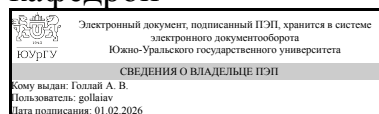


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Заведующий выпускающей
кафедрой



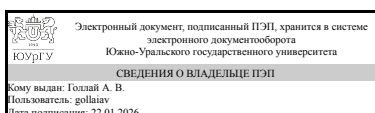
А. В. Голлай

РАБОЧАЯ ПРОГРАММА

**дисциплины 1.Ф.П0.02 Разработка клиент-серверных приложений
для направления 09.03.02 Информационные системы и технологии
уровень Бакалавриат
профиль подготовки ИТ-инженерия
форма обучения очная
кафедра-разработчик Центр подготовки топ-специалистов в сфере ИТ "Цифровой
Урал"**

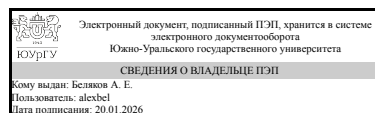
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, утверждённым приказом Минобрнауки от 19.09.2017 № 926

Зав.кафедрой разработчика,
Д.техн.н., доц.



А. В. Голлай

Разработчик программы,
старший преподаватель



А. Е. Беляков

1. Цели и задачи дисциплины

Целью дисциплины является освоение современных технологий разработки корпоративных клиент-серверных приложений на платформе .NET, включающих контейнеризацию, веб-сервисы, базы и хранилища данных, средства безопасности и мониторинга приложений. Задачи: освоение на практике последовательного процесса проектирования и программирования компонентов корпоративного клиент-серверного приложения, включающего такие этапы, как разработка схемы базы данных, разработка серверного слоя бизнес-логики в виде веб-сервиса (API), разработка клиентского веб-интерфейса пользователя в виде javascript-приложения.

Краткое содержание дисциплины

В рамках дисциплины изучается поэтапный процесс разработки клиент-серверного приложения и применяются на практике следующие технологии: 1. Терминал Windows с PowerShell 7. 2. Платформа .NET 8.0 LTS, язык C# 12, Visual Studio Community 2022. 3. ОС Windows: Docker Desktop (требуется WSL2). ОС Linux: Docker. Репозиторий образов DockerHub. 4. Entity Framework Core, провайдер для PostgreSQL (NuGet-пакет в VS), миграции БД, EF CodeFirst, LINQ. 5. Web-сервисы на .NET (ASP.NET Core WebAPI), Postman, C# async. 6. Интерфейс на Vue.js 3 версии. 7. Обратный прокси NGINX (балансировщик нагрузки, web-сервер для Vue.js). 8. JWT-токены, аутентификация, авторизация (роли) в приложениях Vue.js и ASP.NET Core. 9. Логирование в приложениях. 10. Кеширование web-запросов, снижение нагрузки на БД. 11. Резервное копирование и восстановление БД в Postgres.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ПК-5 Способен разрабатывать требования и проектировать программное обеспечение	Знает: принципы построения и виды архитектуры компьютерного программного обеспечения; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке компьютерного программного обеспечения; нормативно-технические документы (стандарты), определяющие требования к технической документации на компьютерное программное обеспечение; методы и средства проектирования компьютерного программного обеспечения; устройство и функционирование ИТ-систем/продуктов; методы моделирования и описания устройства и функционирования ИТ-систем/продуктов, их частей, обеспечения и окружения; базовые технологии взаимодействия и интеграции систем и компонентов, методы функциональной декомпозиции ИТ-систем Умеет: разрабатывать и изменять архитектуру компьютерного программного обеспечения и согласовывать ее с системным аналитиком и

	архитектором программного обеспечения; проектировать: структуры данных, программные интерфейсы; декомпозировать ИТ-системы и ИТ-продукты на подсистемы и элементы поставки; описывать: интерфейсы пользователя на логическом уровне, интеграции со смежными системами на логическом уровне Имеет практический опыт: проектирования компьютерного программного обеспечения с учетом требований актуальных нормативно-технических документов
--	---

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
Анализ требований и проектирование ПО, Архитектура ЭВМ	Разработка облачных приложений

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
Анализ требований и проектирование ПО	Знает: методы выявления, анализа и разработки требований при проектировании сложных программных систем; нотации моделирования бизнес-процессов в нотации BPMN для формирования требований к ПО; классификацию и структуру проектной документации на программные системы; стандарты и нотации моделирования (UML, BPMN IDEF0) для описания архитектуры и процессов; основы архитектуры программных систем и влияние архитектурных решений на качество ПО; правила оформления и ведения спецификаций, технических заданий, основные стандарты оформления технической документации на компьютерное программное обеспечение, возможности типовой ИС, предметную область автоматизации, инструменты и методы моделирования бизнес-процессов, методы и средства проектирования компьютерного программного обеспечения, методы и средства проектирования программных интерфейсов Умеет: создавать и описывать объектно-ориентированные модели предметной области; разрабатывать техническую и проектную документацию на основе требований и стандартов; применять унифицированные языки моделирования (UML, IDEF0) для описания архитектуры и компонентов системы; использовать шаблоны проектирования при разработке архитектуры и спецификаций

	модулей; моделировать бизнес-процессы в нотации BPMN и анализировать требования, выявлять противоречия и формировать корректные спецификации; оформлять документы в соответствии с установленными нормами и шаблонами организации, применять заданные стандарты и шаблоны для составления и оформления технической документации, анализировать исходную документацию в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС, разрабатывать документы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС, выбирать средства реализации требований к компьютерному программному обеспечению, вырабатывать варианты реализации компьютерного программного обеспечения, проводить оценку и обоснование рекомендуемых решений Имеет практический опыт: навыками подготовки и оформления документации (техническое задание, спецификация, концепция проекта, архитектурное описание) в соответствии со стандартами; инструментами и средствами CASE-технологий для создания диаграмм и спецификаций; навыками работы с инструментальными средствами моделирования бизнес-процессов и архитектуры ПО
Архитектура ЭВМ	Знает: методику настройки и наладки программно-аппаратных комплексов, возможности существующей программно-технической архитектуры, возможности современных и перспективных средств разработки программных продуктов, технических средств, методологии разработки компьютерного программного обеспечения и технологии программирования, методы оптимизации (кэш, память, тактовая частота процессора), инструменты профилирования, архитектуру компиляторов, оптимизации для выбранных архитектур процессоров, методику настройки и наладки программно-аппаратных комплексов Умеет: производить коллективную настройку и наладку программно-аппаратных комплексов, проводить сбор и систематизацию требований к компьютерному программному обеспечению, выявлять взаимосвязи и документировать требования к компьютерному программному обеспечению, вырабатывать варианты реализации требований к компьютерному программному обеспечению, анализировать бенчмарки, выявлять узкие места, разрабатывать код, оптимизированный для выбранной аппаратной архитектуры Имеет практический опыт: коллективной настройки и наладки программно-аппаратных комплексов

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 2 з.е., 72 ч., 52,25 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам
		в часах
		Номер семестра
		5
Общая трудоёмкость дисциплины	72	72
Аудиторные занятия:	48	48
Лекции (Л)	16	16
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	32	32
Лабораторные работы (ЛР)	0	0
Самостоятельная работа (СРС)	19,75	19,75
Разработка компьютерных программ	15	15
Подготовка к зачету	4,75	4,75
Консультации и промежуточная аттестация	4,25	4,25
Вид контроля (зачет, диф.зачет, экзамен)	-	зачет

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Проектирование архитектуры клиент-серверных приложений	6	2	4	0
2	Разработка слоя доступа к данным	6	2	4	0
3	Разработка слоя бизнес-логики (Backend)	12	4	8	0
4	Разработка интерфейса пользователя (Frontend)	12	4	8	0
5	Безопасность: авторизация и роли	6	2	4	0
6	Производительность и отслеживание ошибок	6	2	4	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Обзор архитектур клиент-серверных приложений	2
2	2	Технологии контейнеризации приложений. Docker (установка, команды, скрипты автоматизации)	1
3	2	Технология Entity Framework Core. Разработка системы сущностных классов. Миграции в базах данных	1
4	3	Технология ASP.NET MVC для разработки веб-сервисов	3
5	3	Отладка веб-сервисов через утилиту Postman	1
6	4	Разработка интерфейса пользователя на Vue.js	3
7	4	Балансировка нагрузки на интерфейса пользователя с использованием NGINX	1

8	5	Технологии обеспечения безопасности доступа к корпоративным приложениям	1
9	5	Авторизация в приложении по JWT-токенам	1
10	6	Логирование событий. Кеширование запросов. Резервное копирование данных	2

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Проектирование архитектуры приложения	4
2	2	Разработка слоя доступа к данным	4
3	3	Разработка слоя бизнес-логики	6
4	3	Отладка слоя бизнес-логики	2
5	4	Разработка интерфейса пользователя	6
6	4	Отладка интерфейса пользователя	2
7	5	Разработка логики авторизации и прав доступа	4
8	6	Оптимизация приложения (логирование и кеширование запросов)	4

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Разработка компьютерных программ	Слайды лекций, методические указания к заданиям	5	15
Подготовка к зачету	Слайды лекций, методические указания к заданиям	5	4,75

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учитывается в ПА
1	5	Текущий контроль	ЛР №1	15	15	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в	зачет

						виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 5 балла.	
2	5	Текущий контроль	ЛР №2	15	15	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 5 балла.	зачет
3	5	Текущий контроль	ЛР №3	15	15	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 5 балла.	зачет
4	5	Текущий контроль	ЛР №4	15	15	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла.	зачет

						2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 5 балла.	
5	5	Текущий контроль	ЛР №5	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	зачет
6	5	Текущий контроль	ЛР №6	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	зачет
7	5	Промежуточная аттестация	Зачет	-	100	По результатам ответов на вопросы устного зачета. Первый теоретический вопрос - максимум 30 баллов, второй теоретический вопрос - максимум 30 баллов, третий теоретический вопрос - максимум 40 баллов. За успешное выполнение ставится максимум баллов по заданию, если задание выполнено частично или с ошибками, то ставится от 0 до максимума по заданию пропорционально выполненному объему. Длительность зачета - 2 академических часа.	зачет

6.2. Процедура проведения, критерии оценивания

Вид	Процедура проведения	Критерии
-----	----------------------	----------

промежуточной аттестации		оценивания
зачет	При оценивании результатов учебной деятельности обучающегося по дисциплине используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся (Положение о БРС утверждено приказом ректора от 24.05.2019 г. № 179, в редакции приказа ректора от 10.03.2022 г. № 25-13/09). Оценка за дисциплину формируется на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля. Отлично: Величина рейтинга обучающегося по дисциплине 85...100 %. Хорошо: Величина рейтинга обучающегося по дисциплине 75...84 %. Удовлетворительно: Величина рейтинга обучающегося по дисциплине 60...74 %. Неудовлетворительно: Величина рейтинга обучающегося по дисциплине 0...59 %. Если студент согласен с оценкой, полученной по результатам текущего контроля, то он может в день, предшествующий промежуточной аттестации дать свое согласие в личном кабинете. В случае явки студента на промежуточную аттестацию, давшего свое согласие на автомат в личном кабинете, студент имеет право пройти мероприятия текущего контроля по дисциплине на промежуточной аттестации для улучшения своего рейтинга в день ее проведения. Снижение оценки в этом случае запрещено. Если студент не дал согласия в личном кабинете, то он может согласиться с оценкой лично на промежуточной аттестации в день ее проведения. Если студент не согласен с оценкой, то он имеет право пройти контрольно-рейтинговые мероприятия на промежуточной аттестации для улучшения своего рейтинга в день ее проведения. Фиксация результатов учебной деятельности по дисциплине проводится в день промежуточной аттестации на основе согласия студента, данного им в личном кабинете. При отсутствии согласия в журнале дисциплины фиксация результатов происходит при личном присутствии студента. Если студент не дал согласие в личном кабинете и не явился на промежуточную аттестацию – ему выставляется «неявка».	В соответствии с пп. 2.5, 2.6 Положения

6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ						
		1	2	3	4	5	6	7
ПК-5	Знает: принципы построения и виды архитектуры компьютерного программного обеспечения; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке компьютерного программного обеспечения; нормативно-технические документы (стандарты), определяющие требования к технической документации на компьютерное программное обеспечение; методы и средства проектирования компьютерного программного обеспечения; устройство и функционирование ИТ-систем/продуктов; методы моделирования и описания устройства и функционирования ИТ-систем/продуктов, их частей, обеспечения и окружения; базовые технологии взаимодействия и интеграции систем и компонентов, методы функциональной декомпозиции ИТ-систем	+	+	+	+	+	+	+
ПК-5	Умеет: разрабатывать и изменять архитектуру компьютерного программного обеспечения и согласовывать ее с системным аналитиком и	+	+	+	+	+	+	+

	архитектором программного обеспечения; проектировать: структуры данных, программные интерфейсы; декомпонировать ИТ-системы и ИТ-продукты на подсистемы и элементы поставки; описывать: интерфейсы пользователя на логическом уровне, интеграции со смежными системами на логическом уровне									
ПК-5	Имеет практический опыт: проектирования компьютерного программного обеспечения с учетом требований актуальных нормативно-технических документов	+	+	+	+	+	+	+	+	+

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) *основная литература:*

Не предусмотрена

б) *дополнительная литература:*

Не предусмотрена

в) *отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:*

Не предусмотрены

г) *методические указания для студентов по освоению дисциплины:*

1. Методические указания по разработке клиент-серверных приложения
2. Методические указания к проектированию приложения
3. Методические указания к структуре программного кода в виде шаблона приложения

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Методические указания по разработке клиент-серверных приложения
2. Методические указания к проектированию приложения
3. Методические указания к структуре программного кода в виде шаблона приложения

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	ЭБС издательства Лань	Сычев, А. В. Web-технологии : учебное пособие / А. В. Сычев. — Воронеж : ВГУ, 2021. — 163 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/455018 . — Режим доступа: для авториз. пользователей.
2	Основная литература	ЭБС издательства Лань	Лок, Э. ASP.Net Core в действии : руководство / Э. Лок ; перевод с английского Д. А. Беликова. — 3-е изд. — Москва : ДМК Пресс, 2025. — 1046 с. — ISBN 978-5-93700-183-2. —

			Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/464255 . — Режим доступа: для авториз. пользователей.
3	Основная литература	ЭБС издательства Лань	Гольчевский, Ю. В. Технологии разработки веб-приложений : учебное пособие / Ю. В. Гольчевский, А. В. Ермоленко, Н. О. Котелина. — Сыктывкар : СГУ им. Питирима Сорокина, 2023. — 84 с. — ISBN 978-5-87661-851-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/410771 . — Режим доступа: для авториз. пользователей.
4	Основная литература	ЭБС издательства Лань	Разработка фронт-энд части веб-ориентированного информационного обеспечения систем управления в организационных системах : учебное пособие / В. Н. Волков, А. В. Демидов, С. В. Новиков [и др.]. — Орел : ОГУ имени И.С. Тургенева, 2023. — 427 с. — ISBN 978-5-9929-1541-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/451013 . — Режим доступа: для авториз. пользователей.
5	Дополнительная литература	ЭБС издательства Лань	Саблина, В. А. Основы программирования на JavaScript : учебное пособие / В. А. Саблина, Е. А. Трушина. — Рязань : РГРТУ, 2022. — 96 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/380477 . — Режим доступа: для авториз. пользователей.
6	Дополнительная литература	ЭБС издательства Лань	Ермаков, С. Р. Основы веб-разработки: Практикум : учебное пособие / С. Р. Ермаков, П. В. Беляев, А. В. Симонова. — Москва : РТУ МИРЭА, 2023. — 128 с. — ISBN 978-5-7339-1892-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/382517 . — Режим доступа: для авториз. пользователей.
7	Дополнительная литература	ЭБС издательства Лань	Майкл, С. М. Разработка одностраничных веб-приложений / С. М. Майкл, К. П. Джош ; перевод с английского А. А. Слинкина. — Москва : ДМК Пресс, 2014. — 512 с. — ISBN 978-5-97060-072-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/69951 . — Режим доступа: для авториз. пользователей.
8	Дополнительная литература	ЭБС издательства Лань	Суханов, М. В. Основы Microsoft .NET Framework и языка программирования C# : учебное пособие / М. В. Суханов, И. В. Бачурин, И. С. Майров. — Архангельск : САФУ, 2014. — 96 с. — ISBN 978-5-261-00934-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/96543 . — Режим доступа: для авториз. пользователей.

Перечень используемого программного обеспечения:

1. Microsoft-Visual Studio(бессрочно)
2. Docker-Docker(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Лекции	240 (36)	Компьютер, ОС Windows, MS Visual Studio, VS Code, Docker, доска, проектор
Практические занятия и семинары	809 (36)	Компьютеры, ОС Windows, MS Visual Studio, VS Code, Docker