

**КАЗАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ (ФИЛИАЛ)
АВТОНОМНОЙ НЕКОММЕРЧЕСКОЙ ОБРАЗОВАТЕЛЬНОЙ
ОРГАНИЗАЦИИ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

КОМПЬЮТЕРНЫЕ СЕТИ

Специальность: 10.05.01 Компьютерная безопасность

Специализация: «Разработка защищенного программного обеспечения»

Формы обучения: очная

Объем дисциплины:

в зачетных единицах: 2 з.е.

в академических часах: 72 ак.ч.

Форма промежуточной аттестации: зачет

Рабочая программа по дисциплине «Компьютерные сети» по специальности 10.05.01 Компьютерная безопасность, специализация «Разработка защищенного программного обеспечения», составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования – специалитет 10.05.01 Компьютерная безопасность, утвержденного приказом Министерства науки и высшего образования Российской Федерации от «26» ноября 2020 г. № 1459, профессионального стандарта 06.032 «Специалист по безопасности компьютерных систем и сетей», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. № 533н, профессионального стандарта 06.033 «Специалист по защите информации в автоматизированных системах», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. №525н.

Рабочая программа:

обсуждена и рекомендована к утверждению решением кафедры естественных дисциплин, сервиса и туризма от 25 марта 2025 г., протокол № 6

одобрена Научно-методическим советом Казанского кооперативного института (филиала) от 2 апреля 2025 г., протокол № 3

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы	4
3. Перечень планируемых результатов обучения по дисциплине	5
4. Объем дисциплины и виды учебной работы.....	6
5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий.....	7
5.1. Содержание дисциплины	7
5.2. Разделы, темы дисциплины и виды занятий	8
6. Практические занятия	9
7.Лабораторные работы	10
8. Примерная тематика курсовых работ (проектов).....	10
9. Самостоятельная работа студента	10
10. Оценивание результатов обучения и уровня сформированности компетенций	16
11. Учебно-методическое обеспечение дисциплины.....	19
12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины.....	20
13. Материально–техническое обеспечение дисциплины	19
Обновление рабочей программы дисциплины (модуля)	20

1. Цель и задачи освоения дисциплины

Целью данной дисциплины является определение сути, содержания и практической необходимости современных сетевых технологий. Особое внимание уделяется низкоуровневому сетевому программированию клиент-серверных взаимодействий, анализу средств обеспечения безопасности информационных ресурсов информационной системы.

Разработка в рамках курса реальных проектов формирует у слушателей способность применять в профессиональной деятельности сетевые технологии, современные языки программирования, операционные системы, электронные библиотеки и пакеты программ.

Приобретенные профессиональные навыки работы с информационными и компьютерными технологиями можно эффективно использовать в научной и профессиональной практической деятельности, а также в социальной сфере.

Воспитательной целью дисциплины является формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения.

Задачи:

- изучение теоретических основ построения и функционирования компьютерных сетей;
- ознакомление со стандартными моделями взаимодействия процессов в распределенном приложении;
- изучение протоколов TCP/IP и основных принципов взаимодействия его компонентов;
- освоение интерфейса Windows Socket API, как основы для построения распределенных приложений в среде TCP/I;
- освоение элементов параллельного программирования, необходимых при разработке сетевых приложений в режиме множественного доступа;
- приобретение навыков практической разработки сетевых приложений..

2. Место дисциплины в структуре образовательной программы

Дисциплина «Компьютерные сети» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» основной образовательной программы – программы специалитета по специальности 10.05.01 Компьютерная безопасность, специализация «Разработка защищенного программного обеспечения».

Код компетенции	Предшествующие дисциплины (модули), практики	Изучаемые в текущем семестре дисциплины (модули), практики	Последующие дисциплины (модули), практики
-----------------	--	--	---

ОПК-15	-	Компьютерные сети	Сети и системы передачи информации Производственная практика, преддипломная практика
--------	---	--------------------------	--

3. Перечень планируемых результатов обучения по дисциплине

Изучение дисциплины направлено на формирование у обучающихся профессиональных компетенций.

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
ОПК-15 Способен администрировать компьютерные сети контролировать корректность их функционирования	ОПК-15.1. Применяет знания моделей взаимодействия открытых систем, технологий коммутации, маршрутизации, межсетевого экранирования, виртуализации при решении задач профессиональной деятельности	Знать: Классификация сетей по: топологии, типу управления сетью; территориальному признаку; функциональному назначению. Методы коммутации: каналов, сообщений, пакетов. Модель взаимодействия открытых систем OSI, клиент-серверная архитектура сетевых приложений. Основные стеки сетевых протоколов. Типы сетевых соединений; Уметь: Проводить анализ моделей взаимодействия открытых систем. Выбирать степень открытости спецификаций системы. Использовать оптимальные технологии коммутации, учитывать вопросы безопасности при разработке инфраструктуры сети. Выбирать необходимый тип соединения: передача данных без установления соединения (дейтаграммное соединение); передача с установлением соединения (потокосое соединение). Понимать модель сокетов, принцип действия базовых функций библиотеки Winsock. Новая редакция библиотеки. WWW-сервис. Прикладной сетевой протокол http. Форматы http-сообщений. Процессы и потоки. Параллельные технологии в сетевой индустрии. Владеть: Владеть навыками разработки серверных и клиентских компонент системных и прикладных сетевых проектов клиент-серверной архитектуры на основе библиотеки Winsock. Анализировать возможности применяемых стеков сетевых протоколов. Применять механизмы синхронизации потоков (критические секции, мьютексы, события, семафоры) при разработке сетевых приложений.

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
		Моделировать и реализовать публичный и приватный варианты chat-приложения.
	ОПК-15.2. Имеет практический опыт администрирования сетевых клиентских и серверных компонентов операционных систем, а также проектирования корпоративной сетевой инфраструктуры	Знать: Виды клиент-серверной архитектуры сетевых проектов. Понимать парадигму сокетов. Основные функции библиотеки Winsock. Причины модификации функций библиотеки Winsock. Основные принципы разработки Web- приложения. Механизмы синхронизации параллельных потоков. Механизмы разработки многопользовательского сервера. Основные задачи администрирования сетей. Уметь: Выбирать и использовать оптимальные виды сетевых соединений и протоколы при проектировании сетевого программного обеспечения. Учитывать вопросы безопасности при разработке инфраструктуры корпоративной сети. Новая редакция базовых функций Winsock. Применять механизмы параллельных вычислений при создании сетевых приложений. назначение сетевой службы, разработать структуру сервиса. Владеть: Владеть навыками низкоуровневой разработки серверных и клиентских компонент системных и прикладных сетевых проектов на основе библиотеки сокетов. Проводить качественный анализ механизмов коммутации сетевых каналов. Владеть навыками разработки клиент-серверной архитектуры проектов. Применять системный подход при планировании работ с рисками в соответствии с полученным заданием.

4. Объем дисциплины и виды учебной работы

Объем дисциплины и виды учебной работы в академических часах с выделением объема контактной работы обучающихся с преподавателем и самостоятельной работы обучающихся

очная форма обучения

Вид учебной деятельности	ак. часов	
	Всего	По семестрам
		4
1. Контактная работа обучающихся с преподавателем:	35	35

Аудиторные занятия, часов всего, в том числе:		34	34
• занятия лекционного типа		18	18
• занятия семинарского типа:		16	16
практические занятия		16	16
лабораторные занятия		-	-
в том числе занятия в форме практической подготовки		-	-
Консультации		0,5	0,5
Контактные часы на аттестацию в период экзаменационных сессий		0,5	0,5
2. Самостоятельная работа студентов всего, в том числе		37	37
- подготовка курсовой работы		-	-
-проработка учебного (теоретического) материала		10	10
-выполнение индивидуальных заданий (подготовка сообщений, презентаций)		20	20
- реферат		-	-
- подготовка к зачету		5,8	5,8
Промежуточная аттестация: зачет		-	-
ИТОГО: общая трудоемкость	часов	72	72
	зач. ед.	2	2

5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Теоретические основы сетевых технологий

Компьютерная сеть как распределенная вычислительная система. Классификация сетей по: топологии, типу управления сетью; территориальному признаку; функциональному назначению.

Методы коммутации: каналов, сообщений, пакетов.

Тема 2. Структура распределенных приложений

Понятие клиента, понятие сервера. Многозвенная клиент-серверная архитектура. Модель взаимодействия открытых систем OSI. Понятие сетевого протокола, сетевого интерфейса. Стеки сетевых протоколов.

Тема 3. Способы передачи данных в сети

Передача данных без установления соединения (дейтаграммное соединение); передача с установлением соединения (потокосое соединение). Используемые протоколы.

Тема 4. Интерфейс сокетов

Модель сокетов. Библиотека функций Winsock. Инициализация Winsock. Создание сокета. Закрытие сокета. Конфигурация сокета. Сетевой формат данных. Привязка адреса к сокету. Соединение сокета. Серверные сокеты. Передача данных.

Тема 5. Современные надстройки WinSock.

Причины модификации функций библиотеки Winsock. Функции getaddrinfo и freeaddrinfo. Функции inet_pton и inet_ntop. Новая редакция базовых функций Winsock.

Тема 6. HTTP-взаимодействия

WWW-сервис. Прикладной сетевой протокол http. Форматы http-сообщений. Использование http как транспортного средства. Принципы разработки Web-приложения.

Тема 7. Многопоточность

Процессы и потоки. Несинхронизированные потоки. Механизмы синхронизации потоков: критические секции, мьютексы, события, семафоры.

Тема 8. Разработка параллельных серверов

Использование элементов параллельного программирования для реализации многопользовательского TCP-сервера. Использование потоков в модуле клиента для реализации асинхронности операций приема и передачи данных

Тема 9. Проектирование сетевых служб

Структура и назначение сетевой службы.

Моделирование chat-приложения. Разработка публичного и приватного вариантов сервиса.

5.2. Разделы, темы дисциплины и виды занятий

очная форма обучения

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки (лабораторные работы)	самостоятельная работа	
1	Тема 1. Теоретические основы сетевых технологий	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
2	Тема 2. Структура распределенных приложений	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки (лабораторные работы)	самос- тоятельная работа	
3	Тема 3. Способы передачи данных в сети	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
4	Тема 4. Интерфейс сокетов	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
5	Тема 5. Современные настройки WinSock.	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
6	Тема 6. HTTP- взаимодействия	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
7	Тема 7. Многопоточность	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
8	Тема 8. Разработка параллельных серверов	2	2	4	ОПК-15 ОПК-15.1 ОПК-15.2
9	Тема 9. Проектирование сетевых служб	2	-	5	ОПК-15 ОПК-15.1 ОПК-15.2
	Всего	18	16	37	

6. Практические занятия

№ п/п	Наименование раздела, темы дисциплины	Содержание практических работ	Объем (час.)
			Очная форма обучения
1.	Тема1. Теоретические основы сетевых технологий	Компьютерная сеть как распределенная вычислительная система. Классификация сетей по: топологии, типу управления сетью; территориальному признаку; функциональному назначению. Методы коммутации: каналов, сообщений, пакетов.	2/2

№ п/п	Наименование раздела, темы дисциплины	Содержание практических работ	Объем (час.)
			Очная форма обучения
2.	Тема 2. Структура распределенных приложений	Понятие сервера и клиента. Клиент-серверная архитектура. Модель взаимодействия открытых систем OSI. Понятие сетевого протокола, сетевого интерфейса .Стеки сетевых протоколов.	2/2
3.	Тема 3 Способы передачи данных в сети	Дейтаграммные взаимодействия. Взаимодействия с установлением устойчивого соединения. Протоколы транспортного уровня TCP и UDP.	2/2
4.	Тема 4. Интерфейс сокетов	Исследование модели сокетов. Изучение библиотечных функций Winsock для реализации процесса сетевого взаимодействия на основе интерфейса сокетов. Определение сервисных, клиентских функций, функций дейтаграммного взаимодействия и при потоковом соединении. Примеры сетевых взаимодействий на основе socket-интерфейса для различных способов соединения.	2/2
5.	Тема5. Современные надстройки WinSock.	Исследование необходимости модификации функций библиотеки Winsock. Назначение новых функций getaddrinfo и freeaddrinfo, функции inet_pton и inet_ntop. Нюансы применения базовых функций Winsock в современных средах программирования.	2/2
6.	Тема 6. Http-взаимодействия	Структура http-форматов. Использование http как транспортного средства Использование в Web-приложении стандартных модулей клиентов и серверов. Программирование на C++ при использовании интерфейса сокетов пользовательских клиентов/серверов	2/2
7.	Тема 7. Многопоточность	Разработка несинхронизированных многопоточных сетевых приложений. Реализация и анализ различных механизмов синхронизации потоков (критические секции, мьютексы , события, семафоры).	2/2
8.	Тема 8. Разработка параллельных серверов. Чат.	Реализация многопользовательского режима при поточных соединениях на основе многопоточности. Моделирование chat-приложения. Разработка публичного и приватного вариантов сервиса.	2/2
	Всего		16

7.Лабораторные работы

Практические занятия не предусмотрены.

8. Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) не предусмотрены.

9. Самостоятельная работа студента

Самостоятельная работа студента при изучении дисциплины «Компьютерные сети» направлена на:

- освоение нормативных правовых актов и учебной литературы, необходимых для освоения дисциплины;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашних заданий по практическим занятиям;
- подготовку к зачету.

Краткие рекомендации по выполнению самостоятельной работы:

Успешное усвоение дисциплины предполагает большой, упорный, серьезный, систематический труд студентов. Важнейшая его составная часть – выполнение разных видов самостоятельной работы.

1. **Составление тематического конспекта** на основе изученной основной и дополнительной учебной литературы. В тематическом конспекте за основу берется содержание темы, вопросы для обсуждения.

Этапы работы.

1.1 Конспектирование делается только после того, как прочитан или усвоен материал для конспектирования.

1.2. Необходимо мысленно или письменно составить план конспекта. По этому плану и будет строиться конспект далее.

1.3. Составление самого конспекта. Можно сказать, что конспект – это расширенные тезисы, дополненные рассуждениями и доказательствами, содержащимися в материалах для конспекта, а также собственными мыслями и положениями составителя конспекта.

Писать конспект рекомендуется четко и разборчиво. В конспекте можно выделять места текста в зависимости от их значимости. Для этого применяются различного размера буквы, подчеркивания, замечания на полях.

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет) Вопросы для подготовки к зачету:

1. Понятие компьютерной сети.
2. Эволюция вычислительных систем.
3. Классификация компьютерных сетей.
4. Топология сетей.
5. Системы передачи данных. Основные характеристики.
6. Понятие канала.
7. Типы коммутации каналов.
8. Аналоговые и дискретные каналы.
9. Цифровая и аналоговая модуляция

10. Беспроводные каналы связи. Спутниковый канал.
 11. Достоверность передачи данных. Помехоустойчивое кодирование.
 12. Метод Хэмминга.
 13. Модели сетевых взаимодействий.
 14. Понятие открытой системы. Примеры.
 15. Понятие сетевого протокола и сетевого интерфейса.
 16. Примеры сетевых протоколов. Их функционал. Стеки сетевых протоколов.
 17. Методы разделения общей среды (Ethernet, Token ring, ArcNet).
 18. Сетевой адрес. Используемые форматы.
 19. Алгоритмы маршрутизации.
 20. Назначение алгоритмов безадресной маршрутизации.
 21. Фиксированная маршрутизация.
 22. Адаптивная маршрутизация.
 23. Протокол IP.
 24. IP-адресация. Классы адресов. Маски адресов.
 25. Сервера ARP, DHCP, DNS.
 26. Протокол HTTP.
 27. Клиент-серверная модель.
 28. Создание клиент-серверного приложения на основе файлов.
 29. Распределенное сетевое приложение.
 30. Понятие порта, сокета. Стандартные положения.
 31. Транспортные протоколы UDP и TCP.
 32. Алгоритм построения сетевого соединения на основе технологии сокетов.
 33. Создание клиентского модуля.
 34. Создание серверного модуля.
 35. Способы передачи данных разных типов в приложениях C++.
 36. Организация сетевого взаимодействия по протоколу HTTP.
 37. Необходимость использования параллельных потоков. Примеры.
- Средства реализации.
38. Понятие сетевого экрана.
 39. Средства интегрирования сетей. Физическое и логическое структурирование.
 40. Повторитель, мост, маршрутизатор, шлюз.
 41. Безопасность компьютерных сетей. Основные модели.
 42. Аутентификация и авторизация.
 43. Вид сервиса CHAT. Публичный и приватный варианты.
 44. Механизмы, используемые при написании приложения CHAT на C++.

Примеры практических сетевых проектов

Для выполнения практических заданий необходимо иметь несколько

объединенных TCP/IP-сетью компьютеров с операционной системой Windows x.

Кроме того, для разработки приложений на языке C++ требуется среда разработки Microsoft Visual Studio.

Приложение в проекте разрабатывается как распределенное клиент-серверное.

Проект №1. Клиент-серверное приложение.

Общая среда – файл

Сценарий 1 (один клиент, один сервер)

Приложение составить из двух программ: клиент и сервер. Для взаимодействия этих программ использовать 2 файла. Файл f1 заполняется клиентом (клиент пишет в него запросы серверу), сервер считывает из него очередной запрос клиента. Файл f2 заполняется сервером (сервер пишет в него свой ответ на запрос клиента), клиент считывает из него ответ на свой запрос.

Взаимодействие клиента и сервера происходит по следующему алгоритму:

Клиент дописывает свой запрос в конец файла f1. Сервер в бесконечном цикле проверяет, появились ли в файле f1 новые запросы. Для этого он сравнивает предыдущий размер файла с текущим (в начале работы предыдущий и текущий размеры файлов равны 0!).

Если эти размеры совпали – нет новых запросов. Иначе – появился новый запрос, в этом случае сервер считывает его из файла f1, обрабатывает и результат записывает в конец файла f2.

Клиент аналогично постоянно проверяет файл f2 на наличие новых ответов от сервера.

Реализовать данный сценарий для случаев:

- файлы текстовые;
- файлы бинарные (передача структур);

Сценарий 2 (для большого количества клиентов)

Приложение состоит из двух отдельных модулей (программ на C++): клиент и сервер.

Для общения используются файл con (создается сервером, используется сервером и всеми клиентами) и для каждого клиента файл, имя которого совпадает с именем клиента (каждый такой файл создается клиентом, используется сервером и клиентом, имя которого совпадает с именем файла). Итого: количество файлов равно $1 + m$, где m – количество активных клиентов. Получив от клиента его имя, сервер определяет файл для общения с этим клиентом.

Клиент и сервер заранее согласовывают тип файлов и формат сообщений. Возможные варианты предметных областей для разрабатываемого проекта:

1. Сервер – медицинский центр. Клиент передает серверу фамилию

студента, его рост и вес. Сервер на основании этих данных выдает результат о нормальности развития студента (нормально, превышение веса, нехватка веса).

2. Сервер – деканат. Клиент передает серверу фамилию студента и четыре оценки по экзаменам сессии. Сервер на основании этих данных выдает результат, есть ли у студента задолженности; если нет, то возможно ли получение стипендии и ее размер.

Проект №2. Интерфейс сокетов. TCP и UDP сокет

Целью данного проекта является выявление особенностей и реализация сетевых взаимодействий двух видов: потоковое соединение и дейтаграммное. При разработке конструкций приложений использование параллельного программирования не предусматривается. Особое внимание уделить рассмотрению общения сервера с клиентами в каждом случае взаимодействия.

1. Реализовать модельные сетевые приложения:

а) диалоговое общение на основе TCP-сокетов; б) эхо-сервер на основе UDP-сокетов.

Объяснить использование необходимых функций интерфейса WinSocket.

2. Предусмотреть:

а) использование в приложениях произвольного количества клиентов (1 или более);

б) распределенный характер приложения, т. е. выполнение клиентов и сервера на различных компьютерах сети, указать необходимые изменения в кодах приложений.

3. Изменить код приложений таким образом, чтобы возможно было передавать от клиента к серверу и обратно данные произвольных типов (структуру). Можно реализовать сценарий, используемый в проекте 1 при создании клиент-серверных приложений на основе бинарных файлов.

Проект №3. HTTP-взаимодействия

Для выполнения проекта необходимо изучить основные положения протокола HTTP (форматы запросов клиента и ответа сервера) и выполнить следующие действия:

1. Написать на C++ клиент серверное приложение осуществляющее взаимодействие по протоколу HTTP на основе TCP-сокетов;

2. Используя HTTP-клиент, написанный на C++, обратиться методом GET к одному из стандартных Web-серверов (например, json.org или library.ru).

3. Используя стандартный браузер, обратиться к HTTP-серверу, написанному на C++.

4. С помощью программы telnet отправить HTTP-запрос к веб-серверу. Запрос к HTTP-серверу вначале записать в текстовый файл. Для соединения с веб-сервером с использованием telnet нужно выполнить команду: telnet <ip-

адрес-веб-сервера > 80 и далее через буфер обмена мышью скопировать запрос из текстового файла, при необходимости нажать Enter два раза. В новых версиях Windows по умолчанию telnet не устанавливается и его нужно установить отдельно через установщик компонентов Windows.

Результаты работы представить в виде скринов.

Проект №4. Многопоточность. Средства синхронизации

Организовать параллельную работу сервера с клиентами на основе ТСР-соединения.

1. Создание многопоточного эхо-сервера.
2. Передача структур данных между клиентом и сервером.

По возможности реализовать распределенное клиент-серверное приложение (на разных компьютерах).

Коды клиента и сервера представить в виде файлов *.cpp.

3. Изучить механизмы синхронизации данных, разделяемых потоками (критические секции, мьютексы, события, семафоры).

Для каждого механизма написать программу, иллюстрирующую синхронизацию данных.

Работоспособность программ подтвердить при помощи снимков экрана (скриншоты).

Проект №5. Создание многопоточного клиент-серверного сетевого приложения чат(chat) с использованием технологии WinSocket

Разработать клиент-серверное приложение для реализации сетевого сервиса. Для реализации многопользовательского режима использовать элементы параллельного программирования. В качестве модельной задачи можно рассмотреть варианты чат-сервиса:

1. Создание публичного чата (переданная в чат информация мгновенно передается всем входящим в чат участникам).

2. Разработать возможность указания ника (прозвища) для участников чата. При подключении участника сообщать всем остальным членам чата, что подключился участник с определенным ником. При уходе участника также сообщать о его уходе с указанием его ника. При отправке сервером сообщения клиентам отправлять и ник участника, который это сообщение выложил в чат. Для этого, возможно, придется расширить количество значений перечислимого типа PACKET, задающего тип пакета, и разработать формат пакета для каждого типа.

3. Включить в приложение чат возможность приватного общения, когда при передаче сообщения участник указывает серверу ник участника, которому это сообщение нужно передать. Остальные участники, естественно, данное сообщение не получают.

Проверить работу приложения, используя интерфейс обратной петли, а также в сетевом варианте, запустив процессы сервера и клиентов на различных компьютерах.

10. Оценивание результатов обучения и уровня сформированности компетенций

Оценивание результатов обучения по дисциплине и уровня сформированности компетенций (части компетенции) осуществляется в рамках текущего контроля успеваемости и промежуточной аттестации в соответствии оценочным материалам.

11. Учебно-методическое обеспечение дисциплины

Основная литература:

1. Пятибратов А.П., Гудыко Л.П., Кириченко А.А. Вычислительные системы, сети и коммуникации. Учебное пособие. URL:<https://books.google.ru/>
2. Компьютерные сети. Мноуровневая архитектура Интернета URL:<https://conlex.kz/>
3. Компьютерные сети от А до Я: классификация, стандарты и уровни. Библиотека программиста. URL: <https://proglib.io/p/computer-network/>
4. Касперский К. Секреты среды разработки Visual Studio/URL: <https://citforum.ru/book/cook/studio.shtml>.
5. Низкоуровневое программирование. URL: <https://lecturesnet.readthedocs.io/>

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечных системах «Лань» и «Юрайт».

Дополнительная литература:

1. Букатов А.А., Гуда С.А. Компьютерные сети. Расширенный начальный курс. СПб., 2019.
2. Васильев А. Программирование на C++ в примерах и задачах. М., 2017.
3. Власов Ю. В., Рицкова Т.И. Администрирование сетей на платформе MS Windows Server: учеб. пособие. М., 2010.
4. Гельбух С. Сети ЭВМ и телекоммуникации. Архитектура и организация: учеб. пособие. СПб., 2019.
5. Дубовик С. C++. Сетевое программирование. URL: <https://www.vr-online.ru/content/s-setevoe-programmirovanie-chast-1-3999>
6. Каев А. Программирование сети в VC++. URL: <https://http://www.firststeps.ru/mfc/net/socket/>
7. Касперски К. Секреты поваров компьютерной кухни или ПК: решение проблем. СПб., 2003.

8. Кенин А., Колисниченко Д. Самоучитель системного администратора. 5-е изд. СПб., 2019.
9. Кумар В., Кровчик Э., Лагари Н. Net. Сетевое программирование. М., 2014.
10. Мэрфи Н., Мэлоун Д. IPv6. Администрирование сетей. СПб., 2007.
11. Неволин А.О. Базовые принципы сетевого взаимодействия: учеб. пособие для вузов. М., 2020.
12. Ногл М. Иллюстрированный учебник. М., 2001.
13. Оланд Д., Джонс Э. Программирование в сетях Windows. СПб., 2002.
14. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. СПб., 2020.
15. Побегайло А. П. Системное программирование в Windows. СПб., 2005.
16. Снейдер Й. Эффективное программирование TCP/IP. Создание сетевых приложений. М., 2019.
17. Стивен П. Язык программирования C++. Лекции и упражнения. М., 2018.
18. Стивенс У. Р. Протоколы TCP/IP: практическое руководство. СПб., 2003.
19. Таненбаум Э., Уэзеролл Д. Компьютерные сети. СПб., 2019.
20. Шмидт Д., Хьюстон С. Программирование сетевых приложений на C++. М., 2010.
21. Winsock для всех. Клуб программистов «Весельчак.У». URL: <https://club.shelek.ru/>
22. Russell J. Winsock (на англ. яз.). М., 2012.
23. William T. Block Introduction to Multi-threaded Code, 26 Mar 2000. URL: <https://www.codeproject.com/Articles/438>

Электронные библиотечные системы (ЭБС) и электронные образовательные ресурсы

1. Электронно-библиотечная система ZNANIUM <https://www.znanium.com>
2. Универсальная справочно-информационная полнотекстовая база данных периодических изданий «East View» (ИВИС) <https://dlib.eastview.com/>
3. Электронно-библиотечная система «BOOK.ru» www.book.ru
4. Электронно-библиотечная система «IPRbooks» <http://www.iprbookshop.ru/>
5. ЭБС «Национальный цифровой ресурс «Руконт» <https://lib.rucont.ru/search>

6. Образовательная платформа ЮРАЙТ <http://www.ura.it.ru>

Электронный каталог Университета – Автоматизированная информационная библиотечная система (АИБС «МегаПро»)

12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины

1. Лицензионное программное обеспечение, в том числе отечественного производства:
 - DsktpEdu ALNG LicSARk OLV F1 1Y Acdmc Ent (MS Windows) (подписка на ПО).
 - Программная система для обнаружения текстовых заимствований в учебных и научных работах «Антиплагиат.ВУЗ» (отечественное ПО).
 - СПС КонсультантПлюс.
 - 1С: Предприятие 8.3
 - ФинЭкАнализ (профессиональная автоматизированная система комплексного финансово-экономического и управленческого анализа хозяйственной деятельности предприятия)
1. Свободно распространяемое программное обеспечение, в том числе отечественного производства:
 - 7-Zip (свободный файловый архиватор с высокой степенью сжатия данных) (отечественное ПО).
 - FastStone Image Viewer (программа для просмотра изображений в Microsoft Windows (лицензия бесплатного программного обеспечения)).
 - Foxit Reader (Бесплатное прикладное программное обеспечение для просмотра электронных документов в стандарте PDF (лицензия бесплатного программного обеспечения)).
 - MyTest (система программ для создания и проведения компьютерного тестирования знаний).
 - Google Chrome (браузер).
 - Mozilla Firefox (браузер).
 - Adobe Acrobat Reader DC (• ПО для просмотра, печати и комментирования документов в формате PDF)
 - Visual Studio Code (интегрированная среда разработки программного обеспечения)
 - NVDA (социально-информационный проект для людей с ограниченными возможностями по зрению, использующих бесплатную программу экранного доступа)
1. Профессиональные базы данных не используются.
2. Информационные справочные системы:
 - СПС КонсультантПлюс. Компьютерная справочная правовая система, широко используется учеными, студентами и преподавателями (подписка на ПО)

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронно-библиотечной системе и электронной информационно-образовательной среде.

13. Материально–техническое обеспечение дисциплины

Образовательный процесс обеспечен учебными аудиториями для проведения учебных занятий, а именно для проведения занятий лекционного типа, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещениями для самостоятельной работы студентов.

Учебная аудитория для проведения занятий лекционного типа

Рабочее место преподавателя (стол, стул), рабочие места обучающихся, доска для написания мелом, кафедра.

Технические средства обучения: персональный компьютер-1, мультимедийное оборудование (проектор, проекционный экран) – 1.

Учебная аудитория для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации

Рабочее место преподавателя (стол, стул), рабочие места обучающихся, доска для написания мелом, кафедра.

Технические средства обучения: персональный компьютер-1, мультимедийное оборудование (проектор, проекционный экран) – 1.

Помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду института

Рабочие места обучающихся.

Технические средства обучения: персональный компьютер - 13.

Учебные аудитории соответствуют действующим противопожарным правилам и нормам, укомплектованы учебной мебелью.

Обновление рабочей программы дисциплины (модуля)

Наименование раздела рабочей программы, в который внесены изменения

(измененное содержание раздела)

обсуждено и рекомендовано к утверждению решением кафедры
наименование кафедры

от ____ ____ 20__ г., протокол № ____

одобрено Научно-методическим советом ____ от ____ ____ 20__ г.,
протокол № ____