

УТВЕРЖДАЮ
Проректор по учебной работе и
качеству образования

_____ И. А. Долгова

30 сентября 2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФТД.02

Создание и внедрение чат-ботов

Направление подготовки:	45.04.04 Интеллектуальные системы в гуманитарной среде
Профиль подготовки:	Цифровая лингвистика
Квалификация:	магистр
Форма обучения:	очная
Год начала подготовки:	2027

Рабочая программа составлена в соответствии с:

- федеральным государственным образовательным стандартом высшего образования – магистратура по направлению подготовки 45.04.04 Интеллектуальные системы в гуманитарной среде, утверждённым приказом Министерства науки и высшего образования Российской Федерации от 13 августа 2020 года № 1003.

Разработчики программы: Макаров А. А., доктор технических наук, профессор; Кириллов А. Г., кандидат филологических наук, доцент

Рабочая программа согласована с руководителем образовательной программы 45.04.04 Интеллектуальные системы в гуманитарной среде. Рабочая программа рассмотрена и одобрена на заседании кафедры информационных систем и компьютерных технологий 10 сентября 2026, протокол № 2.

1. ОБЩАЯ ХАРАКТЕРИСТИКА

1.1. Цель и задачи дисциплины

Цель изучения дисциплины — формирование у обучающихся теоретических представлений об архитектуре диалоговых систем (чат-ботов) и практических навыков интеграции программного кода, развертывания, тестирования и комплексной отладки программного обеспечения в профессиональной сфере.

Задачи изучения дисциплины:

- изучение архитектурных стандартов, моделей ветвления диалога и требований к развертыванию интеллектуальных чат-ботов;
- освоение методов интеграции написанного программного кода со встроенными базами данных для обеспечения сохранения контекста общения;
- формирование базовых навыков проведения модульного тестирования (Unit Testing), поиска, локализации и устранения синтаксических и логических ошибок в коде;
- развитие умений публиковать готовые программные релизы в системах контроля версий и составлять сопутствующую техническую документацию.

1.2. Требования подготовленности обучающегося к освоению содержания дисциплины

Освоение дисциплины «Создание и внедрение чат-ботов» базируется на ИТ-компетенциях и навыках кодинга, полученных магистрантами во 2 семестре в рамках курса «Язык программирования Python» и параллельного курса 3 семестра «Программирование в гуманитарной сфере».

Обучающийся перед началом освоения дисциплины должен:

Знать:

- синтаксис языка Python, правила проектирования пользовательских функций и стандарт оформления кода PEP 8;
- базовые SQL-команды манипулирования данными и структуру таблиц реляционных СУБД;
- основы работы с распределенной системой контроля версий Git.

Уметь:

- разрабатывать и запускать локальные программные скрипты в среде Geany;
- осуществлять чтение, запись и потоковую обработку структурированных файлов данных формата JSON.

Владеть:

- базовыми навыками документирования кода с использованием комментариев;
- навыками работы в консоли (терминале) операционной системы.

1.3. Планируемые результаты обучения по дисциплине

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
ПК-2. Способен применять средства интеллектуального анализа данных и технологии искусственного интеллекта для решения	ПК-2.1. Разрабатывает алгоритмы и программные скрипты на современных языках программирования для автоматизации сбора и	ПК-2.1.4. Знает стандарты и требования к версионности, тестированию и развёртыванию программного обеспечения в профессиональной сфере

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
прикладных лингвистических задач	предварительной обработки языковых данных	ПК-2.1.5. Умеет интегрировать написанный программный код и базы данных в единую работающую программную систему, устранять ошибки и проводить финальную отладку

2. ОБЪЁМ, СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1. Объём дисциплины

Семестр	Контактная работа			СР	Форма ПА	Итоговый объём, часов/з. е.
	Л	П	ПА			
4	–	12	–	60	зачёт	72/2
Итого:	–	12	–	60		72/2

Л — лекции, П — все виды занятий семинарского типа, ПА — промежуточная аттестация, СР — самостоятельная работа обучающегося.

2.2. Структура дисциплины

Наименование тематического раздела дисциплины	Количество часов		
	Л	П	СР
Архитектура чат-ботов и интеграция с базами данных	–	4	30
Тестирование программного обеспечения и отладка кода	–	4	30
Версионность, развертывание и релиз-менеджмент	–	4	30
Всего:	–	12	60

2.3. Содержание тематических разделов дисциплины

Раздел 1. Архитектура чат-ботов и интеграция с базами данных
Принципы функционирования диалоговых систем. Архитектура асинхронных скриптов (на примере библиотеки <code>python-telegram-bot</code>). Интеграция программного кода бота со встроенной СУБД SQLite (модуль <code>sqlite3</code>). Разработка алгоритмов сохранения состояния диалога (State Machine), запись и чтение логов общения пользователей из таблиц локальной базы данных.
Раздел 2. Тестирование программного обеспечения и отладка кода
Стандарты и методология тестирования ПО в профессиональной сфере. Основы модульного тестирования (Unit Testing). Симуляция критических ошибок (багов) в логике бота (некорректный ввод данных, сбои базы данных, разрыв сетевого соединения). Лабораторный практикум по перехвату исключений <code>try-except</code> , логированию ошибок, отладке и финальному рефакторингу программного кода.
Раздел 3. Версионность, развертывание и релиз-менеджмент
Требования к версионности и развертыванию программных систем. Структурирование файлов проекта. Управление зависимостями (создание файлов <code>requirements.txt</code>). Практикум по фиксации стабильного релиза в локальной системе Git (тегирование версий). Публикация готового задокументированного программного продукта в удаленном репозитории на платформе GitHub.

2.4. Организация учебных занятий

Дисциплина может реализовываться с применением электронного обучения и дистанционных образовательных технологий.

3. ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

3.1. Учебная литература

3.1.1. Основная

1. Воробьев, А. Е. Введение в искусственный интеллект : учебное пособие : [16+] / А. Е. Воробьев, К. А. Воробьев, К. К. Кушеков. – Москва ; Вологда : Инфра-Инженерия, 2026. – 132 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=725633>.
2. Киселев, Г. М. Информационные технологии в педагогическом образовании : учебник / Г. М. Киселев, Р. В. Бочкова. – 6-е изд., стер. – Москва : Дашков и К°, 2024. – 300 с. : ил., табл., схем. – (Учебные издания для бакалавров). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=711130>.
3. Минаков, А. И. Искусственный интеллект и нейросети в образовании : учебник : [16+] / А. И. Минаков. – Москва : Директ-Медиа, 2024. – 164 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=715303>.

3.1.2. Дополнительная

1. Евстафьев, В. А. Искусственный интеллект и нейросети : практика применения в рекламе : учебное пособие / В. А. Евстафьев, М. А. Тюков. – 4-е изд. – Москва : Дашков и К°, 2026. – 426 с. : ил., табл. – (Учебные издания для вузов). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=731187>.
2. Кревецкий, А. В. Основы технологий искусственного интеллекта : учебное пособие : [16+] / А. В. Кревецкий, Ю. А. Ипатов, Н. И. Роженцова ; под общ. ред. А. В. Кревцового ; Поволжский государственный технологический университет. – Йошкар-Ола : Поволжский государственный технологический университет, 2023. – 272 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=714624>.

3.2. Профессиональные базы данных и информационные справочные системы

Наименование ресурса	Адрес	Тип доступа
Электронно-библиотечные системы		
ЭБС «Университетская библиотека онлайн»	https://biblioclub.ru/	Индивидуальный неограниченный доступ после регистрации
Профессиональные базы данных и информационные справочные системы		
Документация модуля unittest в Python	https://docs.python.org/3/library/unittest.html	Открытый ресурс
Официальный репозиторий python-telegram-bot	https://github.com/python-telegram-bot/python-telegram-bot	Открытый ресурс

3.3. Сетевые ресурсы

Наименование ресурса	Адрес
Всё о чат-ботах: виды, преимущества, успешные примеры	https://practicum.yandex.ru/blog/chto-takoe-chat-bot/
Документация по разработке Telegram бота	https://core.telegram.org/
GitHub	https://github.com

3.4. Методическое обеспечение дисциплины

3.4.1. Методические указания для преподавателя

Изучение дисциплины проводится в форме лекций, практических занятий, организации самостоятельной работы студентов, консультаций.

Лекции реализуются через изложение учебного материала с возможным мультимедийным сопровождением. Основными целями лекции являются системное освещение ключевых понятий и положений по соответствующей теме, обзор и оценка существующей проблематики, её методологических и социокультурных оснований, возможных вариантов решения, предложение методических рекомендаций для дальнейшего изучения курса, в том числе литературы и источников. Лектор должен стимулировать обучающихся к участию в обсуждении вопросов лекционного занятия, к высказыванию собственной точки зрения по обсуждаемой проблеме. Главное назначение лекции — обеспечить теоретическую основу обучения, развить интерес к учебной деятельности и конкретной учебной дисциплине, сформировать у обучающихся ориентиры для самостоятельной работы над курсом.

Практические занятия направлены на развитие самостоятельности обучающихся в исследовании изучаемых вопросов и приобретение умений и навыков. Практические занятия традиционно проводятся в форме обсуждения проблемных вопросов в группе при активном участии обучающихся. Они способствуют углубленному изучению наиболее фундаментальных и сложных проблем курса, служат важной формой анализа и синтеза исследуемого материала, а также подведения итогов самостоятельной работы обучающихся, стимулируя развитие профессиональной компетентности, навыков и умений. На практических занятиях обучающиеся учатся работать с научной литературой, чётко и доходчиво излагать проблемы и предлагать варианты их решения, аргументировать свою позицию, оценивать и критиковать позиции других, свободно публично высказывать свои мысли и суждения, грамотно вести полемику и представлять результаты собственных исследований. Практические занятия проводятся в форме устных и письменных опросов, диспута, тестирования, выполнения заданий, обсуждения докладов, выполнения контрольных заданий и пр.

Результаты работы на практических занятиях учитываются преподавателем при выставлении итоговой оценки по данной дисциплине. На усмотрение преподавателя обучающиеся, активно отвечающие на занятиях и выполняющие рекомендации преподавателя при подготовке к ним, могут получить повышающий балл к своей оценке в рамках промежуточной аттестации.

3.4.2. Методические рекомендации по самостоятельной работе обучающихся

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью

самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Самостоятельная работа включается в общую трудоёмкость учебной нагрузки обучающегося.

Реализация поставленной цели предполагает решение следующих задач:

- освоение и расширение теоретических знаний по изучаемой дисциплине;
- систематизация и закрепление полученных теоретических знаний и практических навыков;
- формирование умений по поиску и использованию нормативной, справочной и специальной литературы, а также других источников информации;
- развитие познавательных способностей и активности, творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самообразованию, самосовершенствованию и самореализации;
- развитие научно-исследовательских навыков;
- формирование умения решать практические задачи (в профессиональной деятельности), используя приобретённые знания, способности и навыки.

Самостоятельная работа должна быть выполнена индивидуально или являться частью коллективной работы (в случае выполнения группового задания в работе делается соответствующая оговорка).

3.5. Материально-техническое обеспечение дисциплины

3.5.1. Аудитории для проведения занятий

Специальные помещения представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, выполнения курсовых работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования.

Специальные помещения укомплектованы учебной мебелью, в том числе мебелью для преподавателя дисциплины, учебной доской.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечиваются электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Материально-техническое оснащение учебных аудиторий конкретизировано на официальном сайте Университета в подразделе «Материально-техническое обеспечение и оснащённость образовательного процесса. Доступная среда» раздела «Сведения об образовательной организации».

3.5.2. Оборудование и технические средства обучения

Специальные помещения укомплектованы демонстрационным оборудованием (мультимедийный проектор, экран, компьютер, звуковые колонки).

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

Электронно-библиотечная система (электронная библиотека) и электронная информационно-образовательная среда обеспечивают одновременный доступ не менее 25 процентов обучающихся.

3.5.3. Программное обеспечение

Наименование	Сведения о лицензии
Aimylogic, конструктор ботов	Сервис с ограниченным бесплатным функционалом
Moodle, среда дистанционного обучения	GNU GPL, свободно распространяемое с открытым исходным кодом
PuzzleBot ://, конструктор ботов для Telegram	Сервис с ограниченным бесплатным функционалом
Geany	Свободная (GNU General Public License v2+)
Python (включая модуль sqlite3 и библиотеку unittest)	Свободная (Python Software Foundation License — PSFL)
Git	Свободная (GNU GPLv2)

4. ОЦЕНОЧНЫЕ СРЕДСТВА

4.1. Методика проведения текущего контроля успеваемости и промежуточной аттестации

Оценивание уровня учебных достижений обучающихся по дисциплине осуществляется в виде текущего контроля успеваемости и промежуточной аттестации. Фонд оценочных средств по дисциплине включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

4.1.1. Балльно-рейтинговая карта дисциплины

Виды работы	Виды контроля, критерии оценки и количество баллов
Текущий контроль	
Аудиторная работа (0-20 баллов)	Посещение занятий и участие в работе: 10 баллов — посещение не менее 50% занятий 15 баллов — посещение 60-80% занятий, участие в обсуждениях материала 20 баллов — посещение 90-100% занятий, активное участие в обсуждениях материала
Самостоятельная работа (0-40 баллов)	
Промежуточная аттестация	
Контрольное задание (0-40 баллов)	Выполнение письменного контрольного задания в СДО Moodle

4.1.2. Шкала перевода рейтинговых баллов в пятибалльную систему оценивания

Общее количество баллов	Оценка сформированности компетенций	Оценка результатов обучения по дисциплине	Оценка ECTS
0–34	Компетенции не сформированы.	неудовлетворительно (не зачтено)	F
Теоретическое содержание не освоено, практические навыки не сформированы, большинство учебных заданий не выполнено, качество их выполнения минимальное, все задания содержат грубые ошибки. Обучающийся не готов решать типовые профессиональные задачи.			
35–49	Уровень владения компетенциями недостаточный для их формирования в результате обучения по дисциплине.	неудовлетворительно (не зачтено)	FX
Теоретическое содержание освоено частично, практические навыки не сформированы, большинство учебных заданий не выполнено, качество их выполнения минимальное, большинство заданий выполнено с ошибками. Обучающийся не готов решать типовые профессиональные задачи.			
50–59	Уровень владения компетенциями	удовлетворительно	E

Общее количество баллов	Оценка сформированности компетенций	Оценка результатов обучения по дисциплине	Оценка ECTS
	посредственный для их формирования в результате обучения по дисциплине.	(зачтено)	
Теоретическое содержание освоено частично, практические навыки сформированы фрагментарно, многие учебные задания не выполнены, качество их выполнения минимальное, многие задания выполнены с ошибками. Обучающийся готов решать типовые профессиональные задачи.			
60–69	Уровень владения компетенциями удовлетворительный для их формирования в результате обучения по дисциплине.	удовлетворительно (зачтено)	D
Теоретическое содержание освоено частично, пробелы не носят существенного характера, практические навыки в основном сформированы, большинство учебных заданий выполнено, качество их выполнения удовлетворительное, некоторые задания выполнены с ошибками. Обучающийся готов решать типовые профессиональные задачи.			
70–89	Уровень владения компетенциями преимущественно высокий для их формирования в результате обучения по дисциплине.	хорошо (зачтено)	C
Теоретическое содержание освоено полностью, некоторые практические навыки сформированы недостаточно, все учебные задания выполнены, качество их выполнения высокое, некоторые задания выполнены с ошибками. Обучающийся готов решать типовые и ситуативные профессиональные задачи.			
90–94	Уровень владения компетенциями высокий для их формирования в результате обучения по дисциплине.	отлично (зачтено)	B
Теоретическое содержание освоено полностью, практические навыки сформированы, все учебные задания выполнены, качество их выполнения близко к максимальному, однако есть несколько незначительных ошибок. Обучающийся готов эффективно решать типовые и ситуативные профессиональные задачи, в том числе повышенного уровня сложности.			
95–100	Уровень владения компетенциями превосходный для их формирования в результате обучения по дисциплине.	отлично (зачтено)	A
Теоретическое содержание освоено полностью, практические навыки сформированы, все учебные задания выполнены, качество их выполнения близко к максимальному. Обучающийся готов эффективно решать типовые и ситуативные профессиональные задачи, в том числе повышенного уровня сложности, и способен разрабатывать новые решения.			

4.2. Оценочные средства текущего контроля

4.2.1. Вопросы для подготовки к семинарским занятиям

Раздел 1. Архитектура чат-ботов и интеграция с базами данных

1. Схема взаимодействия и жизненный цикл запроса: Опишите архитектуру сквозного взаимодействия по схеме «пользователь — Telegram — сервер бота». Какую роль в этом

- процессе играет BotFather, что такое API-токен и по каким причинам его публикация в открытом доступе компрометирует безопасность системы?
2. Методы получения обновлений: В чем заключается принципиальное технологическое различие между методами получения данных Long Polling и Webhooks? Каковы критерии выбора между ними при проектировании архитектуры диалоговой системы?
 3. Асинхронное программирование в ботах: Почему язык Python доминирует в сфере разработки чат-ботов? Зачем в архитектуре современных скриптов (на примере библиотеки `python-telegram-bot` или `aiogram`) применяется концепция асинхронности (`async/await`)?
 4. Объекты обновлений и автоматические метаданные: Из каких обязательных полей состоит структура данных (JSON-объект `Update`), приходящая от серверов Telegram? Какие именно метаданные о пользователе бот извлекает автоматически после отправки команды `/start`?
 5. Диспетчеризация и фильтрация сообщений: Что такое «хэндлер» (`handler`) в контексте разработки бота и как устроена логика диспетчеризации входящих запросов? Какую задачу решают встроенные и кастомные фильтры сообщений?
 6. Проектирование пользовательского интерфейса: Чем типы клавиатур `ReplyKeyboardMarkup` концептуально и функционально отличаются от `InlineKeyboardMarkup`? Каким образом бот может динамически изменять (редактировать) текст или разметку уже отправленного сообщения, и как отправлять медиаконтент (документы, изображения)?
 7. Интерактивное взаимодействие через Callback: Для чего нужен параметр `callback_data` в Inline-кнопках, каковы ограничения на его объем (в байтах) и как сервер бота перехватывает и обрабатывает тип события `CallbackQuery`?
 8. Автоматы конечных состояний (FSM): В каких ситуациях при проектировании диалога необходимо использовать механизм `Finite State Machine` (машину состояний)? Опишите пошаговый алгоритм сохранения текущего состояния диалога (`State Machine`) при заполнении пользователем многоэтапных анкет или форм.
 9. Интеграция с СУБД SQLite: Каковы преимущества использования встроенной СУБД SQLite (модуль `sqlite3`) для локального хранения данных в небольших проектах? Опишите базовые SQL-запросы, необходимые для создания таблиц пользователей, а также записи и чтения логов общения из базы.
 10. Безопасное конфигурирование окружения: Зачем при интеграции программного кода с базами данных и внешними API использовать конфигурационные файлы `.env` вместо жесткого прописывания (хардкодинга) токенов и путей к базам прямо в теле скрипта?

Раздел 2. Тестирование программного обеспечения и отладка кода

1. Стандарты и цели тестирования: Каковы международные стандарты и отраслевые методологии тестирования программного обеспечения в сфере веб-сервисов и диалоговых систем? Какие задачи решает верификация кода перед его релизом?
2. Модульное тестирование (Unit Testing): В чем заключается сущность модульного тестирования кода чат-ботов? Как с помощью библиотек (`unittest` или `pytest`) изолированно проверить корректность работы отдельных функций (например, валидаторов ввода или парсеров) без реальной отправки запросов в Telegram?
3. Механизмы имитации зависимостей (Mocking): Каким образом концепция Mock-объектов позволяет симулировать ответы от серверов Telegram API или базы данных при проведении автоматизированных тестов логики бота?
4. Симуляция ошибок ввода данных: Опишите стратегию тестирования бота на устойчивость к некорректному пользовательскому вводу. Какие уязвимости в коде могут возникнуть, если хэндлеры не проводят предварительную валидацию типов данных (например, ожидая число, но получая строку)?

5. Тестирование отказоустойчивости СУБД: Как симулировать критический сбой в работе базы данных (блокировка файла SQLite, отсутствие нужной таблицы)? Какие архитектурные решения позволяют боту не завершать аварийно аварийный процесс (крашиться) при отказе локальной СУБД?
6. Симуляция сетевых сбоев: Каким образом тестируется поведение асинхронного скрипта при потере соединения с серверами Telegram (разрыв сетевого соединения, таймауты API)? Как настроить автоматические повторные попытки отправки запросов (retries)?
7. Перехват исключений и блоки try-except: Опишите правила проектирования безопасного асинхронного кода с использованием конструкций try-except-else-finally. Почему перехват базового класса Exception во всех хэндлерах считается плохой практикой (anti-pattern)?
8. Конфигурирование модуля логирования: Что такое логирование и как встроенный модуль logging в Python помогает при поиске скрытых багов? Как правильно разграничить уровни логирования (DEBUG, INFO, WARNING, ERROR, CRITICAL) в файлах чат-бота?
9. Инструменты интерактивной отладки: Каким образом специализированные инструменты (встроенный дебаггер pdb, точки останова breakpoints в IDE) используются для пошагового отслеживания значений переменных в асинхронных функциях во время симуляции багов?
10. Рефакторинг и оптимизация структуры кода: На основе каких критериев принимается решение о необходимости рефакторинга программного кода бота после успешного отлова ошибок? Как разделение монолитного скрипта на модули (хэндлеры, базы данных, клавиатуры, утилиты) повышает поддерживаемость кода?

Раздел 3. Версионность, развертывание и релиз-менеджмент

1. Анатомия структуры файлов проекта: Каковы профессиональные требования к структурированию файлов и каталогов в проектах по разработке ПО? Опишите назначение директорий для хранения модулей бота, конфигураций, миграций баз данных и статических файлов.
2. Управление зависимостями проекта: Какую задачу решает фиксация внешних библиотек в проекте? Как сгенерировать файл requirements.txt с помощью пакетного менеджера pip и почему важно указывать строгие версии устанавливаемых пакетов?
3. Изоляция окружения через Virtual Environments: Зачем перед началом развертывания проекта на целевом сервере создавать изолированное виртуальное окружение (venv)? К каким конфликтам может привести установка зависимостей бота в глобальную систему?
4. Инициализация локального репозитория Git: Опишите базовые команды системы контроля версий Git, необходимые для инициализации проекта, отслеживания изменений и формирования коммитов (git init, git add, git commit).
5. Конфигурирование исключений в Git: Какую роль выполняет файл .gitignore в корне проекта? Какие файлы и каталоги (база *.db, файлы .env, папки __pycache__ и .venv) строго запрещено отправлять в репозиторий и почему?
6. Релиз-менеджмент и тегирование версий: Что такое семантическое версионирование (Semantic Versioning) программного обеспечения? Как с помощью легковесных и аннотированных тегов в Git (git tag) зафиксировать стабильную версию (релиз) продукта?
7. Работа с удаленными репозиториями на GitHub: Каков алгоритм связывания локального Git-репозитория с удаленной платформой GitHub? Опишите процесс публикации (git push) задокументированного программного кода.
8. Документирование проекта (файлы README): Из каких обязательных разделов должен состоять файл README.md готового чат-бота на GitHub, чтобы сторонний разработчик смог успешно запустить и развернуть этот программный продукт?

9. Стратегии развертывания (VPS vs Домашний ПК): В чем заключаются ключевые преимущества и технологические приоритеты развертывания бота на арендованном виртуальном сервере (VPS/VDS) по сравнению с постоянным запуском скрипта на домашнем компьютере?
10. Обеспечение непрерывной работы (демонизация): Как с помощью системных менеджеров процессов (например, `systemd` в Linux) превратить скрипт чат-бота в системную службу (демона) для обеспечения фоновой работы, логирования вывода и автоматического перезапуска бота при случайном сбое сервера?

Критерии оценки работы на практическом занятии

Критерии	Максимальное количество баллов за занятие
Устный опрос, коллоквиум	
Основные теоретические положения по вопросу раскрыты. Имеются элементы обоснования выводов. Имеются элементы систематизации информации, факты применения профессиональной терминологии. Очевидно использование источников рекомендованной литературы.	5 баллов

4.2.2. Темы докладов

1. Сравнение библиотек `aiogram` и `python-telegram-bot`: архитектурные различия и критерии выбора инструмента разработки.
2. Асинхронность в Python (`asyncio`) на практике: почему это критически важно для высоконагруженных диалоговых систем.
3. Webhook vs Polling: глубокий технический разбор, настройка SSL-сертификатов и условия перехода на Webhook.
4. Использование архитектурного паттерна Middleware в Telegram-ботах: перехват и сквозная обработка входящих сообщений для сбора статистики и логирования.
5. Использование Docker для контейнеризации ботов: принципы упаковки проекта для обеспечения идентичности сред разработки и развертывания.
6. Тренды дизайна интерфейсов в Telegram: проектирование удобного меню с помощью Web Apps (Mini Apps) и превращение бота в полноценное приложение.
7. Машина состояний (FSM): проектирование сложных пользовательских сценариев, ветвлений диалогов и сохранения контекста.
8. Интернационализация (i18n): архитектура многоязычных ботов и методы автоматического определения локали пользователя.
9. Inline-режим в мессенджерах: проектирование и создание ботов, способных функционировать в пространстве любого чата.
10. Безопасное хранение данных в ботах: использование переменных окружения (`.env`), шифрование конфигураций и защита от утечки токенов.
11. Интеграция с SQL и NoSQL базами данных: сравнительный анализ применения SQLite, PostgreSQL и MongoDB в архитектуре чат-ботов.
12. Redis как распределенное хранилище для состояний FSM: обеспечение мгновенного отклика и сохранности контекста при перезагрузке сервера.
13. Обработка персональных данных в Telegram: юридические и технические аспекты соблюдения требований ФЗ-152 и GDPR.
14. Интеграция с внешними API: методология безопасного взаимодействия бота со сторонними сервисами (погода, платежные шлюзы, геокарты).

15. Платежные системы в Telegram: настройка приема оплаты за цифровые и физические товары через официальных провайдеров.
16. Автоматизация тестирования ботов: особенности написания Unit-тестов для асинхронного кода обработчиков (хэндлеров).
17. Логирование и мониторинг: использование систем типа Sentry или ELK-стека для отслеживания ошибок и исключений в реальном времени.
18. Искусственный интеллект в Telegram: как интеграция больших языковых моделей (LLM) меняет подход к разработке и архитектуре чат-ботов.

Шкала и критерии оценки доклада

Критерии	Показатели	Баллы
1. Степень раскрытия сущности проблемы	– соответствие теме доклада; – полнота и глубина раскрытия основных понятий; – умение работать с литературой, систематизировать и структурировать материал; – умение обобщать, сопоставлять различные точки зрения по рассматриваемому вопросу, аргументировать основные положения и выводы.	70
2. Обоснованность выбора источников	– круг, полнота использования литературных источников по теме; – привлечение новейших работ (журнальные публикации, материалы сборников научных трудов и т.д.).	15
3. Изложение	– литературный стиль.	15

Доклад оценивается по 100 балльной шкале, баллы переводятся в оценки успеваемости следующим образом:

90 – 100 баллов – «отлично»;

70 – 89 баллов – «хорошо»;

50 – 69 баллов – «удовлетворительно»;

менее 50 баллов – «неудовлетворительно».

4.3. Оценочные средства промежуточной аттестации

4.3.1. Контрольные задания

ПК-3.1.1-1. Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа.

Какой объект в Telegram Bot API содержит информацию о входящем событии (сообщении, нажатии кнопки и т.д.):

- А. Message
- Б. Update
- В. User
- Г. Dispatcher

--	--

ПК-3.1.1-2. Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа.

Какая функция в библиотеке aiogram (или аналогичных) используется для запуска процесса опроса серверов Telegram:

- А. bot.run()
- Б. bot.start()
- В. dp.start_polling()
- Г. executor.execute()

ПК-3.1.1-3. Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа.

Что такое FSM в контексте разработки ботов:

- А. Способ быстрой отправки сообщений
- Б. Машина состояний для ведения цепочки диалога
- В. Формат хранения данных в базе
- Г. Метод шифрования токена

ПК-3.1.1-4. Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа.

Какой префикс в коде указывает на то, что функция является асинхронной:

- А. await
- Б. sync
- В. defer
- Г. async

ПК-3.1.1-5. Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа.

Где в Telegram настраивается описание бота, его аватар и команды:

- А. В настройках личного профиля
- Б. В коде программы (через константы)
- В. В официальном боте @BotFather
- Г. Через службу поддержки Telegram

--

ПК-3.1.1-6. Прочитайте текст и установите последовательность.

Установите последовательность действий для реализации следующей задачи: создать Reply-клавиатуру с одной кнопкой «Помощь»:

- А. `ReplyKeyboardMarkup(`
- Б. `keyboard =`
- В. `resize_keyboard=True)`
- Г. `keyboard=[[KeyboardButton(text="Помощь")]] ,`
- Д. `InlineKeyboardButton(text="Помощь")`

Запишите соответствующую последовательность букв слева направо:

ПК-3.1.1-7. Прочитайте текст и установите последовательность.

Установите последовательность действий для реализации следующей задачи: Регистрация хэндлера, который реагирует на команду /start:

- А. `async def cmd_start(message: Message):`
- Б. `@dp.message(`
- В. `Command("start")`
- Г. `)`
- Д. `F.text == "/start"`

Запишите соответствующую последовательность букв слева направо:

ПК-3.1.2-1. Прочитайте текст и установите последовательность.

Установите последовательность действий для реализации следующей задачи: Отправка текстового ответа пользователю на его сообщение:

- А. `message.answer(`
- Б. `"Привет!"`
- В. `await`
- Г. `)`

Запишите соответствующую последовательность букв слева направо:

ПК-3.1.2-2. Прочитайте текст и установите последовательность.

Установите последовательность действий для реализации следующей задачи: Получение значения токена из переменной окружения:

- А. `os.getenv(`
- Б. `"BOT_TOKEN"`
- В. `token =`
- Г. `)`

Запишите соответствующую последовательность букв слева направо:

ПК-3.1.2-3. Прочитайте текст и установите последовательность.

Установите последовательность действий для реализации следующей задачи: Переключение пользователя в состояние «ожидания имени» (FSM):

- А. state.set_state(
- Б. await
- В. Form.name
- Г.)

Запишите соответствующую последовательность букв слева направо:

ПК-3.1.2-4. Прочитайте текст и установите соответствие.

Сопоставьте понятия (буквы) и сущности (цифры):

А. Токен бота	1. Управление всеми входящими обновлениями и фильтрами
Б. Диспетчер (Dispatcher)	2. Уникальная строка вида 12345:ААН...
В. Обработчик (Handler)	3. Отслеживание ошибок и работы программы в консоли
Г. Логирование (Logging)	4. Функция, реагирующая на конкретное событие

Запишите последовательность букв и соответствующим им цифр слева направо:

ПК-3.1.2-5. Прочитайте текст и установите соответствие.

Сопоставьте понятия (буквы) и сущности (цифры):

А. answer_photo	1. Отправка PDF-файла или архива
Б. answer_document	2. Изменение содержания уже отправленного сообщения
В. edit_text	3. Отправка координат на карте
Г. answer_location	4. Отправка изображения (jpg/png)

Запишите последовательность букв и соответствующим им цифр слева направо:

ПК-3.1.2-6. Прочитайте текст и установите соответствие.

Сопоставьте понятия (буквы) и сущности (цифры):

А. Command("settings")	1. Реакция на конкретную строковую константу
Б. F.text == "Да"	2. Ограничение доступа только для администратора
В. F.photo	3. Реакция на команду /settings
Г. F.from_user.id == ADMIN_ID	4. Реакция на присланную картинку

Запишите последовательность букв и соответствующим им цифр слева направо:

ПК-3.1.2-7. Прочитайте текст и установите соответствие.

Сопоставьте понятия (буквы) и сущности (цифры):

А. url="https://..."	1. Параметр Reply-кнопки для скрытия клавиатуры после нажатия
Б. callback_data="btn1"	2. Параметр Inline-кнопки для перехода на сайт
В. request_contact=True	3. Параметр Inline-кнопки для обработки нажатия сервером
Г. one_time_keyboard=True	4. Параметр Reply-кнопки для запроса номера телефона

Запишите последовательность букв и соответствующим им цифр слева направо:

ПК-3.1.2-8. Прочитайте текст и установите соответствие.

Сопоставьте понятия (буквы) и сущности (цифры):

А. async def	1. Точка входа для запуска асинхронного приложения
Б. await	2. Ожидание завершения выполнения асинхронной задачи
В. asyncio.run()	3. Цикл событий, управляющий задачами
Г. loop	4. Объявление асинхронной функции (корутины)

Запишите последовательность букв и соответствующим им цифр слева направо:

4.3.2. Ключи к контрольным заданиям

ПК-3.1.1-1	2 Update – это контейнер для информации о действиях пользователя
ПК-3.1.1-2	3 Это функция для постоянного опроса сервера
ПК-3.1.1-3	2 FSM(Finite State Machine) позволяет следить за диалогом бота с пользователем
ПК-3.1.1-4	4 стандартное обозначение асинхронной функции в Python
ПК-3.1.1-5	3 стандартный бот для задания характеристик проектируемого бота
ПК-3.1.1-6	БАГВ
ПК-3.1.1-7	БВГА
ПК-3.1.2-1	ВАБГ
ПК-3.1.2-2	ВАБГ
ПК-3.1.2-3	БАВГ
ПК-3.1.2-4	A2B1B4Г3

ПК-3.1.2-5	A4B1B2Г3
ПК-3.1.2-6	A3B1B4Г2
ПК-3.1.2-7	A2B3B4Г1
ПК-3.1.2-8	A4B2B1Г3

Шкала и критерии оценки текущего тестирования

Число правильных ответов	Оценка
90-100% правильных ответов	Оценка «отлично»
70-89% правильных ответов	Оценка «хорошо»
50-69% правильных ответов	Оценка «удовлетворительно»
Менее 50% правильных ответов	Оценка «неудовлетворительно»

5. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ В ОТНОШЕНИИ ЛИЦ ИЗ ЧИСЛА ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Профессорско-преподавательский состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Министерства науки и высшего образования Российской Федерации, в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создания комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учётом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Согласно требованиям к порядку реализации образовательной деятельности в отношении инвалидов и лиц с ОВЗ, установленным Министерством науки и высшего образования Российской Федерации, необходимо иметь в виду, что:

1) инвалиды и лица с ОВЗ по зрению имеют право присутствовать на занятиях вместе с ассистентом, оказывающим обучающемуся необходимую помощь;

2) инвалиды и лица с ОВЗ по слуху имеют право на использование звукоусиливающей аппаратуры.

При проведении промежуточной аттестации по дисциплине обеспечивается соблюдение следующих общих требований:

- проведение аттестации для инвалидов в одной аудитории совместно с обучающимися, не являющимися инвалидами, если это не создает трудностей для инвалидов и иных обучающихся при промежуточной аттестации;

- присутствие в аудитории ассистента (ассистентов), оказывающего обучающимся инвалидам необходимую техническую помощь с учётом их индивидуальных особенностей (занять рабочее место, передвигаться, прочитать и оформить задание, общаться с экзаменатором);

- пользование необходимыми обучающимся инвалидам техническими средствами при прохождении промежуточной аттестации с учётом их индивидуальных особенностей;

- обеспечение возможности беспрепятственного доступа обучающихся инвалидов в аудитории, туалетные и другие помещения, а также их пребывания в указанных помещениях.

По письменному заявлению обучающегося инвалида продолжительность прохождения испытания промежуточной аттестации (зачёта, экзамена, и др.) обучающимся инвалидом может быть увеличена по отношению к установленной продолжительности его сдачи:

- продолжительность сдачи испытания, проводимого в письменной форме, — не более чем на 90 минут;

- продолжительность подготовки обучающегося к ответу, проводимому в устной форме, — не более чем на 20 минут.

В зависимости от индивидуальных особенностей обучающихся с ОВЗ Университет обеспечивает выполнение следующих требований при проведении аттестации:

а) для слепых:

- задания и иные материалы для прохождения промежуточной аттестации оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением для слепых, либо зачитываются ассистентом;

- письменные задания выполняются обучающимися на бумаге рельефно-точечным шрифтом Брайля или на компьютере со специализированным программным обеспечением для слепых, либо надиктовываются ассистенту;

- при необходимости обучающимся предоставляется комплект письменных принадлежностей и бумага для письма рельефно-точечным шрифтом Брайля, компьютер со специализированным программным обеспечением для слепых;

б) для слабовидящих:

- задания и иные материалы для сдачи экзамена оформляются увеличенным шрифтом;

- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;

- при необходимости обучающимся предоставляется увеличивающее устройство, допускается использование увеличивающих устройств, имеющихся у обучающихся;

в) для глухих и слабослышащих, с тяжелыми нарушениями речи:

- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающимся предоставляется звукоусиливающая аппаратура индивидуального пользования;

- по их желанию испытания проводятся в письменной форме;

г) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются обучающимися на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по их желанию испытания проводятся в устной форме.

О необходимости обеспечения специальных условий для проведения аттестации обучающийся должен сообщить письменно не позднее, чем за 10 дней до начала аттестации. К заявлению прилагаются документы, подтверждающие наличие у обучающегося индивидуальных особенностей (при отсутствии указанных документов в организации).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.