебной деятельности обучающегося, возможность следить за конкретными действиями обучающегося с ОВЗ при решении конкретных задач, внесения, при необходимости, требуемых корректировок в деятельность обучающегося и преподавателя; данные технологии позволяют эффективно обеспечивать коммуникации обучающегося с ОВЗ не только с преподавателем, но и с другими обучающимися в процессе познавательной деятельности;

– наиболее эффективными формами и методами дистанционного обучения являются персональные сайты преподавателей, обеспечивающие онлайн поддержку профессионального образования обучающихся с ОВЗ, электронные УМК и РПД, учебники на электронных носителях, видеолекции и т.д.

В группах, в состав которых входят обучающиеся с ОВЗ, преподавателю желательно использовать в процессе учебных занятий технологии, направленные на активизацию учебной деятельности, такие как:

– система опережающих заданий, способствующих актуализации знаний и более эффективному восприятию обучающимися с ОВЗ данной учебной дисциплины;

– работа в диадах (парах) смешного состава, включающих обучающегося с ОВЗ и его однокурсников, не имеющих отклонений в психосоматическом развитии;

– опорные конспекты и схемы, позволяющие систематизировать и адаптировать изучаемый материал в соответствии с особенностями развития обучающихся с ОВЗ различной нозологии;

– бланковые методики, с использованием карточек, включающих индивидуальные многоуровневые задания, адаптированные с учетом особенностей развития и образовательных потребностей обучающихся с ОВЗ и их возможностей;

– методика ситуационного обучения (кейс-методы);

– методика совместного оставления проектов как способа достижения дидактической цели через детальную разработку актуальной проблемы, которая должна завершиться вполне реальным, осозаемым практическим результатом, оформленным тем или иным образом временной инициативной группой разработчиков из числа обучающихся с ОВЗ и их однокурсников, не имеющих отклонений в психосоматическом развитии;

– методики совместного обучения, реализуемые в составе временных инициативных групп, которые создаются в процессе учебных занятий из числа обучающихся с ОВЗ и их однокурсников, не имеющих отклонений в психосоматическом развитии, с целью совместного написания докладов, рефератов, эссе, а также подготовки библиографических обзоров научной и методической литературы, проведения экспериментальных исследований, подготовки презентаций, оформления картотеки нормативно-правовых документов, регламентирующих профессиональную деятельность и т.п.

В процессе учебных занятий в группах, в состав которых входят обучающиеся с ОВЗ, преподавателю желательно использовать технологии, направленные на позитивное стимулирование их учебной деятельности:

– предоставлять реальную возможность для получения в процессе занятий индивидуальной консультативно-методической помощи;

– давать возможность для выбора привлекательного задания, после выполнения обязательного, предупреждать возникновение неконструктивных конфликтов между обучающимися с ОВЗ и их однокурсниками, исключая, таким образом, возможность возникновения у участников образовательного процесса стрессовых ситуаций и негативных реакций.

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе учебных занятий преподавателю желательно использовать технологии, направленные на диагностику уровня и темпов профессионального становления обучающихся с ОВЗ, а также технологии мониторинга степени успешности формирования у них компетенций, предусмотренных ФГОС ВО при изучении данной учебной дисциплины, используя с этой целью специально адаптированные оценочные материалы и формы проведения промежуточной и итоговой аттестации, специальные технические средства, предоставляя обучающимся с ОВЗ дополнительное время для подготовки ответов, привлекая тьютеров.

По результатам текущего мониторинга степени успешности формирования у обучающихся с ОВЗ компетенций, предусмотренных ФГОС ВО в рамках изучения данной учебной дисциплины, при возникновении объективной необходимости, обусловленной оптимизацией темпов профессионального становления конкретного обучающегося с ОВЗ, преподавателю совместно с тьютером и службой психологической поддержки МТУСИ следует разработать адаптированный индивидуальный маршрут овладения данной учебной дисциплиной, адекватный его образовательным потребностям и возможностям.

10.Методические рекомендации обучающимся по освоению дисциплины

Приступая к изучению дисциплины, обучающимся необходимо внимательно ознакомиться с тематическим планом занятий, списком рекомендованной литературы. Следует уяснить последовательность выполнения индивидуальных учебных заданий. Самостоятельная работа обучающегося предполагает работу с научной и учебной литературой, умение создавать тексты. Уровень и глубина усвоения дисциплины зависят от активной и систематической работы на лекциях, изучения рекомендованной литературы, выполнения контрольных письменных заданий.

При изучении дисциплины обучающиеся выполняют следующие задания: изучают рекомендованную научно-практическую и учебную литературу; выполняют задания, предусмотренные для самостоятельной работы.

Основными видами аудиторной работы обучающихся являются практические занятия.

Практические занятия нацелены на изучение наиболее важных тем учебной дисциплины. Они служат для закрепления изученного материала, развития умений и навыков подготовки докладов, сообщений, приобретения опыта устных публичных выступлений, ведения дискуссии, аргументации и защиты выдвигаемых положений, а также для контроля преподавателем степени подготовленности обучающихся по изучаемой дисциплине.

Качество учебной работы обучающихся преподаватель оценивает с использованием фонда оценочных средств дисциплины по организации самостоятельной работы по дисциплине).

УТВЕРЖДАЮ:
Декан факультета _____

“ ____ ” 20__ г.

Лист актуализации рабочей программы практики

«_____»
наименование

Направление: (код, название направления/специальности)

Направленность (профиль): _____

Форма обучения: _____

(Возможны следующие варианты):

- а) Рабочая программа действует без изменений.
б) В рабочую программу вносятся следующие изменения:

- 1);
- 2);
- 3)

Разработчик (и): _____
(ФИО, ученая степень, ученое звание) «__» 20__ г.

Рабочая программа пересмотрена и одобрена на заседании кафедры _____
протокол № _____ от «__» 20__ г.

Заведующий кафедрой _____