

**Компонент
ОПОП**

09.03.01 Информатика и вычислительная техника

наименование ОПОП

направленность (профиль) «Технологии разработки программного обеспечения»

Б1.В.ДВ.03.01

шифр дисциплины

РАБОЧАЯ ПРОГРАММА

**Дисциплины
(модуля)**

Разработка web-приложений на языке Python

Разработчик (и):

Ляш О.И.

ФИО

зав.кафедрой

должность

канд.пед.наук,

доцент

ученая степень,

звание

Утверждено на заседании кафедры
информационных технологий

наименование кафедры

протокол № 6 от 24.02.2026

Заведующий кафедрой ИТ



подпись

Ляш О.И.

ФИО

**Мурманск
2026**

Пояснительная записка

Объем дисциплины 4 з.е.

1. Результаты обучения по дисциплине (модулю), соотнесенные с индикаторами достижения компетенций, установленными образовательной программой

Компетенции	Индикаторы достижения компетенций	Результаты обучения по дисциплине (модулю)
ПК-1 Способен разрабатывать технические спецификации на программные компоненты и их взаимодействие	ИД-1_{ПК-1} Анализирует существующую документацию (техническое задание, спецификации) и требования к программному обеспечению для выделения перечня подлежащих разработке программных компонентов. ИД-2_{ПК-1} Разрабатывает, согласовывает и утверждает технические спецификации на программные компоненты, описывающие их структуру, функции, логику работы и атрибуты качества. ИД-3_{ПК-1} Проектирует интерфейсы взаимодействия программных компонентов (API) и протоколы обмена данными между ними, фиксируя результаты в спецификации. ИД-4_{ПК-1} Оформляет техническую документацию на компоненты и их взаимодействие в соответствии с требованиями стандартов.	Знать: – Архитектурные принципы построения web-приложений: клиент-серверное взаимодействие, протокол HTTP/HTTPS, WSGI/ASGI. – Возможности и особенности web-фреймворков Python: Django/Flask/FastAPI. – Способы организации маршрутизации, middleware, шаблонизации, работы с формами и валидации данных. – Методы проектирования и интеграции реляционных баз данных через ORM. – Подходы к аутентификации, авторизации и обеспечению безопасности web-приложений. Уметь: – Выбирать и обосновывать подходящий фреймворк под задачи проекта. – Проектировать структуру web-приложения: маршруты, контроллеры, шаблоны, модели БД, формы. – Реализовывать REST API с автоматической документацией и обработкой ошибок. – Настраивать аутентификацию и разграничение доступа без использования готовых «стартовых» проектов. – Писать интеграционные и модульные тесты для API и бизнес-логики, отлаживать приложение. Владеть: – Навыками разработки полноценного web-приложения на Django/Flask/FastAPI. – Приёмами работы с формами, валидацией данных и выводом сообщений об ошибках. – Техниками отладки (логирование, точка останова, отладчик) и профилирования запросов. – Навыками контейнеризации web-приложения в Docker и развёртывания на сервере.
ПК-2 Способен проектировать компьютерное программное обеспечение	ИД-1_{ПК-2} Выбирает архитектуру программного обеспечения и декомпозирует его на модули и компоненты. ИД-2_{ПК-2} Применяет объектно-ориентированные или иные методологии для разработки моделей предметной области, структур данных и бизнес-логики будущего ПО. ИД-3_{ПК-2} Разрабатывает логическую и физическую схемы баз данных, а также осуществляет проектирование пользовательских интерфейсов с учетом требований эргономики. ИД-4_{ПК-2} Создает проектную документацию на программное обеспечение, позволяющую	Знать: – Архитектурные принципы построения web-приложений: клиент-серверное взаимодействие, протокол HTTP/HTTPS, WSGI/ASGI. – Возможности и особенности web-фреймворков Python: Django/Flask/FastAPI. – Способы организации маршрутизации, middleware, шаблонизации, работы с формами и валидации данных. – Методы проектирования и интеграции реляционных баз данных через ORM. – Подходы к аутентификации, авторизации и обеспечению безопасности web-приложений. Уметь: – Выбирать и обосновывать подходящий фреймворк под задачи проекта. – Проектировать структуру web-приложения: маршруты, контроллеры, шаблоны, модели БД, формы. – Реализовывать REST API с автоматической документацией и обработкой ошибок. – Настраивать аутентификацию и разграничение доступа без использования готовых «стартовых» проектов. – Писать интеграционные и модульные тесты для API и бизнес-логики, отлаживать приложение. Владеть: – Навыками разработки полноценного web-приложения на Django/Flask/FastAPI. – Приёмами работы с формами, валидацией данных и выводом сообщений об ошибках. – Техниками отладки (логирование, точка останова, отладчик) и профилирования запросов. – Навыками контейнеризации web-приложения в Docker и развёртывания на сервере.

	осуществить кодирование и дальнейшее развертывание продукта. ИД-5пк-2 Оценивает соответствие разработанного проекта исходным требованиям и техническому заданию, а также анализирует возможные риски реализации.	– Методами защиты от распространённых уязвимостей веб-приложений. декомпозиции на модули (сцены, контроллеры, сетевые менеджеры).
--	---	--

2. Содержание дисциплины (модуля)

Тема 1. Основы разработки web-приложений на Python. Введение в веб-фреймворки Python: Django, Flask, FastAPI. Установка и настройка окружения. Принципы работы HTTP: запросы, ответы, методы, коды состояния. WSGI и ASGI: различия и применение. Создание первого веб-приложения на Flask: маршрутизация, обработка GET/POST-запросов, возврат HTML и JSON. Шаблонизатор Jinja2: переменные, фильтры, циклы, условные операторы, наследование шаблонов. Работа со статическими файлами (CSS, JS, изображения). Формы и валидация данных (WTForms). Обработка ошибок и логирование. Основы ORM на примере SQLAlchemy: модели, сессии, простые запросы. Подключение к базе данных SQLite/PostgreSQL. Миграции (Alembic). Базовая аутентификация: сессии и куки.

Тема 2. Разработка полнофункционального web-приложения и API. Архитектура Django: модели, представления (FBV/CBV), шаблоны, формы, админка. Создание моделей данных, связей (ForeignKey, ManyToMany). Миграции в Django. ORM Django: фильтрация, агрегации, аннотации. Встроенная система аутентификации и авторизации. Расширение модели пользователя. Проектирование REST API с использованием Django REST Framework (DRF) или FastAPI. Серялизаторы, viewsets, роутеры. Автодокументация API (OpenAPI/Swagger). Пагинация, фильтрация, сортировка данных. Безопасность API: JWT-токены, OAuth2 (на примере FastAPI или DRF). Работа с формами и валидация на стороне сервера. Middleware: назначение, написание собственного middleware. Кэширование (кэш на уровне представлений, Redis). Паттерны проектирования в веб-приложениях: MVC, репозиторий, сервисный слой.

Тема 3. Развертывание, оптимизация и сопровождение web-приложений на Python. Подготовка приложения к production: настройка переменных окружения (python-dotenv, django-environ). Сбор статистики и медиафайлов. Использование Gunicorn/uWSGI для WSGI-приложений и Uvicorn/Hypercorn для ASGI. Контейнеризация с Docker: написание Dockerfile, docker-compose для связки приложение + БД + кэш. CI/CD пайплайны (GitHub Actions, GitLab CI). Развертывание на облачных платформах (Heroku, Render, Yandex Cloud). Базовая безопасность: защита от CSRF, XSS, SQL-инъекций, подделка запросов, HTTPS (Let's Encrypt). Мониторинг логирования (Sentry, ELK). Нагрузочное тестирование (Locust, k6) и профилирование (django-debug-toolbar, py-spy). Оптимизация запросов к БД (n+1 проблема, индексы, explain). Асинхронные задачи (Celery, Dramatiq, FastAPI BackgroundTasks). Веб-сокеты для real-time функционала (Django Channels, FastAPI WebSocket). Финальная сборка, отладка, документирование проекта. Демонстрация готового web-приложения на Python.

3. Перечень учебно-методического обеспечения дисциплины (модуля)

- мультимедийные презентационные материалы по дисциплине (модулю) представлены в электронном курсе в ЭИОС МАУ;
- методические указания к выполнению лабораторных/практических/контрольных работ (выбрать) представлены в электронном курсе в ЭИОС МАУ;

– методические материалы для обучающихся по освоению дисциплины (модуля) представлены на официальном сайте МАУ в разделе «Информация по образовательным программам, в том числе адаптированным».

4. Фонд оценочных средств по дисциплине (модулю)

Является отдельным компонентом образовательной программы, разработан в форме отдельного документа, включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины (модуля);
- задания текущего контроля;
- задания промежуточной аттестации;
- задания внутренней оценки качества образования.

5. Перечень основной и дополнительной учебной литературы (печатные издания, электронные учебные издания и (или) ресурсы электронно-библиотечных систем)

Основная литература:

1. Меле, А. Django 5 в примерах : руководство / А. Меле ; перевод с английского А. В. Логунова. — Москва : ДМК Пресс, 2025. — 866 с. — ISBN 978-5-93700-259-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/464264>.

2. Гринберг, М. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг. — Москва : ДМК Пресс, 2014. — 272 с. — ISBN 978-5-97060-138-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/90103>.

3. Елисеев, А. И. Разработка программных интерфейсов веб-приложений с использованием фреймворка FastAPI : учебное пособие / А. И. Елисеев, Ю. В. Минин. — Тамбов : ТГТУ, 2024. — 81 с. — ISBN 978-5-8265-2821-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/472310>.

Дополнительная литература:

4. Елисеев, А. И. Разработка веб-приложений с использованием фреймворка Flask : учебное пособие : в 2 частях / А. И. Елисеев, Ю. В. Минин, В. А. Гриднев. — Тамбов : ТГТУ, 2021 — Часть 2 — 2021. — 84 с. — ISBN 978-5-8265-2438-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/320435>.

5. Елисеев, А. И. Разработка веб-приложений с использованием фреймворка Flask : учебное пособие : в 2 частях / А. И. Елисеев, Ю. В. Минин, В. А. Гриднев. — Тамбов : ТГТУ, 2020 — Часть 1 — 2020. — 82 с. — ISBN 978-5-8265-2188-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/320318>.

6. Майтак, Р. В. Python, Django, Data Science : учебное пособие / Р. В. Майтак, П. А. Пылов, А. В. Протодяконов. — Вологда : Инфра-Инженерия, 2025. — 516 с. — ISBN 978-5-9729-2143-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/499760>.

7. Меле, А. Django 4 в примерах : руководство / А. Меле ; перевод с английского А. В. Логунова. — Москва : ДМК Пресс, 2023. — 800 с. — ISBN 978-5-93700-204-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/348113>.

6. Профессиональные базы данных и информационные справочные системы

1) Официальный сайт языка программирования Python – URL:

<https://www.python.org/>

2) Официальная документация фреймворка Django – URL:
<https://docs.djangoproject.com>

3) Официальная документация фреймворка Flask – URL:
<https://flask.palletsprojects.com/en/stable/>

4) Официальная документация фреймворка FastApi – URL:
<https://fastapi.tiangolo.com/>

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства

- 1) Интегрированная среда разработки: OpenIDE
- 2) Редактор кода: Geany / VS Codium / VS Code
- 3) Фреймворк: Django, Flask, FastAPI
- 4) Дистрибутив операционной системы Linux: Debian / ALT Linux / Astra Linux / RedOS
- 5) Инструменты управления виртуальными машинами: Virt-manager / VirtualBox
- 6) Офисный пакет: LibreOffice / Р7-Офис / МойОфис
- 7) Браузер: Chromium / Firefox / Яндекс Браузер

8. Обеспечение освоения дисциплины лиц с инвалидностью и ОВЗ

Обучающиеся из числа инвалидов и лиц с ОВЗ обеспечиваются печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

9. Материально-техническое обеспечение дисциплины (модуля) представлено в приложении к ОПОП «Материально-технические условия реализации образовательной программы» и включает:

- учебные аудитории для проведения учебных занятий, предусмотренных программой бакалавриата, оснащенные оборудованием и техническими средствами обучения;

- помещения для самостоятельной работы обучающихся, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к электронной информационно-образовательной среде МАУ;

Допускается замена оборудования его виртуальными аналогами.

10. Распределение трудоемкости по видам учебной деятельности

Таблица 1 - Распределение трудоемкости

Вид учебной деятельности	Распределение трудоемкости дисциплины (модуля) по формам обучения			
	Очная		Заочная	
	Семестр	Всего часов	Семестр	Всего часов
	5		4	
Лекции	20	20	4	4
Практические занятия				
Лабораторные работы	40	40	8	8
Самостоятельная работа	84	84	128	128
Подготовка к промежуточной аттестации				
Всего часов по дисциплине	144	144	144	144
/ из них в форме практической подготовки				
Формы промежуточной аттестации и текущего контроля				
Экзамен	-	-	-	-
Зачёт с оценкой	1	1	1	1
Курсовая работа (проект)	-	-	-	-
Количество расчетно-графических работ	1	1	1	1
Количество контрольных работ	-	-	-	-
Количество рефератов	-	-	-	-
Количество эссе	-	-	-	-

Перечень лабораторных работ по формам обучения

№ п/п	Темы лабораторных работ
1	2
	Очная форма
	Модуль 1. Основы разработки web-приложений на Python
1.	Установка и настройка окружения. Создание первого веб-приложения на Flask.
2.	Маршрутизация, шаблонизация и статические файлы.
3.	Формы и валидация данных. Подключение базы данных через SQLAlchemy.
4.	Миграции и аутентификация на сессиях.
	Модуль 2. Разработка полнофункционального web-приложения и API
5.	Создание проекта на Django. Модели, панель администрирования и ORM.
6.	REST API на Django REST Framework (DRF) или FastAPI.
7.	Аутентификация JWT и права доступа.
	Модуль 3. Развертывание, оптимизация и сопровождение web-приложений на Python
8.	Контейнеризация приложения с Docker и docker-compose.
9.	Настройка production-среды: Gunicorn, Nginx, переменные окружения.
10.	CI/CD пайплайн и развёртывание на облачной платформе.
11.	Безопасность, оптимизация запросов и асинхронные задачи.
	Заочная форма
	Модуль 1. Основы разработки web-приложений на Python
1.	Установка и настройка окружения. Создание первого веб-приложения на Flask.
2.	Маршрутизация, шаблонизация и статические файлы.
	Модуль 2. Разработка полнофункционального web-приложения и API
3.	Создание проекта на Django. Модели, панель администрирования и ORM.
4.	REST API на Django REST Framework (DRF) или FastAPI.
	Модуль 3. Развертывание, оптимизация и сопровождение web-приложений на Python
5.	Контейнеризация приложения с Docker и docker-compose.

