

Письменные ответы на вопросы

Вопросы

1. Какую технику тест-дизайна ты используешь, чтобы составить тест-кейсы для задачи ниже? Обоснуй свой выбор.

Для составления тест-кейсов поиска квартиры можно использовать классы эквивалентности и граничные значения.

Классы эквивалентности помогают протестировать представительные значения для каждого фильтра, например, разные ЖК, типы квартир, отделка и наличие балкона.

Граничные значения фокусируются на крайних и средних значениях, таких как этажи 1, 9 и 5, чтобы выявить ошибки, возникающие на границах диапазонов.

Таким образом, тесты будут включать проверки для разных ЖК, типов квартир, отделки, балкона и этажей на границах и в середине диапазона.

2. Посмотри на тест-кейс ниже. Что стоит изменить и почему?

Заголовок можно сделать более конкретным, чтобы сразу было ясно, о чем идет речь. Например: “Проверка отображения главной страницы Яндекс Практикума для неавторизованного пользователя”. Также стоит уточнить предусловия, например: “Пользователь не авторизован в системе Яндекс Практикума, все куки и данные авторизации удалены” или используется режим инкогнито. Это поможет избежать влияния предыдущих сессий. Ожидаемый результат можно описать более подробно, например: “На экране отображаются элементы ‘Войти’ и ‘Регистрация’”.

3. Ты тестируешь чек-бокс «Заплатить сразу» в форме создания заказа. Ожидаемый результат — чек-бокс отмечен, в базе данных это поле содержит true. Но на самом деле в базе почему-то стоит false. Это ошибка — её нужно локализовать. Тебе известно, что архитектура приложения включает фронтенд, бэкенд и базу данных. Опиши, что нужно сделать в Devtools, чтобы локализовать ошибку.

Сначала нужно проверить в Elements, правильно ли устанавливается атрибут checked у чек-бокса при его активации. Если атрибут не меняется,

нужно проверить код на стороне фронтенда, который отвечает за изменение состояния чек-бокса.

Далее нужно убедиться в Console или Sources, что события, связанные с чек-боксом, такие как onChange или onClick, корректно отрабатывают. Для этого можно добавить логирование или точки останова, чтобы увидеть, как данные передаются при изменении состояния чек-бокса.

Затем нужно проверить в Network, какой запрос отправляется на сервер при изменении состояния чек-бокса. Нужно убедиться, что запрос содержит правильные данные, такие как payImmediately: true, и что сервер успешно обрабатывает запрос.

Если с запросом всё в порядке, нужно проверить логи сервера или провести отладку серверного кода, чтобы убедиться, что данные правильно сохраняются в базу данных.

4. Возможна ли такая ситуация: отправленный GET-запрос удаляет всю информацию в базе данных? Обоснуй свой ответ.

Теоретически, GET-запрос может удалить информацию в базе данных, но это указывает на серьезные ошибки в архитектуре системы. По стандарту, GET-запросы предназначены только для получения данных и не должны изменять состояние системы. Если сервер настроен неправильно или код содержит ошибки, GET-запрос может вызвать удаление данных.

5. В базе данных есть две таблицы. Какие поля стоит сделать первичными ключами, а какие — внешними? Обоснуй своё решение.

Поле id в таблице employee будет первичным ключом, поскольку оно уникально идентифицирует каждого сотрудника. Поле position_id в этой таблице будет внешним ключом, так как оно связывает сотрудников с их должностями в таблице position. Поле id в таблице position будет первичным ключом, так как оно уникально идентифицирует каждую должность.

6. Посмотри информацию из пятого вопроса и напиши запрос, который выведет данные в таком формате: ФИО сотрудника, название должности и зарплата.

```
SELECT e.fio AS "ФИО сотрудника", p.name AS "Название должности",  
p.salary AS "Зарплата"  
FROM employee e  
JOIN position p ON e.position_id = p.id;
```

7. Код ниже должен проверять, что оператор `//` действительно выполняет деление на целое число. Чего здесь не хватает и почему?

В текущем коде проверка `a == 2` не сообщает об ошибке, если условие не выполняется. Можно использовать `assert`, он будет проверять, что результат деления 5 на 2 равен 2. Если это условие не выполняется, будет выброшено исключение с сообщением. Пример кода:

```
def test_integer_division():
```

```
    a = 5 // 2
```

```
    assert a == 2, "Ошибка: результат деления не равен 2" test_integer_division()
```

8. В какой ситуации классы эквивалентности и граничные значения могут существовать по отдельности? Аргументируй свой ответ и приведи примеры.

Классы эквивалентности и граничные значения могут использоваться по отдельности.

Классы эквивалентности применяются, когда важно протестировать функциональность для групп данных, которые обрабатываются одинаково. Например, при тестировании функции, которая определяет возрастную группу, можно протестировать одну запись из каждого возрастного диапазона.

Граничные значения важны, когда нужно проверить, как программа обрабатывает данные на границах диапазонов. Например, для функции, которая принимает числа от 1 до 100, следует проверить значения 1, 100, а также 0 и 101, чтобы убедиться, что функция правильно работает на границах и за пределами диапазона.

Таким образом, классы эквивалентности полезны для тестирования групп данных, а граничные значения — для проверки обработки данных на границах диапазонов.

9. Можно ли исключить проверку в середине диапазона в пользу проверок на границах, входящих в диапазон? Аргументируй свой ответ.

Исключение проверки в середине диапазона в пользу проверки только на границах может быть оправдано, так как ошибки часто возникают именно на границах диапазона. Например, проблемы могут быть связаны с включением или исключением крайних значений, переполнением или неправильной обработкой крайних случаев.

Однако проверка только на границах не всегда достаточна. Ошибки могут проявляться и в середине диапазона из-за логических и вычислительных ошибок. Например, функция, которая корректно обрабатывает значения 1 и

100, может неправильно работать с числом 50, если логика внутри диапазона неполноценна или ошибочна.

Таким образом, для обеспечения надежности программы важно проводить тестирование как на границах, так и в середине диапазона, чтобы выявить ошибки, которые могут возникать в разных частях диапазона входных данных.

10. Ты тестируешь мобильную версию Яндекс Аренды. Там обнаружился баг: при переходе в раздел «Показать на карте» появляется сообщение «Непредвиденная ошибка», и приложение закрывается. Разработчик просит воспроизвести баг и прислать логи. Как это сделать?

1) Воспроизвести баг:

- ☐ Открыть приложение Яндекс Аренда на мобильном устройстве.
- ☐ Перейти в раздел, где возникает ошибка, выбрав нужный объект или раздел, и нажать на «Показать на карте».
- ☐ Обратить внимание на сообщение «Непредвиденная ошибка» и на то, как приложение закрывается.

2) Собрать логи:

На Android:

- ☐ Включить отладку по USB на устройстве: перейти в «Настройки» -> «О телефоне» -> нажать несколько раз на «Номер сборки», чтобы включить режим разработчика. Затем в «Настройки» -> «Для разработчиков» включить «Отладка по USB».
- ☐ Подключить устройство к компьютеру через USB.
- ☐ Использовать adb (Android Debug Bridge) для сбора логов. Открыть командную строку и выполнить команду:
bash
adb logcat > logcat.txt
- ☐ Воспроизвести ошибку в приложении. Логи будут записываться в файл logcat.txt.

На iOS:

- ☐ Подключить устройство к компьютеру и активировать режим разработчика.
- ☐ Использовать Xcode для получения логов. Открыть Xcode, перейти в меню «Window» далее «Devices and Simulators», выбрать устройство и в разделе «Device Logs» можно увидеть логи приложения.
- ☐ Воспроизвести ошибку в приложении, чтобы логи обновились.

3) Проанализировать логи:


- ☐ Открыть собранные логи (файл logcat.txt для Android или Device Logs в Xcode для iOS) и найти записи, связанные с моментом возникновения ошибки.
- ☐ Искать ключевые слова или фразы, связанные с «Непредвиденной ошибкой» или исключениями.


4) Отправить логи разработчику:

- ☐ Сохранить собранные логи в удобном формате (например, logcat.txt для Android или лог-файл из Xcode для iOS).
- ☐ Прикрепить файл логов к отчету и отправить разработчику для анализа.

Отчёт о тестировании

Функциональное тестирование веб-приложения

Приложение проверено на стенде ([стенд](#)). Все известные требования были покрыты чек-листом:  Чек лист

Результаты выполнения тестов можно посмотреть здесь:  Чек лист . Из 110 успешно прошло 90, не прошло — 18 и 2 невозможно проверить.

Список багов, найденных при тестировании, разбит по приоритетам:

1. Блокирующие:

<https://hisoka123.youtrack.cloud/issue/DR-16>

2. Критичные:

<https://hisoka123.youtrack.cloud/issue/DR-4>

<https://hisoka123.youtrack.cloud/issue/DR-7>

<https://hisoka123.youtrack.cloud/issue/DR-6>

<https://hisoka123.youtrack.cloud/issue/DR-8>

<https://hisoka123.youtrack.cloud/issue/DR-9>

<https://hisoka123.youtrack.cloud/issue/DR-10>

<https://hisoka123.youtrack.cloud/issue/DR-17>

3. Средний приоритет:

<https://hisoka123.youtrack.cloud/issue/DR-2>

<https://hisoka123.youtrack.cloud/issue/DR-3>

<https://hisoka123.youtrack.cloud/issue/DR-5>

<https://hisoka123.youtrack.cloud/issue/DR-11>

<https://hisoka123.youtrack.cloud/issue/DR-12>

<https://hisoka123.youtrack.cloud/issue/DR-13>

<https://hisoka123.youtrack.cloud/issue/DR-14>

<https://hisoka123.youtrack.cloud/issue/DR-15>

<https://hisoka123.youtrack.cloud/issue/DR-18>

<https://hisoka123.youtrack.cloud/issue/DR-19>

Заключение:

В заключении, анализируя текущее состояние веб-приложения, можно сделать вывод о его неготовности к релизу. Проблема, связанная с невозможностью оформления заказа в браузере Google Chrome из-за выявленного бага, представляет серьезную угрозу для функциональности приложения. Учитывая, что более трети пользователей предпочитают использовать именно этот браузер, существует значительный риск потери аудитории и негативного влияния на репутацию и финансовое положение компании.

Кроме того, отсутствие валидации в важных полях может создать возможности для атак с использованием межсетевого скриптинга, что может привести к утечкам пользовательских данных и серьезному нарушению конфиденциальности.

Кроме того, обнаружены проблемы, связанные с неоднозначностью требований к допустимым символам в полях ввода. Например, хотя в требованиях указано, что допускаются символы, такие как тире, точка и запятая, не уточняется, можно ли вводить только символы.

Ретест багов в мобильном приложении

Был проверен фикс багов. Из них не исправлено 1, исправлено — 3, при исправлении появился новый баг. [Баг-репорт](#).

Список багов можно посмотреть здесь: [ссылки](#). У меня не хочет на компе работать YouTrack, поэтому перенес все в Qase

Регрессионное тестирование мобильного приложения по готовым тест-кейсам

Результаты выполнения регрессионных тестов можно посмотреть здесь: [ссылка](#)

Из 10 успешно прошло 1, не прошло — 9.

Список багов, найденных при тестировании, разбит по приоритетам:

1. Блокирующие:

<https://hisoka123.youtrack.cloud/issue/DR-20>

<https://hisoka123.youtrack.cloud/issue/DR-21>

2. Критичные:

<https://hisoka123.youtrack.cloud/issue/DR-22>

<https://hisoka123.youtrack.cloud/issue/DR-23>

<https://hisoka123.youtrack.cloud/issue/DR-24>

<https://hisoka123.youtrack.cloud/issue/DR-25>

3. Средний приоритет:

<https://hisoka123.youtrack.cloud/issue/DR-26>

4. Низкий приоритет:

<https://hisoka123.youtrack.cloud/issue/DR-27>

<https://hisoka123.youtrack.cloud/issue/DR-28>

Заключение:

Программа в незавершенном состоянии и имеет критические ошибки, которые существенно нарушают ее работоспособность. Из 10 выполненных тестов успешным оказался лишь один, что свидетельствует о крайне низкой стабильности и надежности. Ключевые проблемы включают серверную ошибку при попытке принять отмененный заказ и отсутствие уведомления об ошибке при принятии заказа, уже занятого другим курьером. В текущем виде программу нельзя пускать в релиз, и необходимо устранить эти баги до выхода на рынок.

Выводы о проделанной работе

Первая часть не показалась мне сложной, даже после 5-месячного перерыва было легко вернуться к работе. Единственная трудность возникла при попытке восстановить аккаунт в YouTrack — обратился в поддержку, но пока ответа нет. Устную часть еще не сдавал, нужно немного освежить теорию. Руки помнят, а голова подзабыла. Надеюсь, что все пройдет успешно, и я смогу спокойно завершить работу над второй частью и защитить диплом.