МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 41

УТВЕРЖДАЮ
Руководитель образовательной программы

доц.,к.т.н.,доц.
(должность, уч. степень, звание)

О.Я. Солёная
(инициалы, фамилия)

(подпись)

17 апреля 2026 г.

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

## «ИТ-модуль "Программирование на языке Python"»
(Наименование дисциплины)

| | |
|---|---|
| Код направления подготовки/ специальности | 13.03.02 |
| Наименование направления подготовки/ специальности | Электроэнергетика и электротехника |
| Наименование направленности/ специализации | Энергетические электрические машины |
| Форма обучения | очная |
| Год приема | 2024 |

Санкт-Петербург– 2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

доцент, к.т.н. _____ С.А. Чернышев
(должность, уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Программа одобрена на заседании кафедры № 41
«16» февраля 2026 г, протокол №07-2025/26

Заведующий кафедрой № 41

д.т.н.,проф. _____ Г.А. Коржавин
(уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Заместитель директора института №3 по методической работе

доц.,к.т.н. _____ Н.В. Решетникова
(должность, уч. степень, звание) (подпись, дата) (инициалы, фамилия)

Аннотация

Дисциплина «ИТ-модуль "Программирование на языке Python"» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 13.03.02 «Электроэнергетика и электротехника» направленности/специализации «Энергетические электрические машины». Дисциплина реализуется кафедрой «№41».

Дисциплина не является обязательной при освоении обучающимся образовательной программы и направлена на углубленное формирование следующих компетенций:

УК-6 «Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни»

Содержание дисциплины охватывает круг вопросов, связанных с с нюансами языка программирования Python, изучением его базовых синтаксических конструкций, разработкой и тестированием программного обеспечения в процедурном и объектно-ориентированном стиле программирования, работе с файлами и базами данных, разработкой телеграм-ботов, реализацией многопроцессорных и асинхронных программ на Python.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции в формате онлайн, практические занятия, самостоятельная работа обучающегося, консультации с применением дистанционных образовательных технологий, производственную практику.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме зачета (5 семестр), дифференцированного зачета (6 семестр).

Общая трудоемкость освоения дисциплины составляет 7 зачетных единиц, 252 часа.

Язык обучения по дисциплине «русский »

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

- формирование у слушателей профессиональной компетенции, необходимой для работы в среде программирования и реализации программного кода на языке программирования Python..
- формирование у слушателей профессиональных компетенций, необходимых для работы с файлами, базами данных средствами ORM SQLAlchemy и написанию телеграм-ботов.

1.2. Дисциплина является факультативной дисциплиной по направлению образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

| Категория (группа) компетенции | Код и наименование компетенции | Код и наименование индикатора достижения компетенции |
|---|---|---|
| Универсальные компетенции | УК-6 Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни | УК-6.З.1 знать основные приемы эффективного управления собственным временем; основные методики самоконтроля, саморазвития и самообразования<br>УК-6.З.2 знать образовательные Интернет-ресурсы, возможности и ограничения образовательного процесса при использовании цифровых технологий<br>УК-6.У.1 уметь управлять своим временем; ставить себе образовательные цели под возникающие жизненные задачи<br>УК-6.У.2 уметь использовать цифровые инструменты в целях самообразования<br>УК-6.В.1 владеть навыками саморазвития и самообразования<br>УК-6.В.2 владеть навыками использования цифровых инструментов для саморазвития и самообразования |

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Информатика»,
- «Алгоритмизация и программирование»,

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

| Вид учебной работы | Всего | Трудоемкость по семестрам | |
|---|---|---|---|
| | | №5 | №6 |

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| ***Общая трудоемкость дисциплины***, ЗЕ/ (час) | 7/ 252 | 3/ 108 | 4/ 144 |
| ***Из них часов практической подготовки*** | | | |
| ***Аудиторные занятия***, всего час. | 119 | 51 | 68 |
| в том числе: | | | |
| лекции (Л), (час) | 34 | 17 | 17 |
| практические/семинарские занятия (ПЗ), (час) | 85 | 34 | 51 |
| лабораторные работы (ЛР), (час) | | | |
| курсовой проект (работа) (КП, КР), (час) | | | |
| экзамен, (час) | | | |
| ***Самостоятельная работа***, всего (час) | 133 | 57 | 76 |
| ***Вид промежуточной аттестации*:** зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.) | Зачет, Дифф. зач., | Зачет, | Дифф. зач., |

## 4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий. Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

| Разделы, темы дисциплины | Лекции (час) | ПЗ (СЗ) (час) | ЛР (час) | КП (час) | СРС (час) |
|---|---|---|---|---|---|
| Семестр 5 | | | | | |
| Раздел 1. Краткая история Python, его особенности, типы данных и управляющие конструкции<br>Тема 1.1. Краткая история Python и его ключевые особенности<br>Тема 1.2. Установка и основные типы данных Python.<br>Тема 1.3. Операторы и управляющие конструкции Python. | 4 | 10 | | | 13 |
| Раздел 2. Функции, модули, пакеты и их тестирование<br>Тема 2.1. Функция с точки зрения уровней абстракции и объявление функций в Python<br>Тема 2.2. Области видимости переменных, замыкание и аннотация функций<br>Тема 2.3. Лямбда-функции, декораторы и генераторы<br>Тема 2.4. Создание модулей, пакетов и принцип работы импортирования<br>Тема 2.5. Тестирование функций и работа со статическими анализаторами кода | 5 | 10 | | | 18 |

| | | | | | |
|---|---|---|---|---|---|
| Раздел 3. Объектно-ориентированное программирование<br>Тема 3.1. Класс, как очередной уровень абстракции и основные принципы ООП<br>Тема 3.2. Объявление класса в Python. Типы переменных и методов<br>Тема 3.3. Наследование, абстрактные классы и переопределение методов<br>Тема 3.4. Перечисления и классы данных<br>Тема 3.5. «Магические» (дандер) методы классов и Generics (Обобщения)<br>Тема 3.6. Исключения и тестирование классов | 8 | 14 | | | 26 |
| Итого в семестре: | 17 | 34 | | | 57 |
| Семестр 6 | | | | | |
| Раздел 4. Работа с файлами<br>Тема 4.1. Работа с файлами и директориями. Основные принципы<br>Тема 4.2. Пример работы с файлами<br>Тема 4.3. Пример работы с excel-файлами и визуализация данных | 5 | 10 | | | 10 |
| Раздел 5. Работа с СУБД SQLite<br>Тема 5.1. Основные команды SQLite<br>Тема 5.2. Пример работы с СУБД SQLite средствами Python<br>Тема 5.3. Работа с СУБД SQLite средствами ORM SQLAlchemy<br>Тема 5.4. Транзакции | 4 | 10 | | | 15 |
| Раздел 6. Потоки, процессы и асинхронное программирование в Python<br>Тема 6.1. Основные определения<br>Тема 6.2. Потоки и процессы в Python<br>Тема 6.3. Асинхронное программирование в Python | 4 | 10 | | | 15 |
| Тема 7. Разработка телеграм-ботов<br>Тема 7.1. Настройка рабочего окружения, создание моделей и инициализация БД<br>Тема 7.2 Разработка основного функционала телеграм-бота<br>Тема 7.3 Генерация отчетов | 4 | 10 | | | 15 |
| Итоговый проект | - | 11 | | | 21 |
| Итого в семестре: | 17 | 51 | | | 76 |
| Итого | 34 | 85 | 0 | 0 | 133 |
| | | | | | |

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| **1.** | **Краткая история Python, его особенности, типы данных и управляющие конструкции**<br>*Тема 1.1. Краткая история Python и его ключевые особенности*<br>История создания и развития Python. В Python все объект. Интернированные объекты в Python. Глобальная блокировка интерпретатора и подходы к сборке мусора.<br><br>*Тема 1.2. Установка и основные типы данных Python*<br>Способы установки Python. Числа. Строки. Словари. Списки. Множества. Ввод данных с клавиатуры.<br><br>*Тема 1.3. Операторы и управляющие конструкции Python*<br>Основные операторы Python. Зачем нужны управляющие конструкции? Конструкции ветвления. Циклы. Списковые включения. |
| **2.** | **Функции, модули, пакеты и их тестирование**<br>*Тема 2.1. Функция с точки зрения уровней абстракции и объявление функций в Python*<br>Функция как способ написания художественного произведения. Варианты объявления функций. Значения аргументов по умолчанию. Режим сопоставления аргументов функций. Возвращение результатов выполнения функции.<br><br>*Тема 2.2. Области видимости переменных, замыкание и аннотация функций*<br>Локальная область видимости вложенных функций и правило LEGB. Способы организации замыканий. Зачем нужен Type Hinting? Как использовать аннотацию типов в процессе разработки приложений. Псевдонимы типов.<br><br>*Тема 2.3. Лямбда-функции, рекурсия, декораторы и генераторы*<br>Что такое лямбда-функции и как их использовать. Виды рекурсии и их организация. Объявление и использование декораторов. Основы генераторов и в чем их отличие от списковых включений.<br><br>*Тема 2.4. Создание модулей, пакетов и принцип работы импортирования*<br>Что такое модуль? Зачем выделять код в отдельный модуль? Принцип работы импортирования. Создание и использование модулей и пакетов.<br><br>*Тема 2.5. Тестирование функций и работа со статическими анализаторами кода*<br>Тестирование функций с использованием библиотеки PyTest. Статический анализатор кода mypy. Следим за единым стилем в команде, используя flake8. |
| **3.** | **Объектно-ориентированное программирование** |

| | |
|---|---|
| | *Тема 3.1. Класс, как очередной уровень абстракции и основные принципы ООП*<br>Что такое класс и какие у него свойства? Абстракция в ООП. 3 стопа ООП: инкапсуляция, полиморфизм и наследование. Инкапсуляция vs сокрытие.<br><br>*Тема 3.2. Объявление класса в Python. Типы переменных и методов*<br>Объявление класса. Переменные и методы класса. Переменные и методы экземпляра класса. Статические переменные и методы класса.<br><br>*Тема 3.3. Наследование, абстрактные классы и переопределение методов*<br>Наследование в Python. Разница между базовым и абстрактным классом. Переопределение методов в производных классах. Приведение к интерфейсу базового класса.<br><br>*Тема 3.4. «Магические» (дандер) методы классов и Generics (Обобщения)*<br>Что такое магические методы класса? Перегрузка магических методов. Обобщенное программирование на примере реализации односвязного списка.<br><br>*Тема 3.5. Перечисления и классы данных*<br>Зачем нужны классы данных? Разбор модуля dataclasses. Для чего используются перечисления. Типы перечислений.<br><br>*Тема 3.6. Исключения и тестирование классов*<br>Зачем нужны исключения? Конструкция try…except…final. Пользовательские исключения. Работа с оператором assert. Тестирование классов. |
| **4.** | **Работа с файлами**<br>*Тема 4.1. Работа с файлами и директориями. Основные принципы*<br>Создание пустого файла, получение информации о файле, перемещение и переименование файла, удаление файла, проверка на наличие файла, запись и чтение из файла, копирование файла, форматированная запись в файл и чтение из него, создание и удаление каталога, создание цепочки каталогов,<br>создание и удаление временного каталога, получение списка файлов и каталогов в директории, получение пути до текущей директории проекта,<br>перемещение по директориям<br><br>*Тема 4.2. Пример работы с файлами*<br>Разработка моделей данных, разработка функционала сохранения и загрузки состояний моделей, запуск приложения |

| | |
|---|---|
| | *Тема 4.3. Пример работы с excel-файлами и визуализация данных*<br>Разработка молей, разработка парсера данных, сохранение и считывание базы данных, функционал проверки текущей успеваемости, функционал анализа разницы, конфигурируемый запуск приложения |
| **5.** | **Работа с СУБД SQLite**<br>*Тема 5.1. Основные команды SQLite*<br>Оператор CREATE, SELECT, INSERT, UPDATE, DELETE, DROP, JOIN,<br>WHERE, GROUP BY, HAVING, UNION, ALTER TABLE<br><br>*Тема 5.2. Пример работы с СУБД SQLite средствами Python*<br>Создание и заполнение базы дынных, Создание запросов к сотрудникам, Создание запросов к ассортименту пицц, Создание запросов к заказам, Создание главного меню, Создание меню «Сотрудники», Создание меню «Заказы», Создание меню «Ассортимент пицц»<br><br>*Тема 5.3. Работа с СУБД SQLite средствами ORM SQLAlchemy*<br>Объявление моделей SQLAlchemy, Создание и заполнение базы дынных, Создание запросов к сотрудникам, Создание запросов к ассортименту пицц,<br>Создание запросов к заказам, Запуск приложения<br><br>*Тема 5.4. Транзакции*<br>Что такое транзакция и какие ее типы поддерживает SQLite, просмотр генерируемых запросов к БД ORM SQLAlchemy |
| **6.** | **Потоки, процессы и асинхронное программирование в Python**<br>*Тема 6.1. Основные определения*<br>Что такое процесс, поток и в чем разница? Разница между конкурентным и параллельным выполнением кода. Что такое асинхронное программирование? Что и когда выбирать?<br><br>*Тема 6.2. Потоки и процессы в Python*<br>Создание и запуск потока. Синхронизация потоков.Обмен данными между потоками. Пул потоков. Процессы в Python. Создание и запуск процесса.<br>Синхронизация процессов. Обмен данными между процессами. Пул процессов.<br><br>*Тема 6.3. Асинхронное программирование в Python*<br>Базовая концепция цикла и очереди событий. Что такое сопрограммы и как их использовать? Конкурентное выполнение кода с использованием задач. Объекты Future при написании асинхронного кода. Ручное управление циклом событий. |
| **7.** | **Разработка телеграм-ботов** |

| | |
|---|---|
| | *Тема 7.1. Настройка рабочего окружения, создание моделей и инициализация БД*<br>Настройка окружения проекта, загрузка первичных данных из файла MS Excel, Создание моделей, Инициализация базы данных<br><br>*Тема 7.2 Разработка основного функционала телеграм-бота*<br>Разработка функциональности для запуска бота, отмечаем отсутствующих, добавляем дисциплину, группу и студентов, удаление студента и назначение дисциплины группе<br><br>*Тема 7.3 Генерация отчетов*<br>Генерация интерактивного отчета, Генерация отчетов в Excel |
| **8.** | **Итоговый проект**<br><br>**Цель работы**<br>Приобретение обучающимися профессиональных умений и навыков, опыта самостоятельной профессиональной деятельности и закрепление приобретенных компетенций.<br><br>**Задачи**<br>Для достижения поставленной цели требуется решить следующие задачи**:**<br>1. Рассмотреть предметную область и решить, из каких модулей будет состоять разрабатываемое программное обеспечение и какие функции они должны реализовывать.<br>2. Выбрать стек технологий (библиотеки для Python), посредством которого будет реализовано программное обеспечение и обосновать свой выбор.<br>3. Разработать структуру базы данных для реализуемого программного обеспечения.<br>4. Реализовать серверную часть (БД минимум на 5 таблиц, CRUD минимум на 10 запросов и т.д.).<br>5. Реализовать клиентскую часть (телеграм-бот, дашборд и т.д.).<br>6. Реализовать тестовое окружение программного обеспечения.<br>7. Протестировать работу реализованного программного обеспечения и оформить в отчете раздел с демонстрацией его работы<br>8. Загрузить исходный код программного обеспечения на github или gitverse |

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

| № п/п | Темы практических занятий | Формы практических занятий | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|---|---|---|---|---|---|
| Семестр 5 | | | | | |
| 1. | Работа со строками, списками и кортежами | онлайн выполнение практических заданий | 4 | 4 | 1 |
| 2. | Множества и словари | онлайн выполнение практических заданий | 4 | 4 | 1 |
| 3. | Арифметические операции, управляющие конструкции и циклы | онлайн выполнение практических заданий | 4 | 4 | 1 |
| 4. | Списковые включения и функции | онлайн выполнение практических заданий | 4 | 4 | 1,2 |
| 5. | Рекурсия и замыкания | онлайн выполнение практических заданий | 4 | 4 | 1,2 |
| 6. | Декораторы и генераторные функции | онлайн выполнение практических заданий | 4 | 4 | 1,2 |
| 7. | Объектно-ориентированное программирование | онлайн выполнение практических заданий | 5 | 5 | 1,3 |
| 8. | Магические методы классов | онлайн выполнение практических заданий | 5 | 5 | 1,3 |
| Семестр 6 | | | | | |
| 9. | Работа с TXT и JSON файлами | онлайн выполнение практических заданий | 5 | 5 | 1,3,4 |
| 10. | Работа с excel-файлами | онлайн выполнение практических заданий | 5 | 5 | 1,3,4 |
| 11. | Работа с SQLite средствами Python | онлайн выполнение практических заданий | 5 | 5 | 1,3,5 |
| 12. | Работа с SQLite средствами ORM SQLAlchemy | онлайн выполнение практических заданий | 5 | 5 | 1,3,5 |
| 13. | Потоки | онлайн выполнение практических заданий | 5 | 5 | 1,3,6 |
| 14. | Асинхронное программирование | онлайн выполнение практических заданий | 5 | 5 | 1,3,6 |
| 15. | Разработка | онлайн выполнение | 7 | 7 | 1,2,3,4, |

| | | | | | |
|---|---|---|---|---|---|
| | телеграм-бота | практических заданий | | | 5,6,7 |
| 16. | **Итоговый проект** | **онлайн выполнение практических заданий** | 14 | 14 | 1,2,3,4, 5,6,7 |
| | | | 51 | | |
| Всего | | | 85 | | |

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

| № п/п | Наименование лабораторных работ | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|---|---|---|---|---|
| Учебным планом не предусмотрено | | | | |
| | | | | |
| Всего | | | | |

4.5. Выполнение курсового проекта/ курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

| Вид самостоятельной работы | Всего, час | Семестр 5, час | Семестр 6, час |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| Изучение теоретического материала дисциплины (ТО) | 71 | 26 | 45 |
| Курсовое проектирование (КП, КР) | - | - | - |
| Расчетно-графические задания (РГЗ) | - | - | - |
| Выполнение реферата (Р) | - | - | - |
| Подготовка к текущему контролю успеваемости (ТКУ) | 34 | 17 | 17 |
| Домашнее задание (ДЗ) | - | - | - |
| Контрольные работы заочников (КРЗ) | - | - | - |
| Подготовка к промежуточной аттестации (ПА) | 28 | 14 | 14 |
| Всего: | 133 | 57 | 76 |

## 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

| Шифр/ URL адрес | Библиографическая ссылка | Количество экземпляров в библиотеке (кроме электронных экземпляров) |
|---|---|---|
| **Основная литература** | | |
| ЭБ ГУАП | Введение в язык программирования Python: [ Электронный ресурс] учеб.-метод. пособие / С. А. Чернышев. – СПб.: ГУАП, 2023 – 236 с. | - |
| https://urait.ru/bcode/544190 | Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — Текст : электронный // Образовательная платформа Юрайт | - |
| https://urait.ru/bcode/539651 | Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для вузов / Д. Ю. Федоров. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 227 с. — (Высшее образование). — ISBN 978-5-534-17323-9. — Текст : электронный // Образовательная платформа Юрайт | - |
| https://e.lanbook.com/book/241115 | Северанс, Ч. Р. Python для всех / Ч. Р. Северанс ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2021. — 262 с. — ISBN 978-5-93700-104-7. — Текст : электронный // Лань : электронно-библиотечная система. | - |
| https://urait.ru/bcode/544319 | Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2024. — 176 с. — (Высшее образование). — ISBN 978-5-534-14383-6 | - |
| **Дополнительная литература** | | |
| https://e.lanbook.com/book/241031 | Хилл, К. Научное программирование на Python / К. Хилл ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2021. — 646 с. — ISBN 978-5-97060-914-9. — Текст : электронный // Лань : электронно-библиотечная система. | - |
| https://e.lanbook.com/book/202154 | Борзунов, С. В. Алгебра и геометрия с | - |

| Шифр/ URL адрес | Библиографическая ссылка | Количество экземпляров в библиотеке (кроме электронных экземпляров) |
|---|---|---|
| | примерами на Python / С. В. Борзунов, С. Д. Кургалин. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 444 с. — ISBN 978-5-8114-9980-9. — Текст : электронный // Лань : электронно-библиотечная система. | |
| URL: https://e.lanbook.com/book/221678 | Щерба, А. В. Программирование на Python. Первые шаги / А. В. Щерба. — Москва : Лаборатория знаний, 2022. — 250 с. — ISBN 978-5-93208-578-3. — Текст : электронный // Лань : электронно-библиотечная система. | - |
| https://e.lanbook.com/book/297152 | Кольцов, Д. М. PYTHON. Создаем программы и игры : самоучитель / Д. М. Кольцов. — 3-е изд. — Санкт-Петербург : Наука и Техника, 2022. — 432 с. — ISBN 978-5-907592-01-8. — Текст : электронный // Лань : электронно-библиотечная система. | - |
| https://e.lanbook.com/book/402086 | Баркович, А. А. Python. Алгоритмизация и программирование. Решение лингвистических задач : учебное пособие / А. А. Баркович, Н. В. Зыгмантович, А. Е. Жданович. — 2-е изд., стер. — Москва : ФЛИНТА, 2024. — 100 с. — ISBN 978-5-9765-5447-4. — Текст : электронный // Лань : электронно-библиотечная система. | - |
| https://e.lanbook.com/book/293606 | Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин. — Ростов-на-Дону : ЮФУ, 2022. — 242 с. — ISBN 978-5-9275-4108-9. — Текст : электронный // Лань : электронно-библиотечная система. | - |

## 7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

| URL адрес | Наименование |
|---|---|
| https://stepik.org/course/58852 | «Поколение Python»: курс для начинающих |
| https://stepik.org/course/68343 | «Поколение Python»: курс для продвинутых |
| https://stepik.org/course/100707 | Добрый, добрый Python с Сергеем Балакиревым |
| https://stepik.org/course/182173 | Python + Aiogram для начинающих |
| https://stepik.org/course/176862 | Создание Телеграм ботов С НУЛЯ |
| https://stepik.org/course/122947 | Создание бота в Telegram на Python |
| https://stepik.org/course/191596 | SQLAlchemy — продвинутая работа с базой данных через Python |
| https://stepik.org/course/182183 | Python + SQL БД: PostgreSQL, MySQL, MariaDB для начинающих |

## 8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

| № п/п | Наименование |
|---|---|
| | Не предусмотрено |

8.2. Перечень информационно-справочных систем,используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

| № п/п | Наименование |
|---|---|
| | Не предусмотрено |

## 9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице12.

Таблица 12 – Состав материально-технической базы

| № п/п | Наименование составной части материально-технической базы | Номер аудитории (при необходимости) |
|---|---|---|
| 1 | Система дистанционного обучения ГУАП (СДО ГУАП) – lms.guap.ru | - |

## 10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средствдля проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

| Вид промежуточной аттестации | Перечень оценочных средств |
|---|---|
| Дифференцированный зачёт | Список вопросов;<br>Тесты; |
| Зачет | Список вопросов;<br>Тесты; |

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

| Оценка компетенции<br>5-балльная шкала | Характеристика сформированных компетенций |
|---|---|
| «отлично»<br>«зачтено» | Обучающийся:<br>– глубоко и всесторонне усвоил программный материал;<br>– уверенно, логично, последовательно и грамотно его излагает;<br>– опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления;<br>– умело обосновывает и аргументирует выдвигаемые им идеи;<br>– делает выводы и обобщения;<br>– свободно владеет системой специализированных понятий.<br>– правильно выполнил от 90% до 100% тестовых заданий**. |
| «хорошо»<br>«зачтено» | Обучающийся:<br>– твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы;<br>– не допускает существенных неточностей;<br>– увязывает усвоенные знания с практической деятельностью направления;<br>– аргументирует научные положения;<br>– делает выводы и обобщения;<br>– владеет системой специализированных понятий.<br>– правильно выполнил от 70% до 89% тестовых заданий**. |
| «удовлетворительно»<br>«зачтено» | – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы;<br>– допускает несущественные ошибки и неточности;<br>– испытывает затруднения в практическом применении знаний направления;<br>– слабо аргументирует научные положения;<br>– затрудняется в формулировании выводов и обобщений;<br>– частично владеет системой специализированных понятий.<br>– правильно выполнил от 51% до 69% тестовых заданий**. |
| «неудовлетворительно»<br>«не зачтено» | – обучающийся не усвоил значительной части программного материала;<br>– допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении;<br>– испытывает трудности в практическом применении знаний;<br>– не может аргументировать научные положения;<br>– не формулирует выводов и обобщений.<br>– правильно выполнил менее 51% тестовых заданий**. |

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

| № п/п | Перечень вопросов (задач) для экзамена | Код индикатора |
|---|---|---|
| | Учебным планом не предусмотрено | |

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

| № п/п | Примерный перечень вопросов (задач) для зачета / дифф. зачета | Код индикатора |
|---|---|---|
| 1. | Стиль оформления программы. Основные положения PEP8 | УК-6.З.1 |
| 2. | Что такое абстракция и принцип разработки через интерфейс? Как они связаны между собой? | УК-6.З.2 |
| 3. | Структурный подход и его особенности | УК-6.У.1 |
| 4. | Объектный подход и его особенности | УК-6.У.2 |
| 5. | Тестирование «черного» и «белого» ящиков | УК-6.В.1 |
| 6. | Абстракция в объектно-ориентированном программировании | УК-6.В.2 |
| 7. | Что такое абстрактный класс и интерфейс в рамках языка программирования? | УК-6.З.1 |
| 8. | Инструментальные среды разработки и сопровождения программных средств на языке Python | УК-6.З.2 |
| 9. | Что такое ORM? Какие преимущества дает ее использование? Приведите примеры. | УК-6.У.1 |
| 10. | Принципы обработки исключительных ситуаций в Python | УК-6.У.2 |
| 11. | Что такое абстракция? Как она используется при выстраивании структуры проекта? | УК-6.В.1 |
| 12. | Что такое абстракция? Как она используется при функциональном программировании? | УК-6.З.1 |

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

| № п/п | Примерный перечень тем для выполнения курсового проекта/ курсовой работы |
|---|---|
| | Учебным планом не предусмотрено |

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

| № п/п | Примерный перечень вопросов для тестов | Код индикатора |
|---|---|---|
| 1. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какая из перечисленных функций позволяет открыть файл для последующей с работы ним:<br>• **open**<br>• with<br>• open_file<br>• file | УК-6.З.1 |
| 2. | **Прочитайте текст и выберите один правильные ответы.**<br><br>Сопоставьте флаг режима открытия файла с его описанием:<br>• **"r" (read)** - открыть файл для чтения. Если файл не существует, будет сгенерировано исключение.<br>• **"w" (write)** - открыть файл для записи. Если файл уже существует, его содержимое будет удалено. Если файл не существует, он будет создан.<br>• **"a" (append)** - открыть файл для добавления данных в | УК-6.З.2 |

| | | |
|---|---|---|
| | конец. Если файл не существует, он будет создан.<br>• **"x" (exclusive)** - открыть файл для записи, но только если он не существует. Если файл уже существует, будет сгенерировано исключение.<br>• **"b" (binary)** - открыть файл в бинарном режиме. Этот режим используется для работы с файлами, содержащими бинарные данные, такие как изображения или звуковые файлы. | |
| 3. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какая функция модуля *os* используется для перемещения файла?<br>• **rename**<br>• move<br>• remove<br>• move_to | УК-6.У.1 |
| 4. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какая из перечисленных библиотек используется для генерации ложных данных для тестовой инициализации базы данных?<br>• **Faker**<br>• MyPy<br>• SQLAlchemy<br>• GenFake | УК-6.У.2 |
| 5. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какая из перечисленных библиотек используется для работы с Excel-файлами?<br>• **openpyxl**<br>• numpy<br>• matplotlib<br>• xlsdoc | УК-6.В.1 |
| 6. | **Прочитайте текст и выберите один правильные ответы.**<br><br>Укажите, какие из перечисленных библиотек используется для построения графиков?<br>• openpyxl<br>• numpy<br>• **matplotlib**<br>• **plotly** | УК-6.В.2 |
| 7. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какой из перечисленных модулей используется для конфигурирования запускаемого приложения?<br>• os<br>• process<br>• **getopt**<br>• sys | УК-6.З.1 |
| 8. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какой из перечисленных модулей используется для | УК-6.З.2 |

| | | |
|---|---|---|
| | работы с СУБД SQLite?<br>• sql<br>• sqlite<br>• sqlaclchemy<br>• **sqlite3** | |
| 9. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какой из перечисленных модулей отвечает за работу с потоками?<br>• **threading**<br>• multiprocessing<br>• asyncio<br>• thread | УК-6.У.1 |
| 10. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какой из перечисленных модулей отвечает за работу с процессами?<br>• threading<br>• **multiprocessing**<br>• asyncio<br>• process | УК-6.У.2 |
| 11. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какой из перечисленных модулей отвечает за работу выполнение асинхронных операций?<br>• threading<br>• multiprocessing<br>• **asyncio**<br>• async | УК-6.В.1 |
| 12. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, какая из перечисленных ниже функций модуля asyncio позволяет выполнить код в отдельном потоке?<br>• **to_thread**<br>• gather<br>• thread<br>• process | УК-6.В.2 |
| 13. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, в модуле с каким именем следует хранить функции для создания записей, обновления, удаления данных в базе данных?<br>• **crud**<br>• models<br>• data<br>• database | УК-6.З.1 |
| 14. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, в пакете с каким именем следует хранить модели данных, используемые при описании таблиц SQLAlchemy?<br>• crud<br>• **model** | УК-6.З.2 |

| | | |
|---|---|---|
| | • data<br>• database | |
| 15. | **Прочитайте текст и выберите один правильный ответ.**<br><br>Укажите, как расшифровывается аббревиатура ORM?<br>• **Object-Relational Mapping**<br>• Object-Relation Mapping<br>• Object-Relational Map<br>• Object-Relation Map | УК-6.У.1 |

Ключи правильных ответов на тесты размещены в Приложении 1 к РПД и находятся у специалистов по УМР кафедры 41, заместителя заведующего кафедрой и руководителя образовательной программы.

Система оценивания тестовых заданий показана в таблице 18.1

Таблица 18.1 – Система оценивания тестовых заданий

| № | Указания по оцениванию | Результат оценивания (баллы, полученные за выполнение \ характеристика правильности ответа) |
|---|---|---|
| 1 | Задание закрытого типа на установление соответствия считается верным, если установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого столбца) | Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов (либо указывается «верно» \ «неверно») |
| 2 | Задание закрытого типа на установление последовательности считается верным, если правильно указана вся последовательность цифр | Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов (либо указывается «верно»\ «неверно») |
| 3 | Задание комбинированного типа с выбором одного верного ответа из четырех предложенных и обоснованием выбора считается верным, если правильно указана цифра и приведены конкретные аргументы, используемые при выборе ответа | Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов (либо указывается «верно»\ «неверно») |
| 4 | Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развернутым обоснованием выбора считается верным, если правильно указаны цифры и приведены конкретные аргументы, используемые при выборе ответов | Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов (либо указывается «верно»\ «неверно») |
| 5 | Задание открытого типа с развернутым ответом считается верным, если ответ совпадает с эталонным по содержанию и | Правильный ответ за задание оценивается в 3 балла, если допущена одна ошибка \ неточность \ ответ правильный, но не полный - 1 балл, |

| | | |
|---|---|---|
| | полноте | если допущено более 1 ошибки \ ответ неправильный \ ответ отсутствует – 0 баллов (либо указывается «верно»\ «неверно») |

Инструкция по выполнению тестового задания находится в таблице 18.2.

Таблица 18.2 - Инструкция по выполнению тестового задания

| № | Тип задания | Инструкция |
|---|---|---|
| 1 | Задание закрытого типа на установление соответствия | Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце |
| 2 | Задание закрытого типа на установление последовательности | Прочитайте текст и установите последовательность<br>Запишите соответствующую последовательность букв слева направо |
| 3 | Задание комбинированного типа с выбором одного верного ответа из четырех предложенных и обоснованием выбора | Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа |
| 4 | Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развернутым обоснованием выбора | Прочитайте текст, выберите правильные варианты ответа и запишите аргументы, обосновывающие выбор ответов |
| 5 | Задание открытого типа с развернутым ответом | Прочитайте текст и запишите развернутый обоснованный ответ |

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

| № п/п | Перечень контрольных работ |
|---|---|
| | Не предусмотрено |

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

## 11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших

достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

– получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;

– получение опыта творческой работы совместно с преподавателем;

– развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.

– появление необходимого интереса, необходимого для самостоятельной работы;

– получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;

– научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);

– получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

– изложение теоретических вопросов, связанных с рассматриваемой темой;

– мастер-классы;

– обобщение изложенного материала.

11.2. Методические указания для обучающихся по участию в семинарах

**Не предусмотрено учебным планом**

11.3. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимся практических занятий:

– закрепление, углубление, расширение и детализация знаний при решении конкретных задач;

– развитие познавательных способностей, самостоятельности мышления, творческой активности;

– овладение новыми методами и методиками изучения конкретной учебной дисциплины;

– выработка способности логического осмысления полученных знаний для выполнения заданий;

– обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Требования к проведению практических занятий

Отчет о выполнении практической работы должен включать в себя: титульный лист, формулировку цели работы, формулировку задания, полученные результаты в виде скриншотов с необходимыми пояснениями. Файл с отчетом, а также все программные коды должны быть загружены в LMS.

Практические занятия проводятся с применением электронного обучения, дистанционных образовательных технологий (онлайн-курс).

Курс размещён в системе дистанционного обучения ГУАП (СДО ГУАП).

## Методические указания и требования к итоговому проекту

### Цель работы

Приобретение обучающимися профессиональных умений и навыков, опыта самостоятельной профессиональной деятельности и закрепление приобретенных компетенций.

### Задачи

Для достижения поставленной цели требуется решить следующие задачи**:**

1. Рассмотреть предметную область и решить, из каких модулей будет состоять разрабатываемое программное обеспечение и какие функции они должны реализовывать.
2. Выбрать стек технологий (библиотеки для Python), посредством которого будет реализовано программное обеспечение и обосновать свой выбор.
3. Разработать структуру базы данных для реализуемого программного обеспечения.
4. Реализовать серверную часть (БД минимум на 5 таблиц, CRUD минимум на 10 запросов и т.д.).
5. Реализовать клиентскую часть (телеграм-бот, дашборд и т.д.).
6. Реализовать тестовое окружение программного обеспечения.
7. Протестировать работу реализованного программного обеспечения и оформить в отчете раздел с демонстрацией его работы
8. Загрузить исходный код программного обеспечения на github или gitverse

### Общие рекомендации к выполнению итогового проекта

Для реализации поставленных задач и формировании отчета, выполнение практико-ориентированного проекта можно осуществлять индивидуально или в команде.

1. При индивидуальном выполнении реализуется только телеграм-бот без стороннего программного обеспечения (дашборд на Django или FastAPI и т.д.)
2. Команда может состоять из 2-3 человек. Каждый студент должен разработать не менее 1-го модуля проекта. Команда должна выполнить все задания, которые предложены для реализации.
3. Отчет выполняется в виде многостраничного документа объемом не менее 20-25 страниц при командном выполнении или 13-15 страниц при выборе индивидуального выполнения. На титульном листе отчета должны быть написаны фамилии всех студентов, которые принимали участие в проекте. Во введении должно быть четко указано, за какой раздел отчета отвечает конкретный студент. В заключении должна быть ссылка на исходный код в репозитории.

Например. Данный практико-ориентированный проект выполнялся четырьмя студентами из группы 1156: Н.А. Ивановым, А.Т. Тетериным, Д.А., Левашевым и Н.Б. Кругловым.

- Проектирование базы данных Н.А. Иванов, А.Т. Тетерин.
- Разработка тестового окружения Д.А. Левашев. 
  Разработка серверной части Д.А. Левашев и Н.Б. Круглов и т.д.

Каждый студент выкладывает полную версию отчета. Оценка выставляется по результатам проверки отчета в целом без дифференцирования по конкретному студенту.

Последовательность выполнения задания

1. Первым шагом является подготовка титульного листа, который должен соответствовать стандартному формату курсовой работы. В нем указывается название проекта, ФИО исполнителей, номер группы и имя руководителя.
2. Следующий этап – составление аннотации, в которой кратко описывается суть проекта и распределение задач между участниками команды (если работа выполняется коллективно).
3. После этого необходимо определить цель работы и конкретные задачи, которые помогут ее достичь. Это позволяет сформировать четкое представление о конечном результате и установить критерии его успешности.
4. Далее следует изучение предметной области и существующих аналогов программного обеспечения. Этот этап важен для выявления преимуществ и недостатков уже существующих решений, а также для обоснования необходимости разработки нового программного продукта.
5. Выбор стека используемых технологий играет ключевую роль в успешности проекта. Необходимо определить, какие языки программирования, библиотеки, базы данных и инструменты тестирования будут использованы, и пояснить, почему именно они подходят для реализации поставленной задачи.
6. Архитектура проекта должна быть четко продумана и зафиксирована в виде схем и диаграмм. Это может включать диаграмму классов, структуру базы данных или схему взаимодействия компонентов. Такая визуализация позволяет лучше понять внутреннюю организацию системы и избежать возможных проблем в процессе разработки.
7. Этап реализации предполагает непосредственное написание кода. Здесь важно придерживаться принципов модульности и следовать стандартам кодирования. В описании реализации следует подробно рассмотреть основные алгоритмы, взаимодействие между модулями и использованные зависимости.
8. Разработка тестового окружения является неотъемлемой частью процесса. Тестирование помогает выявить ошибки и проверить корректность работы системы в различных условиях. Важно описать стратегию тестирования, привести примеры тест-кейсов и приложить результаты тестирования в виде скриншотов.
9. После выполнения всех технических этапов необходимо подвести итоги работы, проанализировать достигнутые результаты и наметить направления для возможного развития проекта. Этот раздел оформляется в виде эссе, в котором можно рассказать о приобретенных знаниях, возникших сложностях и способах их преодоления.

10. Завершающим этапом является оформление списка использованных источников, который должен включать книги, статьи, официальную документацию и другие материалы, использованные при выполнении работы.
11. Выполните отчет в соответствие с требованиями к структуре и наполнению.
12. Перечитайте раздел 2, в котором указаны критерии оценивания. Проверьте, что требования соблюдены.

Структура отчет по практической работе

1. Титульный лист (соответствует титульному листу курсовой работы https://guap.ru/regdocs/docs/uch)
2. Аннотация, в которой написано, какие разделы отчета выполнены каким студентом.
3. Цель работы и задачи.
4. Краткое описание предметной области и аналогов разрабатываемого программного обеспечения.
5. Выбор стека используемых технологий.
6. Архитектура проекта (структура кода, диаграммы).
7. Описание реализации (основные алгоритмы, модули, зависимости).
8. Разработка тестового окружения (скриншоты, описания тестов).
9. Выводы и анализ (полученные результаты, возможные доработки). Вывод в формате эссе, в котором раскройте цель выполненной работы, описываются знания и навыки, полученные в процессе ее выполнения, а также возникшие проблемы и пути их решения.
10. Список использованных источников.

**Обратите внимание, что скриншоты должны быть связаны поясняющим текстом.**

Критерии оценивания

Оценивание полноты выполнения и оформления отчета по таблице приведены критерии оценивания, на которые будут ориентироваться преподаватели при выставлении балла. В системе СДО возможен автоматический пересчет в рейтинговые баллы, которые Вы увидите в личном кабинете в графе «Оценки» по критериям, представленным в табл.20. Если отчет выполнен позже установленного срока, то он будет оценен с понижающим коэффициентом.

Таблица 20 – Критерии оценивания практической работы

| № | Критерий | Рейтинговый балл |
|---|---|---|
| 1. | Аннотация | Да – 0,5 балла.<br>Нет – 0 баллов. |
| 2. | Описание цели и задач практической работы, содержание отчета | Да – 0,5 балла.<br>Нет – 0 баллов. |

| | | |
|---|---|---|
| 3. | Краткое описание предметной области и аналогов разрабатываемого программного обеспечения | Да – 2 балл.<br>Нет – 0 баллов |
| 4. | Выбор стека используемых технологий | Да – 1 балл.<br>Нет – 0 баллов |
| 5. | Архитектура проекта | Да – 1,5 балла.<br>Нет – 0 баллов. |
| 6. | Описание реализации | Да – 1 балла.<br>Нет – 0 баллов. |
| 7. | Разработка тестового окружения | Да – 2 балла.<br>Нет – 0 баллов. |
| 8. | Выводы и анализ | Да – 1 балла.<br>Нет – 0 баллов. |
| 9. | Список использованных источников | Да – 0,5 балла.<br>Нет – 0 баллов |
| **Максимально возможный балл** | | **10 (зачтено)** |

Таблица 21 – Перевод баллов в оценку

| **Баллы** | **Оценка** | **Зачет/не зачет** |
|---|---|---|
| Менее 4 баллов | неудовлетворительно | Не зачет |
| 4-6 баллов | удовлетворительно | Зачет |
| 7-8 балов | хорошо | |
| 9-10 баллов | отлично | |

Таблица 22 – Примеры тем итоговых проектов

| | |
|---|---|
| 1) | Разработка системы для записи на прием к стоматологу с использованием телеграмм-бота (бот + админка и дашборд) |
| 2) | Разработка прототипа системы для записи на шиномонтаж с использованием телеграмм-бота (бот + админка и дашборд) |
| 3) | Разработка телеграмм-бота расписания кафедры (бот + админка) |

| | |
|---|---|
| 4) | Разработка телеграмм-бота для проведения голосования и опросов, а также программного обеспечения для их формирования и визуализации ответов |
| 5) | Разработка телеграмм-бота для отслеживания посещения студентами занятий и проставления оценок, а также программного обеспечения для редактирования базы данных групп и предметов кафедры |
| 6) | Разработка телеграмм-бота для отслеживания потраченных за месяц денег с функцией графического представления информации |
| 7) | Разработка телеграмм-бота «книга рецептов», а также программного обеспечения для их формирования |

11.4. Методические указания для обучающихся по выполнению лабораторных работ

**Не предусмотрено учебным планом**

11.5. Методические указания для обучающихся по прохождению курсового проектирования/выполнения курсовой работы

**Не предусмотрено учебным планом**

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

В процессе выполнения самостоятельной работы у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет ему развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическим материалом, направляющим самостоятельную работу обучающихся, является учебно-методический материал по дисциплине.

Материалы для самостоятельной работы представлены с применением электронного обучения, дистанционных образовательных технологий (онлайн-курс).

Курс размещён в системе дистанционного обучения ГУАП (СДО ГУАП).

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

**В первом семестре обучения** промежуточная аттестация представляет собой **зачет** – форму оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено». Студентам, набравшим по итогам обучения **от 75 до 100 баллов** выставляется оценка «Зачтено». Студентам, набравшим **75 и менее** баллов предстоит сдача итогового тестирования с автоматическим прокторингом в соответствии с календарным графиком учебного процесса.

**Во втором семестре обучения** промежуточная аттестация представляет собой **дифференцированный зачет** – форму оценки знаний, полученных обучающимся при изучении дисциплины, при выполнении итоговых и курсовых проектов, курсовых работ, научно-исследовательских работ и прохождении практик с аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Для получения **дифференцированного зачета** студентам предстоит набрать за семестр минимум 50 баллов, выполнить итоговый проект (индивидуально или в команде) минимум на оценку «удовлетворительно» и пройти итоговое тестирование. **Итоговая оценка ставится только после защиты итогового проекта по ИТ-модулю и состоит из суммы баллов, приведенной к 100: за работу в течение семестра и баллы за итоговый тест.**

Лист внесения изменений в рабочую программу дисциплины

| Дата внесения изменений и дополнений. Подпись внесшего изменения | Содержание изменений и дополнений | Дата и № протокола заседания кафедры | Подпись зав. кафедрой |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |