

Министерство науки и высшего образования Российской Федерации  
**Муромский институт (филиал)**  
федерального государственного бюджетного образовательного учреждения высшего образования  
**«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»**  
(МИ ВлГУ)

Кафедра *ПИИ*

«УТВЕРЖДАЮ»  
Заместитель директора по УР  
\_\_\_\_\_ Д.Е. Андрианов  
\_\_\_\_\_ 20.05.2025

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

*Web-разработка на Python*

**Направление подготовки**

*09.03.04 Программная инженерия*

**Профиль подготовки**

*Методы и средства разработки  
программного обеспечения*

Семестр	Трудоем- кость, час./зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консультация, час.	Конт- роль, час.	Всего (контакт- ная работа), час.	СРС, час.	Форма промежу- точного контроля (экз., зач., зач. с оц.)
5	144 / 4	32		32	3,2	0,25	67,45	76,55	Зач.
6	180 / 5	16	10	16	3,6	2,35	47,95	96,4	Экз.(35,65)
Итого	324 / 9	48	10	48	6,8	2,6	115,4	172,95	35,65

Муром, 2025 г.

## 1. Цель освоения дисциплины

Цель изучения дисциплины - освоение методов и технологий, связанных с созданием распределенных информационных систем на основе современных технологий, подходов и методологий, а также изучение круга специальных вопросов обеспечения эффективного использования средств обработки информации.

Задачами дисциплины в соответствии с указанной целью являются:

- формирование у студентов представления о современных методах проектирования и разработки распределенных информационных систем;
- формирование опыта разработки системы по технологии ASP.NET Core MVC;
- изучение методов развертывания готовых систем на внешних серверах.

## 2. Место дисциплины в структуре ОПОП ВО

Дисциплина базируется на дисциплинах "Организация баз данных" и "Объектно-ориентированное программирование".

## 3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.5 Применяет методы и средства разработки модулей и компонент программного обеспечения для решения задач автоматизации	Уметь применять методы и средства разработки модулей и компонент программного обеспечения для решения задач автоматизации (ОПК-6.5)	тест
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Выполняет разработку программных модулей web-приложений	Владеть навыками разработки программных модулей web-приложений (ОПК-7.1)	тест
ПК-7 Способность оценивать временную и емкостную сложность программного обеспечения	ПК-7.1 Разрабатывает и подбирает алгоритмы и структуры данных для решения вычислительных задач и задач автоматизации	Знать алгоритмы и структуры данных для решения вычислительных задач и задач автоматизации (ПК-7.1)	тест

#### 4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 9 зачетных единиц, 324 часа.

##### 4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

##### 4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Проектирование и разработка web-приложений	5	32		32					76,55	тестирование
Всего за семестр		144	32		32			3,2	0,25	76,55	Зач.
2	Технология разработки web-приложений на Django	6	16	10	16					96,4	тестирование
Всего за семестр		180	16	10	16		+	3,6	2,35	96,4	Экз.(35,65)
Итого		324	48	10	48			6,8	2,6	172,95	35,65

##### 4.1.2. Содержание дисциплины

##### 4.1.2.1. Перечень лекций

###### Семестр 5

*Раздел 1. Проектирование и разработка web-приложений*

###### Лекция 1.

Понятия распределенных вычислений и распределенной системы (2 часа).

###### Лекция 2.

Логические программные слои распределенных систем (2 часа).

###### Лекция 3.

Способы взаимодействия в распределенных системах (2 часа).

###### Лекция 4.

Распределенные БД. Коллизии (2 часа).

###### Лекция 5.

Основные механизмы в распределенных системах (2 часа).

**Лекция 6.**

Распределенные задачи и алгоритмы (2 часа).

**Лекция 7.**

Надежность и безопасность PCОД (2 часа).

**Лекция 8.**

Введение в веб-разработку на Python (2 часа).

**Лекция 9.**

Модели и миграции в Django (2 часа).

**Лекция 10.**

Базы данных в веб-разработке (2 часа).

**Лекция 11.**

Административная панель Django (2 часа).

**Лекция 12.**

Шаблоны и статические файлы в Django (2 часа).

**Лекция 13.**

Формы в Django (2 часа).

**Лекция 14.**

Аутентификация и авторизация в Django (2 часа).

**Лекция 15.**

Django REST Framework (DRF) (2 часа).

**Лекция 16.**

Оптимизация веб-приложений (2 часа).

**Семестр 6**

*Раздел 2. Технология разработки web-приложений на Django*

**Лекция 17.**

Работа с файлами и медиа в Django (2 часа).

**Лекция 18.**

Развертывание веб-приложений (2 часа).

**Лекция 19.**

Docker и контейнеризация веб-приложений (2 часа).

**Лекция 20.**

CI/CD для веб-приложений (2 часа).

**Лекция 21.**

Безопасность веб-приложений. Основные уязвимости (2 часа).

**Лекция 22.**

RESTful API - основы REST архитектуры (2 часа).

**Лекция 23.**

Тестирование веб-приложений на Python (2 часа).

**Лекция 24.**

Современные тренды в веб-разработке (2 часа).

#### **4.1.2.2. Перечень практических занятий**

**Семестр 6**

*Раздел 2. Технология разработки web-приложений на Django*

**Практическое занятие 1**

Анализ и исследование предметной области для автоматизации (2 часа).

**Практическое занятие 2**

Проектирование web-приложения (2 часа).

**Практическое занятие 3**

Разработка алгоритмов обработки данных (2 часа).

**Практическое занятие 4**

Реализация web-приложения (2 часа).

**Практическое занятие 5**

Тестирование и узкие места web-приложений (2 часа).

### 4.1.2.3. Перечень лабораторных работ

#### Семестр 5

##### *Раздел 1. Проектирование и разработка web-приложений*

##### **Лабораторная 1.**

Знакомство с веб-фреймворком Django. Создание блога (4 часа).

##### **Лабораторная 2.**

Усовершенствование блога и специальные возможности (4 часа).

##### **Лабораторная 3.**

Тегирование. Создание карт веб-сайтов (4 часа).

##### **Лабораторная 4.**

Настройка механизма аутентификации (4 часа).

##### **Лабораторная 5.**

Разработка портала обмена фото материалами (4 часа).

##### **Лабораторная 6.**

Распространение контента на веб-сайте (4 часа).

##### **Лабораторная 7.**

Отслеживание действий пользователя (4 часа).

##### **Лабораторная 8.**

Разработка интернет-магазина (4 часа).

#### Семестр 6

##### *Раздел 2. Технология разработки web-приложений на Django*

##### **Лабораторная 9.**

Управление платежами и заказами (4 часа).

##### **Лабораторная 10.**

Разработка бонусной системы оплаты (4 часа).

##### **Лабораторная 11.**

Интернационализация и локализация (4 часа).

##### **Лабораторная 12.**

Разработка системы управления контентом (4 часа).

### 4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Установка и настройка Python, виртуального окружения и PIP.
2. Основы работы с фреймворком Flask: простое веб-приложение.
3. Основы работы с Django: создание первого проекта.
4. Сравнение Flask и Django: плюсы и минусы.
5. Создание RESTful API на Flask (Flask-RESTful).
6. Разработка API с использованием Django REST Framework.
7. Аутентификация и авторизация: JWT, OAuth2.
8. Работа с базами данных (SQLite, PostgreSQL) в Django.
9. Асинхронная разработка с FastAPI.
10. WebSockets: чат-приложение на Flask-SocketIO или Django Channels.
11. Шаблонизация: Jinja2 в Flask и Django Templates.
12. Интеграция фронтенда (HTML/CSS/JS) с бэкендом на Python.
13. Использование React/Vue.js с Django/Flask.
14. Сборка фронтенда с Webpack в Django-проекте.
15. Работа с SQLAlchemy в Flask.
16. Миграции баз данных в Django (Django Migrations).
17. Оптимизация запросов в Django ORM.
18. Работа с NoSQL (MongoDB) в Python (PyMongo, Djongo).
19. Деплой Flask-приложения на Heroku/Vercel.
20. Развертывание Django-проекта на сервере (Nginx + Gunicorn).

21. Docker и Docker Compose для веб-приложений на Python.
22. CI/CD для Python-проектов (GitHub Actions, GitLab CI).
23. Защита от SQL-инъекций, XSS, CSRF.
24. Кэширование в Django (Redis, Memcached).
25. Оптимизация скорости загрузки веб-страниц.
26. Парсинг веб-сайтов (BeautifulSoup, Scrapy).
27. Интеграция Telegram-бота с веб-приложением.
28. Работа с GraphQL в Django (Graphene).
29. Тестирование веб-приложений (PyTest, Django Test).
30. Разработка микросервисной архитектуры на Python.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

#### **4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР**

Не планируется.

#### **4.1.2.6. Примерный перечень тем курсовых работ (проектов)**

1. Распределенная ИС библиотеки.
2. Распределенная ИС «Кулинарный справочник».
3. Распределенная ИС Страхового агентства.
4. Распределенная ИС Службы такси.
5. Распределенная ИС Ветеринарной клиники.
6. Распределенная ИС Ломбарда.
7. Распределенная ИС Общежития.
8. Распределенная ИС администратора гостиницы.
9. Распределенная ИС домоуправления.

### **5. Образовательные технологии**

В рамках изучения дисциплины используются интерактивные технологии преподавания, выраженные в виде совместных обсуждений проблемных ситуаций, совместного анализа путей решения поставленных задач. В рамках выполнения лабораторных и практических работ формируются небольшие коллективы из студентов для совместного решения задач. Результаты работы отдельных коллективов обсуждаются всей группой, при этом используются средства мультимедийной техники.

Преподаватель выступает в роли координатора работы коллективов студентов, дает оценку их работе.

### **6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

Фонды оценочных материалов (средств) приведены в приложении.

### **7. Учебно-методическое и информационное обеспечение дисциплины.**

#### **7.1. Основная учебно-методическая литература по дисциплине**

1. Петрухнова, Г. В. Введение в распределенные системы : учебное пособие / Г. В. Петрухнова. — Воронеж : Воронежский государственный технический университет, ЭБС АСВ, 2021. — 81 с. — ISBN 978-5-7731-0925-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/111462.html> - <https://www.iprbookshop.ru/111462.html>
2. Ванина, М. Ф. Распределенные информационные системы. Технологии реализации распределенных информационных систем : учебное пособие / М. Ф. Ванина, А. Г. Ерохин. — Москва : Московский технический университет связи и информатики, 2020. — 132 с. — Текст

: электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/97362.html> - <https://www.iprbookshop.ru/97362.html>

## **7.2. Дополнительная учебно-методическая литература по дисциплине**

1. Назаркин, О. А. Современные технологии разработки распределенных вычислительных систем : учебное пособие / О. А. Назаркин, В. А. Алексеев. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2017. — 66 с. — ISBN 978-5-88247-840-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/83172.html> - <https://www.iprbookshop.ru/83172.html>
2. Столбовский, Д. Н. Разработка Web-приложений ASP.NET с использованием Visual Studio .NET : учебное пособие / Д. Н. Столбовский. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 375 с. — ISBN 978-5-4497-0370-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/89469.html> - <https://www.iprbookshop.ru/89469.html>

## **7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института ([www.mivlgu.ru/iop](http://www.mivlgu.ru/iop)), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

- электронная библиотечная системы "IPRBooks" (<http://www.iprbookshop.ru/>);

Программное обеспечение:

Google Chrome (Лицензионное соглашение Google)

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

Notepad++ (GNU GPL 3)

## **7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

[iprbookshop.ru](http://iprbookshop.ru)

[mivlgu.ru/iop](http://mivlgu.ru/iop)

## **8. Материально-техническое обеспечение дисциплины**

Лаборатория информационно – коммуникационных систем

Сервер «Ай Тек» на базе 2 процессоров Intel Xeon; 12 шт. компьютеров Intel Core i5-10400 2,90 GHz/ 8 Gb DDR-4/ SSD-480 Gb/ Hiper 21,5'; интерактивная доска SMART Board 480 со встроенным проектором V25; маршрутизатор Gigabit Switch TEG-S16S. Маркерная доска. Доступ к сети Интернет.

## **9. Методические указания по освоению дисциплины**

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя,

каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

На практических занятиях пройденный теоретический материал подкрепляется решением задач по основным темам дисциплины. Занятия проводятся в компьютерном классе, используя специальное программное обеспечение.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводится в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Курсовая работа выполняется в соответствии с методическими указаниями на курсовую работу. Обучающийся выбирает одну из указанных в перечне тем курсовых работ, исходя из своих интересов, наличия соответствующих литературных и иных источников. В ходе выполнения курсовой работы преподаватель проводит консультации обучающегося. На заключительном этапе обучающийся оформляет пояснительную записку к курсовой работе и выполняет ее защиту в присутствии комиссии из преподавателей кафедры.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.



Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.04 Программная инженерия* и профилю подготовки *Методы и средства разработки программного обеспечения*

Рабочую программу составил *к.т.н., доцент кафедры ПИН Белякова А.С.*\_\_\_\_\_

Программа рассмотрена и одобрена на заседании кафедры *ПИН*

протокол № 27 от 13.05.2025 года.

Заведующий кафедрой *ПИН* \_\_\_\_\_ *Жизняков А.Л.*  
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 15.05.2025 года.

Председатель комиссии ФИТР \_\_\_\_\_ *Кутарова Е.И.*  
(Подпись) (Ф.И.О.)

**Фонд оценочных материалов (средств) по дисциплине  
Web-разработка на Python**

**1. Оценочные материалы для проведения текущего контроля успеваемости  
по дисциплине**

1. В распределенной системе должны быть приняты меры, чтобы часть аппаратуры брала на себя выполнение работы в случае выхода из строя другой части системы. Это требование обеспечивает свойство...

- A. Открытость
- B. Прозрачность
- C. Масштабируемость
- D. Многослойность

2. Гибкость распределенной системы - это

- A. легкость конфигурирования системы, состоящей из различных компонентов
- B. способность к взаимодействию с другими системами
- C. скрывание разницы в способах представления данных и в способах доступа к ресурсам

3. Достоинство одноярусных архитектур

- A. Повышенная производительность
- B. Легкая масштабируемость
- C. Открытость и переносимость

4. Если клиент после отправки запроса выполняет другие действия – это

- A. асинхронное взаимодействие
- B. синхронное взаимодействие
- C. не бывает такого

5. Если работа клиента на время обработки запроса сервером приостанавливается - это

- A. синхронное взаимодействие
- B. асинхронное взаимодействие
- C. нет правильного ответа
- D. это нерабочее взаимодействие

6. Масштабируемость распределенной системы проявляется по отношению к

- A. размеру компонент
- B. географическому расположению
- C. административному устройству
- D. размеру и количеству баз данных

7. Одноярусные архитектуры распределенных систем характеризуются

- A. монолитным исполнением слоев системы
- B. одной базой данных
- C. однопользовательским режимом

8. Основные задачи распределенных систем программного обеспечения

- A. облегчить их пользователям доступ к удаленным ресурсам
- B. контролировать совместное использование удаленных ресурсов
- C. обеспечивать многофакторную авторизацию пользователей
- D. поддерживать возможность использования множества независимых баз данных

9. Открытость РСОД обеспечивается

- A. Использованием стандартных СУБД

- В. Использованием типовых протоколов
- С. Использованием типовых схем интерфейсов
- D. Нет правильного ответа

10. Открытые системы обладают очень важной характеристикой –

- A. гибкостью
- В. независимостью
- С. переносимостью
- D. универсальностью

11. Перечислите слои распределенной системы

- A. презентационный слой
- В. слой прикладной логики
- С. слой управления ресурсами
- D. слой баз данных

12. В состав трехъярусных архитектур входят

- A. клиент
- В. Сервер баз данных (или других ресурсов)
- С. сервер приложений (реализация бизнес-логики)
- D. сервер доступа к глобальной сети

13. Чтобы поддерживать представление различных компьютеров и вычислительных сетей в виде единой системы, организация распределенных систем часто включает в себя дополнительный уровень программного обеспечения, который называется

- A. Уровень системной поддержки (middleware)
- В. Клиентский уровень (frontend)
- С. Уровень логики сервера (backend)

14. В двухъярусных архитектурах клиенты бывают монолитными и многослойными

- Верно
- Неверно

15. Двухъярусные архитектуры - это архитектуры клиент/сервер

- Верно
- Неверно

16. В двухъярусных архитектурах клиенты бывают толстыми и тонкими

- Верно
- Неверно

17. Требование прозрачности РСОД о том, что распределенная система должна скрывать разницу в способах представления данных и в способах доступа к ресурсам

- Верно
- Неверно

18. Распределенная система – это набор независимых компьютеров, представляющихся их пользователям единой объединенной системой

- Верно
- Неверно

## Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Выполнение практических заданий	20
Рейтинг-контроль 2	Выполнение практических заданий	20
Рейтинг-контроль 3	Выполнение практических заданий	20
Посещение занятий студентом		0
Дополнительные баллы (бонусы)		0
Выполнение семестрового плана самостоятельной работы		0

### 2. Промежуточная аттестация по дисциплине

#### Перечень вопросов к экзамену / зачету / зачету с оценкой.

#### Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

1. Какой командой можно создать новый проект Django?

`django-admin startapp`

`django-admin startproject`

`python manage.py newproject`

`python django createproject`

2. Какой файл в Django отвечает за маршрутизацию URL?

`models.py`

`urls.py`

`views.py`

`settings.py`

3. Какой ORM-метод Django используется для получения всех объектов модели?

`Model.get()`

`Model.fetch()`

`Model.objects.all()`

`Model.select_all()`

4. Какой шаблонизатор используется в Django по умолчанию?

Django Template Language (DTL)

Jinja2

Mako

Twig

5. Какой middleware отвечает за защиту от CSRF-атак в Django?

`SessionMiddleware`

`CsrfViewMiddleware`

`AuthenticationMiddleware`

`SecurityMiddleware`

### Методические материалы, характеризующие процедуры оценивания

В качестве тестовых вопросов при проведении промежуточных аттестаций используются вопросы, приведенные в пункте 6.3. Из каждого раздела, освоенного студентом, выбирается по одному теоретическому и одному практическому вопросу. Теоретические вопросы раскрываются в устной, либо в письменной форме. Практические задания как правило реализуются с помощью персонального компьютера.

При проверке знаний, приобретенных в рамках выполнения лабораторных работ, используются контрольные вопросы, приведенные в методических указаниях к лабораторным работам. Защита лабораторных также является средством промежуточной аттестации.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	<i>Уровень сформированности компетенций</i>
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	<b><i>Высокий уровень</i></b>
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<b><i>Продвинутый уровень</i></b>

50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<b><i>Пороговый уровень</i></b>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<b><i>Компетенции не сформированы</i></b>

### 3. Задания в тестовой форме по дисциплине

Примеры заданий:

1. Какой командой запускается встроенный сервер Django для разработки?

python manage.py runserver --start

python manage.py runserver

django-admin runserver

python django startserver

2. Какой класс Django используется для работы с формами?

FormView

forms.Form или forms.ModelForm

TemplateForm

HtmlForm

3. Как в Django добавить статические файлы (CSS, JS)?

Через `<script src="/static/">` без дополнительных настроек

Через тег `{% load static %}` и `{% static 'path/to/file.css' %}`

Через прямое указание пути в settings.py без STATIC\_URL

Только через внешние CDN

4. Какой метод HTTP используется по умолчанию для обработки данных формы в Django?

GET

POST

PUT

DELETE

5. Где в Django определяется структура базы данных?

В файле admin.py

В файле models.py

В файле views.py

В файле urls.py

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=3017>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.