Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 30.10.2026 13:36:03
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Поволжский государственный университет сервиса» (ФГБОУ ВО «ПВГУС»)»

Высшая школа интеллектуальных систем и кибертехнологий

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Ф.01 «РАЗРАБОТКА ИНТЕРНЕТ-ПРИЛОЖЕНИЙ»

Направление подготовки:
10.03.01 Информационная безопасность

Направленность (профиль):
«Организация и технологии защиты информации (по отрасли или в сфере профессиональной деятельности)»

Квалификация выпускника: **бакалавр**

Тольятти 2023

Рабочая программа дисциплины «Разработка интернет-приложений» разработана в соответствии с федеральным государственным образовательным стандартом высшего образования - бакалавриат по направлению подготовки 10.03.01 Информационная безопасность, утвержденным приказом Министерства науки и высшего образования Российской Федерации от 17 ноября 2020 г. № 1427.

Составители:

старший преподаватель (ученая степень, ученое звание) Коновалова А.Д. (ФИО)

РПД обсуждена на заседании Высшей школы интеллектуальных систем и кибертехнологий

10.11.2023 г., протокол № 3

Директор ВШИСиКТ к.э.н., доцент (уч.степень, уч.звание) Филиппова О.А. (ФИО)

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1.1. Цель освоения дисциплины

Целью освоения дисциплины является:

- формирование способности разрабатывать, проверять и сопровождать небольшие интернет-приложения для решения профессиональных задач информационной безопасности с использованием языков высокого уровня, объектно-ориентированного подхода и технологий клиент-серверной разработки.

1.2. Перечень планируемых результатов обучения по дисциплине

Код и наименование компетенции	**Код и наименование индикатора достижения компетенции**	**Планируемые результаты обучения по дисциплине**	**Основание (ПС)** *для профессиональных компетенций
ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности	ИОПК-7.1. Использует языки программирования высокого уровня (объектно-ориентированное программирование) для создания программ ИОПК-7.2. Реализует на языке высокого уровня алгоритмы решения профессиональных задач	**Знает:** принципы клиент-серверного взаимодействия, назначение HTML, CSS, JavaScript и серверного языка Python; способы представления предметной области классами, организацию компонентов веб-фреймворка и работы с данными; алгоритмы проверки входных данных, обработки запросов, изменения состояния объектов и разграничения доступа, способы их тестирования **Умеет:** создавать компоненты интернет-приложения на языке высокого уровня с использованием классов, форм, моделей и шаблонов, связывать клиентскую и серверную части; реализовывать обработку учебных заявок на сопровождение средств защиты, проверку данных и принадлежности объекта пользователю, допустимые переходы состояния и обработку ошибок, проверять алгоритмы на положительных, отрицательных и граничных примерах **Владеет:** навыками разработки и сопровождения небольшого интернет-приложения с использованием Python, Django, JavaScript и системы контроля версий; документирования структуры классов и воспроизводимого запуска проекта; реализации и отладки алгоритмов профессионально ориентированной задачи, автоматизированного тестирования прав доступа, валидации и бизнес-правил, подготовки доказательств корректности и исправления обнаруженных дефектов	-

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина относится к факультативным дисциплинам, Блок Ф.00. Факультативные дисциплины программы бакалавриата.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1. Объем и структура дисциплины

Общая трудоемкость дисциплины составляет **2 з.е. (72 часа)**. их распределение по видам работ и семестрам представлено в таблице.

Виды учебных занятий и работы обучающихся	Трудоемкость, час
Общая трудоемкость дисциплины, час	**72**
Контактная работа обучающихся с преподавателем по видам учебных занятий (всего), в т.ч.:	**30**
занятия лекционного типа (лекции)	12
занятия семинарского типа (семинары, практические занятия, практикумы, коллоквиумы и иные аналогичные занятия)	18
Лабораторные работы	0/0
Самостоятельная работа всего, в т.ч.:	**42**
Самоподготовка по темам (разделам) дисциплины	42
Контроль (часы на зачет)	**0**
Промежуточная аттестация	**Зачёт**

Примечание: -/- объем часов соответственно для очно-заочной форм обучения.

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

В процессе освоения дисциплины может применяться электронное обучение и дистанционные образовательные технологии.

В процессе освоения дисциплины обучающиеся обеспечены доступом к электронной информационно-образовательной среде и электронно-библиотечным системам.

3.2. Содержание дисциплины, структурированное по темам

Планируемые результаты освоения: код формируемой компетенции и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Контактная работа			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 1. АРХИТЕКТУРА ИНТЕРНЕТ-ПРИЛОЖЕНИЯ И ОРГАНИЗАЦИЯ РАЗРАБОТКИ** Содержание лекции: Интернет-приложение как программная система с клиентской и серверной частями. Роли браузера, веб-сервера, приложения и хранилища; границы доверия между ними. Адрес ресурса, маршрут, метод, заголовки и тело HTTP-запроса; статус, заголовки и тело ответа. Назначение GET и POST, успешного ответа, перенаправления и ошибок. Отсутствие состояния в протоколе и способы организации пользовательского сеанса. Разделение представления, обработки запроса и предметной логики. Классы, объекты, свойства и методы как средство реализации предметной области на Python. Требования к учебному реестру заявок: роли пользователей, операции и проверяемые ограничения. Структура проекта Django, назначение приложения, конфигурации и маршрутизации. Изолированное окружение Python, фиксация зависимостей, запуск локального сервера разработки. Репозиторий Git, осмысленное изменение и исключение служебных файлов. Правила обращения с секретами и учебными данными. Перевод требования в критерий приемки и предварительный тест.	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия
	Практическая работа 1. Постановка задачи и создание каркаса интернет-приложения.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 2. КЛИЕНТСКИЙ ИНТЕРФЕЙС, ФОРМЫ И СЦЕНАРИИ JAVASCRIPT** Содержание лекции: Семантическая структура HTML-документа: заголовки, навигация, основной раздел, таблица и форма. Поля ввода, подписи, типы элементов управления и отправка данных. Доступность интерфейса: связанная подпись, клавиатурная навигация, видимый фокус, понятное сообщение об ошибке. CSS: селекторы, каскад, блочная модель, отступы, адаптация ширины, отделение оформления от структуры. JavaScript: значения, функции, объекты, коллекции, DOM и обработчики событий. Получение значений формы, вывод сообщений и подсчет длины текста. Работа с текстом через безопасные текстовые узлы. Клиентская проверка как средство удобства; обязательность независимой серверной проверки. Обработка пустых строк, пробелов и граничных	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия

Планируемые результаты освоения: код формируемой компетенции и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Контактная работа			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	длин. Отличие запрета кнопки в интерфейсе от запрета операции на сервере. Согласованность состояния интерфейса с ответом приложения. Инструменты разработчика: DOM, стили, консоль и сеть. Проверка интерфейса при выключенном JavaScript и узком окне.					
	Практическая работа 2. Разработка интерфейса реестра и формы заявки.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 3. СЕРВЕРНАЯ ОБРАБОТКА ЗАПРОСОВ И АЛГОРИТМЫ ВАЛИДАЦИИ** Содержание лекции: Цикл обработки запроса в Django: URL-конфигурация, представление, форма, предметная логика и шаблон. Получение данных и различение методов запроса. Класс формы, декларативные поля, нормализация и проверка значений; очищенные данные и сообщения об ошибках. Явное отделение доверенных атрибутов от пользовательского ввода. Построение алгоритма: проверка метода, разбор данных, валидация, выполнение действия, формирование ответа. Повторный показ ошибочной формы и перенаправление после успешного POST. Обработка отсутствующего ресурса и недопустимого метода. Шаблоны, контекст и наследование; автоматическое экранирование текстовых данных и ограничения его области действия. Состояния заявки и таблица допустимых переходов. Класс сервиса или метод модели как место предметного правила; исключение дублирования правила в интерфейсе и обработчике. Тестирование чистой функции перехода до подключения хранилища. Диагностика исключений в учебной среде; отсутствие подробностей внутреннего устройства в публичном сообщении об ошибке.	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия
	Практическая работа 3. Реализация серверной формы и правил изменения состояния.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 4. ХРАНЕНИЕ ДАННЫХ, МОДЕЛИ И ОПЕРАЦИИ С ЗАЯВКАМИ** Содержание лекции: Реляционное представление данных и объектно-реляционное отображение. Модель Django как класс: типы полей, обязательность, ограничения, связь с пользователем и временные метки.	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия

Планируемые результаты освоения: код формируемой компетенции и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Контактная работа			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	Первичный ключ как идентификатор, не как разрешение на доступ. Миграции схемы и их хранение вместе с кодом. SQLite для учебного проекта; границы переноса решения в другую среду. Создание, выборка и изменение объектов через ORM. Фильтрация, упорядочение и ограничение объема результатов. Параметризованные операции вместо конкатенации SQL. Согласованность проверок формы, предметного метода и ограничений базы. Назначение транзакции; атомарность изменения заявки и связанных данных. Сценарий одновременного изменения как ограничение наивной реализации. Присвоение владельца сервером по текущему пользователю. Выборка собственных объектов до доступа к карточке. Обработка отсутствующей записи. Подготовка повторяемого набора учебных данных, миграция новой базы и проверка сохранности после перезапуска. Различие резервной копии данных и репозитория исходного кода.					
	Практическая работа 4. Разработка модели данных и сохранение заявок.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов.
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 5. АУТЕНТИФИКАЦИЯ, РАЗГРАНИЧЕНИЕ ДОСТУПА И ЗАЩИТА ПРИЛОЖЕНИЯ** Содержание лекции: Различие аутентификации и авторизации. Использование штатного механизма пользователей, сессий, входа и выхода Django. Хранение паролей средствами фреймворка, исключение самодельной схемы. Проверка доступа к каждой операции и каждому объекту; принцип запрета при отсутствии разрешения. Выборка объектов по владельцу, предотвращение подмены идентификатора и массового назначения доверенных полей. CSRF при операциях с сеансовой аутентификацией: токен, middleware, формы и проверка отказа. Межсайтовое выполнение сценариев: контекст вывода, экранирование, опасность необоснованного safe и вставки HTML. Разграничение защиты от XSS, CSRF и SQL-инъекций. Сессионные cookie, HTTPS и настройки эксплуатационной среды. Секреты вне репозитория, ограниченный DEBUG, разрешенные хосты и журналирование без паролей и токенов. Ограничение частоты попыток входа как отдельная мера, требующая специальной реализации или внешнего компонента. Проверка требований на локальном стенде без обращения к	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия.

Планируемые результаты освоения: код формируемой компетенции и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Контактная работа			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	сторонним системам.					
	Практическая работа 5. Реализация и проверка контроля доступа.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов
ОПК-7 ИОПК-7.1 ИОПК-7.2	**ТЕМА 6. ТЕСТИРОВАНИЕ, ПОДГОТОВКА К ЗАПУСКУ И СОПРОВОЖДЕНИЕ ПРОЕКТА** Содержание лекции: Качество интернет-приложения как соответствие проверяемым требованиям. Уровни тестирования: функции предметной логики, формы, модель, обработчик и сквозной пользовательский сценарий. Положительные, отрицательные и граничные наборы. Изоляция тестовой базы и независимость тестов. Проверка не только статуса ответа, но и содержимого, изменений данных и отсутствия запрещенного эффекта. Тестовый клиент Django, подготовка пользователей, сессий и CSRF. Воспроизведение дефекта: шаги, ожидаемое поведение, фактическое поведение, причина, исправление и регрессионная проверка. Фиксация зависимостей и конфигурации, миграции и стартовые данные. Различие локального сервера разработки и эксплуатационного развертывания через соответствующий сервер приложений. Предварительная проверка настроек, доступов, HTTPS, журналов и восстановления данных. README, структура репозитория, описание классов, правила изменений и известные ограничения. Приемка проекта по матрице требований и демонстрация самостоятельного владения кодом. Направления развития: пагинация, конкурентные изменения, централизованные журналы и управление зависимостями.	2	-	-	-	Лекция-визуализация (в т.ч. в ЭИОС) Тестирование по теме лекционного занятия
	Практическая работа 6. Комплексная проверка и защита интернет-приложения.	-	-	3	-	Отчёт по практической работе
	Самостоятельная работа.	-	-	-	7	Самостоятельное изучение учебных материалов
ИТОГО		**12**	**0/0**	**18**	**42**	

Примечание: -/- объем часов соответственно для очно-заочной форм обучения.

4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

4.1. Общие методические рекомендации по освоению дисциплины, образовательные технологии

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

При проведении учебных занятий по дисциплине обеспечивается развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплины в форме курса, составленного на основе результатов научных исследований, проводимых университетом, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

- балльно-рейтинговая технология оценивания;
- электронное обучение;

Для оценки знаний, умений, навыков и уровня сформированности компетенции по дисциплине применяется балльно-рейтинговая система контроля и оценки успеваемости студентов. В основу балльно-рейтинговой системы положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущего контроля успеваемости. Максимальное количество баллов в семестре – 100.

4.2. Методические указания для обучающихся по освоению дисциплины на занятиях лекционного типа

Лекционный курс предполагает систематизированное изложение основных вопросов тематического плана. В ходе лекционных занятий раскрываются базовые вопросы в рамках каждой темы дисциплины. Обозначаются ключевые аспекты тем, а также делаются акценты на наиболее сложные и важные положения изучаемого материала.

Лекционные занятия проводятся в поточной аудитории с применением мультимедийного проектора в виде учебной презентации или в ЭИОС университета.

В ходе лекционных занятий рекомендуется конспектирование учебного материала. Возможно ведение конспекта лекций в виде интеллект-карт.

Отдельные темы предлагаются для самостоятельного изучения (конспектируются).

Материалы лекций являются опорной основой для подготовки обучающихся к практическим занятиям / лабораторным работам и выполнения заданий самостоятельной работы, а также к мероприятиям текущего контроля успеваемости и промежуточной аттестации по дисциплине.

4.4. Методические указания для обучающихся по освоению дисциплины на занятиях семинарского типа / на практических занятиях

Практические (семинарские) занятия представляют собой детализацию лекционного теоретического материала, проводятся в целях закрепления курса и охватывают все основные разделы. Основной формой проведения семинаров и практических занятий является обсуждение наиболее проблемных и сложных вопросов по отдельным темам, а также решение задач и разбор примеров и ситуаций в аудиторных условиях.

Практические (семинарские) занятия обучающихся обеспечивают:

- проверку и уточнение знаний, полученных на лекциях;

- получение умений и навыков составления докладов и сообщений, обсуждения вопросов по учебному материалу дисциплины;

- подведение итогов занятий по рейтинговой системе, согласно технологической карте дисциплины.

Практические занятия организуются, в том числе в форме практической подготовки, которая предусматривает участие обучающихся в выполнении отдельных элементов

работ, связанных с будущей профессиональной деятельностью.

4.5. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку обучающегося к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающегося на занятиях и в качестве выполненных практических заданий и других форм текущего контроля.

Самостоятельная работа студентов включает:

1. Изучение учебной литературы по курсу.
2. Работу с ресурсами Интернет
3. Самостоятельное изучение учебных материалов

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы.

Для обучающихся по заочной форме обучения самостоятельная работа является основным видом учебной деятельности.

Для обеспечения самостоятельной работы обучающихся используется электронный учебный курс, созданный в ЭИОС университета http://sdo.tolgas.ru/.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Вся литература, включенная в данный перечень, представлена в виде электронных ресурсов в электронной библиотеке университета (ЭБС). Литература, используемая в печатном виде, представлена в научной библиотеке университета в объеме не менее 0,25 экземпляров на одного обучающегося.

Основная литература

1. Заяц, А. М. Инструментальные средства инфокоммуникационных систем. Теория и практика: учеб. пособие для вузов / А. М. Заяц, А. А. Логачев. - Документ Reader. - Санкт-Петербург [и др.]: Лань, 2023. - 208 с. - URL: https://reader.lanbook.com/book/311786 (дата обращения: 29.04.2023). - Режим доступа: для авториз. пользователей. - ISBN 978-5-507-45681-9. - Текст: электронный.

2. Коваленко, В. В. Проектирование информационных систем: учеб. пособие для студентов (бакалавров и специалистов) вузов и магистров по направлению 09.03.03 "Приклад. информатика" (профиль "Приклад. информатика в экономике") / В. В. Коваленко. - 2-е изд., перераб. и доп. - Документ read. - Москва: ФОРУМ [и др.], 2023. - 357 с.: ил., табл. - (Высшее образование. Бакалавриат). - Прил. - URL: https://znanium.ru/read?id=415461 (дата обращения: 04.02.2023). - Режим доступа: для авториз. пользователей. - ISBN 978-5-00091-637-7. - 978-5-16-107012-3. - Текст: электронный.

Дополнительная литература

3. Фридман, А. Л. Построение Интернет-приложений на языке Java: практ. курс / А. Л. Фридман. - 2-е изд., стер. - Москва: Горячая линия -Телеком, 2012. - 335 с.: табл. - ISBN 978-5-9912-0298-5: 320-00. - Текст: непосредственный.

4. Эспозито, Д. Разработка приложений для Windows 8 на HTML5 и JavaScript / Д. Эспозито, Ф. Эспозито. - Санкт-Петербург: Питер, 2014. - 384 с.: ил. - (Библиотека программиста). - ISBN 978-5-496-00794-8: 505-20. - Текст: непосредственный.

5. Фримен, Э. Изучаем программирование на HTML5. Создание веб-приложений с использованием JavaScript = Head First HTML5 Programming / Э. Фримен, Э. Робсон. - Санкт-Петербург: Питер, 2013. - 640 с.: ил. - ISBN 978-5-459-00952-1: 155-00. - Текст: непосредственный.

6. Винарский, Я. С. WEB-аппликации в Интернет-маркетинге. Проектирование, создание и применение: практ. пособие / Я. С. Винарский, Р. Д. Гутгарц. - Документ read. - Москва: ИНФРА-М, 2023. - 269 с. - (Среднее профессиональное образование). - Прил. - URL: https://znanium.ru/read?id=420081 (дата обращения: 24.04.2023). - Режим доступа: для авториз. пользователей. - ISBN 978-5-16-014219-7. - 978-5-16-106719-2. - Текст: электронный.

7. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript и CSS = Learning PHP, MySQL, JavaScript, and CSS / Р. Никсон; пер. с англ. Н. Вильчинского. - 2-е изд. - Санкт-Петербург: Питер, 2013. - 560 с.: ил. - (Бестселлеры O'Reilly). - Прил. - ISBN 978-5-496-00187-8: 300-00. - Текст: непосредственный.

8. Колисниченко, Д. Н. PHP и MySQL. Разработка Web-приложений / Д. Н. Колисниченко. - 5-е изд., перераб. и доп. - Санкт-Петербург: БХВ-Петербург, 2015. - 591 с.: ил. - (Профессиональное программирование). - Прил. - Предм. указ. - ISBN 978-5-9775-3514-4: 513-77. - Текст: непосредственный.

9. Янка, Т. Безопасность WEB-приложений. Исчерпывающий гид для начинающих разработчиков / Т. Янка; [пер. с англ. М. А. Райтмана]. - Москва: Эксмо, 2023. - 462 с.: ил. - (КиберБез. Лучшие книги о безопасности в сетях). - ISBN 978-5-04-171803-9: 544-00. - Текст: непосредственный.

5.2. Профессиональные базы данных, информационно-справочные системы, интернет-ресурсы

1. 3DNEWS.RU : Электронное периодическое издание : сайт. – Москва, 1997 - . - URL: https://3dnews.ru/ (дата обращения: 09.11.2023). – Текст: электронный.
2. eLIBRARY.RU : научная электронная библиотека : сайт. – Москва, 2000 - . - URL: https://elibrary.ru (дата обращения: 09.11.2023). – Режим доступа: для зарегистрир. пользователей. – Текст: электронный.
3. КонсультантПлюс : справочная правовая система : сайт / ЗАО «КонсультантПлюс». – Москва, 1992 - . - URL: http://www.consultant.ru (дата обращения: 09.11.2023). - Текст : электронный.
4. Электронная библиотека. Техническая литература : сайт. - URL : http://techliter.ru/ (дата обращения: 09.11.2023). - Текст : электронный.
5. Электронная библиотечная система Поволжского государственного университета сервиса : сайт / ФГБОУ ВО «ПВГУС». – Тольятти, 2010 - . - URL. : http://elib.tolgas.ru (дата обращения: 09.11.2023). - Режим доступа: для авториз. пользователей. - Текст : электронный.
6. Электронно-библиотечная система Znanium.com : сайт / ООО "ЗНАНИУМ". – Москва, 2011 - . - URL: https://znanium.com/ (дата обращения: 09.11.2023). - Режим доступа: для авториз. пользователей. - Текст : электронный.
7. Электронно-библиотечная система Лань : сайт / ООО "ЭБС ЛАНЬ". - Москва, 2011 - . - URL: https://e.lanbook.com/ (дата обращения: 09.11.2023). - Режим доступа: для авториз. пользователей. - Текст : электронный.

5.3. Программное обеспечение

Информационное обеспечение учебного процесса по дисциплине осуществляется с использованием следующего программного обеспечения (лицензионного и свободно распространяемого), в том числе отечественного производства:

№ п/п	Наименование	Условия доступа
1	Microsoft Windows	из внутренней сети университета (лицензионный договор)
2	Microsoft Office	из внутренней сети университета (лицензионный договор)
3	КонсультантПлюс	из внутренней сети университета (лицензионный договор)
4	СДО MOODLE	из любой точки, в которой имеется доступ к сети Интернет (доступ через ЭИОС университета)

6. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных учебным планом и рабочей программой дисциплины, оснащенные оборудованием и техническими средствами обучения.

Занятия лекционного типа. Учебные аудитории для занятий лекционного типа укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации (стационарные или переносные наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук), учебно-наглядные пособия (презентации по темам лекций), обеспечивающие тематические иллюстрации, соответствующие данной программе дисциплины.

Занятия семинарского типа. Учебные аудитории для занятий семинарского типа укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации (стационарные или переносные наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук). Для проведения лабораторных работ используется учебная аудитория «Лаборатория информационных технологий, информатики и методов программирования // Учебная аудитория для проведения занятий лекционного типа, семинарского типа, курсового проектирования, консультаций, текущего контроля и промежуточной аттестации // Помещение для хранения и профилактического обслуживания учебного оборудования (Г-413)», оснащенная следующим оборудованием: рабочие места обучающихся; Рабочее место преподавателя; Учебно-наглядные пособия, обеспечивающие тематические иллюстрации. Компьютер Acyc Intel G530\iH61\GeForceGT430\4096 Mb DDR3\500Gb\LCD 22" - 14 шт., Компьютер в сборе iP2.0 LCD - 1 шт., принтер HP 1200 - 1 шт., мобильный экран проекционный DraperLuma - 1 шт., мобильный проектор Sanyo PLC, Сетевое оборудование D-Link DES-1024D, 24-портовый; локальная сеть 100 мб с доступом в Интернет 50 мб. Лаборатория оснащена кондиционером.

Промежуточная аттестация. Для проведения промежуточной аттестации по дисциплине используются компьютерные классы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета и/или учебные аудитории, укомплектованные мебелью и техническими средствами обучения.

Самостоятельная работа. Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде университета. Для организации самостоятельной работы обучающихся используются:

- компьютерные классы университета;
- библиотека (медиазал), имеющая места для обучающихся, оснащенные компьютерами с доступом к базам данных и сети Интернет.

Электронная информационно-образовательная среда университета (ЭИОС). Каждый обучающийся в течение всего периода обучения обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде университета (ЭИОС) http://sdo.tolgas.ru/ из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории университета, так и вне ее.

ЭИОС университета обеспечивает:

- доступ к учебным планам, рабочим программам дисциплин (модулей), программам практик, электронным учебным изданиям и электронным образовательным ресурсам, указанным в рабочих программах дисциплин (модулей), программах практик;
- формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы.

В случае реализации образовательной программы с применением электронного обучения, дистанционных образовательных технологий ЭИОС дополнительно обеспечивает:

фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательной программы;

проведение учебных занятий, процедур оценки результатов обучения, реализация которых предусмотрена с применением электронного обучения, дистанционных образовательных технологий;

взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействия посредством сети "Интернет".

7. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБУЧЕНИЯ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса инвалидов и лиц с ограниченными возможностями здоровья, в том числе для дистанционного обучения. Для этого требуется заявление студента (его законного представителя) и заключение психолого-медико-педагогической комиссии (ПМПК).

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида, могут предлагаться следующие варианты восприятия учебной информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.
- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.
- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

8. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (ФОНД ОЦЕНОЧНЫХ СРЕДСТВ) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

8.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Для оценки знаний, умений, навыков и уровня сформированности компетенции по дисциплине применяется балльно-рейтинговая система контроля и оценки успеваемости студентов. В основу балльно-рейтинговой системы положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущего контроля успеваемости. Максимальное количество баллов в семестре – 100

Шкала оценки результатов освоения дисциплины, сформированности результатов обучения

Форма промежуточной аттестации	Уровень сформированности	100-балльная шкала, %	Шкала освоения, %	5-балльная дифференцированная оценка/балл
зачет	допороговый	ниже 61	ниже 61	не зачтено
	пороговый	61–85,9	61–69,9	зачтено
	пороговый	61–85,9	70–85,9	зачтено
	повышенный	86–100	86–100	зачтено

По итогам текущей успеваемости студенту может быть выставлена оценка по промежуточной аттестации в соответствии за набранными за семестр баллами (по накопительному рейтингу). Студентам, набравшим в ходе текущего контроля успеваемости по дисциплине от 61 до 100 баллов и выполнившим все обязательные виды запланированных учебных занятий, по решению преподавателя без прохождения промежуточной аттестации выставляется оценка в соответствии со шкалой оценки результатов освоения дисциплины.

Результат обучения считается сформированным (повышенный уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 86 до 100, что соответствует повышенному уровню сформированности результатов обучения.

Результат обучения считается сформированным (пороговый уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 61 до 85,9, что соответствует пороговому уровню сформированности результатов обучения.

Результат обучения считается несформированным, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже 61, что соответствует допороговому уровню.

Формы текущего контроля успеваемости

Формы текущего контроля	Количество контрольных точек	Количество баллов за 1 контр. точку	Макс. возм. кол-во баллов
Посещение лекционного занятия (конспект лекций)	6	2	12
Выполнение практических работ	6	10	60
Самостоятельная работа (портфолио по темам)	6	4	24
Творческий рейтинг (итоговая защита проекта)	1	4	4
Итого по дисциплине			**100 баллов**

Система оценивания представлена в электронном учебном курсе по дисциплине http://sdo.tolgas.ru/.

8.2. Типовые контрольные задания или иные материалы для ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ

8.2.1. Типовые задания к практическим (семинарским) занятиям

Практическая работа 1. Постановка задачи и создание каркаса интернет-приложения

Задание 1. Определите границы системы, роли двух обычных пользователей А и Б и сценарии создания, просмотра и изменения собственной заявки. Для каждого сценария запишите входные данные, ожидаемый результат и условие отказа. Исключите из минимального объема почтовые рассылки, платежи и интеграцию с рабочими средствами защиты.

Задание 2. Изобразите путь запроса от браузера до обработчика и обратно. Отдельно обозначьте будущую базу данных и место проверки прав. Составьте таблицу маршрутов: список, создание, карточка, изменение и вход. Укажите допустимый HTTP-метод; GET не должен изменять состояние заявки.

Задание 3. Создайте или дополните класс учебной заявки на Python: атрибуты названия и состояния, метод получения краткого представления. Объясните различие класса и экземпляра; подготовьте два независимых объекта и проверьте отсутствие общего изменяемого состояния.

Задание 4. Разверните предоставленный каркас Django в отдельном окружении, выполните проверку конфигурации и запустите локальный сервер. Реализуйте маршрут главной страницы с осмысленным ответом. В инструментах браузера найдите запрос, код ответа и тип содержимого.

Задание 5. Инициализируйте репозиторий или создайте рабочую ветку. Исключите окружение, секреты и локальную базу из отслеживания. Зафиксируйте зависимости и небольшой законченный набор изменений; проверьте состав коммита до сохранения.

Задание 6. Подготовьте README с предпосылками, командами установки и запуска, адресом локальной страницы и известными ограничениями. Проверьте инструкции на новой рабочей копии или после пересоздания окружения; запишите фактический результат проверки.

Практическая работа 2. Разработка интерфейса реестра и формы заявки

Задание 1. Подготовьте семантические страницы списка и карточки заявки с единым заголовком и навигацией. Выведите состояние понятным пользователю текстом, сохранив связь с машинным значением. Не используйте цвет как единственный способ различить состояния.

Задание 2. Создайте форму с явными подписями к полям, ограничениями длины и областью сообщений. Сохраните возможность перехода по элементам с клавиатуры. Ошибка должна указывать поле и способ исправления; исходное корректное содержимое других полей не должно теряться.

Задание 3. Вынесите CSS в отдельный файл. Настройте читаемую ширину содержимого, перенос длинных строк и расположение элементов на узком экране. Проверьте, что увеличение масштаба не скрывает кнопку отправки и сообщения.

Задание 4. Реализуйте на JavaScript подсчет длины названия и проверку пустой строки после trim. Обработчик должен работать с фактическим значением поля и обновлять текст сообщения без интерпретации ввода как HTML. Объясните назначение функции и ее параметров.

Задание 5. Проверьте пустое значение, одни пробелы, допустимый текст, граничную длину и превышение границы. Повторите основной сценарий без JavaScript. Зафиксируйте, какие ограничения интерфейс обеспечивает, а какие требуют проверки на сервере.

Задание 6. Проанализируйте одну ошибку интерфейса средствами браузера: неверный селектор, отсутствующий обработчик или неудобный переход фокуса. Запишите шаги воспроизведения, причину, изменение кода и повторную проверку. Сохраните завершенную версию в репозитории.

Практическая работа 3. Реализация серверной формы и правил изменения состояния

Задание 1. Создайте класс формы с обязательными полями названия и описания. Выполняйте удаление краевых пробелов до проверки непустоты и длины. Объясните назначение класса, наследования и метода дополнительной очистки. Не принимайте владельца из формы.

Задание 2. Реализуйте GET для показа пустой формы и POST для ее обработки. При некорректных данных верните ту же форму с ошибками и кодом 200; при успехе выполните перенаправление с кодом 302 на страницу результата. Примените защиту CSRF, предусмотренную каркасом.

Задание 3. Передавайте данные в шаблон через контекст и выводите пользовательские значения как текст с экранированием. Проверьте отображение строки с угловыми скобками. Не отключайте экранирование для произвольного ввода.

Задание 4. Реализуйте правило изменения состояния методом класса: таблица разрешенных пар либо эквивалентная логика. Для неизвестного состояния, обратного перехода, повтора и пропуска этапа верните понятную ошибку, не изменяя объект.

Задание 5. Создайте проверки двух допустимых переходов и всех остальных сочетаний трех известных состояний. Добавьте неизвестное значение. Для отказов проверяйте одновременно сообщение и сохранение исходного состояния.

Задание 6. Сопоставьте обработчик с псевдокодом: где заканчивается проверка формы, где применяется предметное правило, где выбирается ответ. Продемонстрируйте работу при обходе клиентской проверки и объясните, почему результат остается корректным.

Практическая работа 4. Разработка модели данных и сохранение заявок

Задание 2. Создайте миграцию и примените ее к чистой учебной базе. Проверьте, что схема воспроизводится без ручного изменения таблиц. Зафиксируйте миграцию в репозитории; файл с локальными данными не включайте в исходный код.

Задание 3. Подключите сохранение только после успешной серверной проверки. Начальное состояние и владелец назначаются сервером. Проверьте, что переданные клиентом owner или произвольное начальное состояние не используются для назначения этих атрибутов.

Задание 4. Реализуйте список собственных заявок и получение карточки из выборки, ограниченной текущим пользователем. Упорядочьте записи по времени создания с дополнительным стабильным ключом. Для отсутствующего или чужого объекта в защищенной выборке возвращайте 404.

Задание 5. Подключите изменение названия, описания и состояния к существующим проверкам. Изменение состояния выполняйте только разрешенным переходом. Если несколько связанных операций должны завершаться вместе, обозначьте границу транзакции; не утверждайте, что она сама решает все конфликты параллельного изменения.

Задание 6. Проверьте создание, чтение и допустимое изменение после перезапуска приложения. Сопоставьте состояние базы до и после недопустимого запроса. Подготовьте воспроизводимый набор данных для А и Б и запишите ограничения учебной реализации.

Практическая работа 5. Реализация и проверка контроля доступа

Задание 1. Подключите штатные страницы входа и выхода; выход выполняйте способом, поддерживаемым выбранной версией Django, с POST и CSRF-защитой. Покажите, где хранится сеансовая информация и какие данные нельзя выводить в отчет. Проверьте доступ после завершения сеанса.

Задание 2. Составьте матрицу доступа для А, Б и анонимного посетителя к списку, карточке, созданию и изменению заявки. В каждой защищенной операции проверьте аутентификацию, затем принадлежность объекта. Не ограничивайтесь скрытием кнопки в шаблоне.

Задание 3. Проверьте чтение и изменение чужой заявки прямым запросом по известному

идентификатору. Ожидаются 404 и отсутствие изменения данных. Повторите проверку для несуществующего идентификатора; сравните наблюдаемое поведение.

Задание 4. Проверьте ошибочный или отсутствующий CSRF-токен и корректный запрос. Для автоматизированной проверки используйте тестовый клиент с явно включенной проверкой CSRF. Ожидаемый отказ - 403; до и после запроса состояние базы должно совпадать.

Задание 5. Проверьте вывод учебной строки с символами разметки в названии и описании: она должна отображаться как текст. Найдите в проекте места вставки HTML и отключения экранирования, обоснуйте удаление небезопасных конструкций. Не переносите проверочную строку в сторонние системы.

Задание 6. Подготовьте конфигурационный перечень: секрет вне репозитория, DEBUG=False для эксплуатации, явные разрешенные хосты, HTTPS и защищенные cookie при HTTPS, журналы без паролей и токенов. Отделите фактически реализованную меру от рекомендации для будущего размещения.

Практическая работа 6. Комплексная проверка и защита интернет-приложения

Задание 1. Составьте матрицу «требование - компонент - тест - ожидаемый результат». Обязательно включите оба индикатора: классы формы и модели для ИОПК-7.1; алгоритмы валидации, перехода и доступа для ИОПК-7.2. Для каждого требования укажите проверяемое свидетельство.

Задание 2. Создайте автоматизированные тесты корректной формы, пустого и слишком длинного значения, разрешенного и запрещенного перехода, анонимного запроса, чужого объекта и подмены владельца. Проверяйте состояние базы после отказа. Добавьте отдельную проверку CSRF с соответствующим режимом клиента.

Задание 3. Выполните сквозной сценарий: вход А, создание заявки, просмотр, два разрешенных перехода, выход; затем вход Б и отказ при попытке открыть заявку А. Зафиксируйте ожидаемые статусы и значения до выполнения и сопоставьте их с фактическими.

Задание 4. Выберите воспроизводимый дефект из собственной истории разработки либо предоставленный преподавателем. Добавьте тест, который выявляет дефект, исправьте код и повторите связанные проверки. Объясните первопричину и почему изменение не ломает соседний сценарий.

Задание 5. Проверьте установку по README в чистом окружении, создание базы миграциями и загрузку учебных данных. Подготовьте перечень действий перед эксплуатационным размещением; отметьте, что локальный runserver не используется как производственный сервер.

Задание 6. Защитите проект: покажите один класс, один нетривиальный алгоритм и один отрицательный тест. По измененному преподавателем ограничению объясните необходимые правки и добавьте проверку. Сформулируйте ограничения проекта и приоритет следующего улучшения.

8.3. Типовые контрольные задания или иные материалы для проведения ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Форма проведения промежуточной аттестации по дисциплине: зачет (по результатам накопительного рейтинга или в форме компьютерного тестирования).

Устно-письменная форма по билетам предполагается, как правило, для сдачи академической задолженности.

Перечень вопросов и заданий для подготовки к зачету

ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности

1. Объясните архитектуру клиент-серверного интернет-приложения и покажите границы доверия на примере реестра заявок.
2. Опишите состав HTTP-запроса и ответа, назначение методов GET и POST и выбор статуса для типовых сценариев.
3. Покажите, как требование к учебному сервису преобразуется в маршрут, компонент и проверяемый критерий приемки.
4. Объясните назначение классов, экземпляров и методов Python при моделировании заявки; приведите пример независимых объектов.
5. Опишите структуру проекта Django, назначение URL-конфигурации, представления, формы, модели и шаблона.
6. Объясните порядок подготовки окружения, фиксации зависимостей и ведения репозитория без секретов и локальных данных.
7. Опишите семантическую структуру страницы списка и формы заявки; обоснуйте выбор элементов HTML.
8. Объясните каскад CSS и блочную модель; предложите способ сохранить удобство формы на узком экране.
9. Покажите алгоритм обработки события ввода на JavaScript: получение значения, проверка и вывод сообщения.
10. Сравните клиентскую и серверную валидацию; объясните поведение приложения при отключенном JavaScript.
11. Опишите требования доступности формы: подписи, порядок фокуса, управление клавиатурой и сообщения об ошибке.
12. Объясните безопасный вывод пользовательского текста в DOM и предложите проверку граничных длин полей.
13. Опишите последовательность серверной обработки GET и POST для формы создания заявки.
14. Объясните реализацию класса формы, нормализацию строки, проверку обязательности и длины значения.
15. Сравните обработку корректной и ошибочной формы: статус ответа, сохранение введенных данных и перенаправление.
16. Разработайте алгоритм допустимых переходов new → in_progress → closed; определите поведение для повтора и неизвестного значения.
17. Объясните назначение контекста и наследования шаблонов, автоматического экранирования и его ограничений.
18. Покажите, как проверить предметный метод класса независимо от HTTP-обработчика; приведите положительные и отрицательные примеры.
19. Опишите модель заявки, типы полей, связь с владельцем и назначение временных меток.
20. Объясните роль миграций и порядок воспроизведения схемы базы в новом окружении.
21. Покажите алгоритм создания заявки через ORM с назначением владельца на сервере.
22. Опишите выборку собственных объектов, стабильную сортировку и обработку отсутствующей записи.
23. Объясните назначение транзакции и ограничения простой реализации при одновременном изменении заявки.

24. Сравните ограничения формы, предметного метода и базы данных; объясните применение параметризованных запросов.

25. Различите аутентификацию и авторизацию; опишите вход, сеанс и выход в выбранном фреймворке.

26. Разработайте проверку принадлежности объекта пользователю для чтения и изменения; объясните отказ для чужой заявки.

27. Объясните назначение CSRF-защиты и порядок проверки отказа при отсутствующем токене.

28. Объясните причины XSS и способы безопасного вывода в шаблоне и DOM; укажите ограничения экранирования.

29. Покажите, почему подмена владельца в запросе и скрытие кнопки требуют разных проверок на сервере.

30. Опишите конфигурацию эксплуатации: секреты, DEBUG, разрешенные хосты, HTTPS, cookie и безопасные журналы.

31. Сравните модульные, интеграционные и сквозные проверки проекта; укажите место тестового клиента.

32. Составьте набор положительных, отрицательных и граничных тестов формы и переходов состояния.

33. Объясните, как тестировать отказ доступа и одновременно доказывать отсутствие изменения данных.

34. Опишите цикл исправления дефекта с регрессионным тестом и повторной проверкой связанных сценариев.

35. Составьте инструкцию воспроизводимого запуска и перечень отличий локальной разработки от эксплуатационного размещения.

36. Покажите связь требований, классов, алгоритмов и тестов в итоговом проекте; предложите обоснованное улучшение.