Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 30.10.2026 13:32:54
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42ba

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования «Поволжский государственный университет сервиса»
(ФГБОУ ВО «ПВГУС»)

УТВЕРЖДЕНО

на заседании Высшей школы интеллектуальных систем и кибертехнологий

Протокол от 10.11.2023 № 3

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(фонд оценочных средств)
для проведения промежуточной аттестации по дисциплине

Ф.01 «Разработка интернет-приложений»
наименование дисциплины

по образовательной программе высшего образования – программе **бакалавриат**

«Организация и технологии защиты информации (по отрасли или в сфере профессиональной деятельности)»
наименование образовательной программы

10.03.01 «Информационная безопасность»
шифр, наименование направления подготовки / специальности

Составители Коновалова Анжелика Денисовна, старший преподаватель, Высшая школа интеллектуальных систем и кибертехнологий
ФИО, должность, структурное подразделение, ученая степень, ученое звание

Тольятти
2023

1. Паспорт фонда оценочных средств (далее – ФОС)

1.1. Планируемые результаты обучения по дисциплине

Код компетенции	Наименование компетенции	Код и наименование индикаторов достижения компетенции
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

В результате изучения дисциплины обучающийся должен:

владеть: 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

уметь: 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

знать: 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

1.2. Содержание дисциплины

№	Тема (раздел дисциплины) (в соответствии с РПД)	Код компетенции
1	Архитектура интернет-приложения и организация разработки	ОПК-7
2	Клиентский интерфейс, формы и сценарии JavaScript	ОПК-7
3	Серверная обработка запросов и алгоритмы валидации	ОПК-7
4	Хранение данных, модели и операции с заявками	ОПК-7
5	Аутентификация, разграничение доступа и защита приложения	ОПК-7
6	Тестирование, подготовка к запуску и сопровождение проекта	ОПК-7

1.3. Система оценивания по дисциплине

Дисциплина изучается в течении одного семестра

Форма промежуточной аттестации - зачет.

Уровень сформированности	100-балльная шкала, %	Шкала освоения, %	5-балльная дифференцированная оценка/балл
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

2. Перечень оценочных материалов для проведения промежуточной аттестации

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в ходе текущего контроля успеваемости: практических занятий, тестирования, опросов, проверки самостоятельной работы.

В ходе проведения промежуточной аттестации осуществляется контроль и оценка результатов освоения компетенций.

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.