**Задание 1.** ТЗ от заказчика:**"Реализовать форму, которая по введенным данным определяет, является ли человек совершеннолетним. Приложение должно быть с архитектурой клиент-сервер"**.

**Вопросы:**

1. Какие могут возникнуть вопросы к требованиям?
2. Какие виды и типы тестирования стоит проводить? Объясните, почему.
3. Что больше подойдет для организации тестовой документации по задаче - тест-кейсы или чек-листы? Почему?
4. Какие техники тест-дизайна Вы бы использовали? Почему?
5. Какие можно выделить негативные и позитивные входные данные?
6. Какие теоретически могут быть баги? Опишите 4-6 возможных бага.
7. Есть ли баги любого типа на приведенных скриншотах? Если есть, опишите все.

**Ответы:**

1. Работает ли система корректно (что пишет в ответ), если вводить:

* А) спецсимволы («№%: лет)
* Б) буквы (РАИАОРА лет)
* В) отрицательные значения (-10 лет, -35 лет)
* Г) значения, которые больше, чем реальные года у человека (100 лет, 1000000000 лет)

1. Дымовое тестирование

Функциональное тестирование

Нагрузочное тестирование

Необходимо проверить, как реагирует система при полной загрузке сайта (когда запрос делается со 100 компьютеров, 1000 компьютеров и т.д.), также проверить нужно скорость ответа на запрос и загрузки сайта. Также необходимо создать чек-лист для проверки различных запросов формы, провести их и если есть багги, то завести баг-репорты.

1. Подойдут больше тест-кейсы, потому что они создаются, тестируются и быстрее и можно сразу показать отчёт для баг-репортов.
2. Я бы использовал «граничные значения» и «попарное тестирование», так как в попарном тестировании будет идти сравнение введённых данных с данными для подтверждения совершеннолетия, а «граничные значения» нужны для определения минимальных и максимальных вводимых значений
3. **Негативные данные:**

Я описывал их в пункте 1):

* А) спецсимволы («№%: лет)
* Б) буквы (РАИАОРА лет)
* В) отрицательные значения (-10 лет, -35 лет)
* Г) значения, которые больше, чем реальные года у человека (100 лет, 1000000000 лет)

**Позитивные данные:**

Все остальные «цифры», начиная от 1 года до 17 лет включительно (несовершеннолетный) и заканчивая от 18 лет до 99 лет (совершеннолетный)

1. Баги:

* А) спецсимволы («№%: лет)
* Б) буквы (РАИАОРА лет)
* В) отрицательные значения (-10 лет, -35 лет)
* Г) значения, которые больше, чем реальные года у человека (100 лет, 1000000000 лет)

1. А) На скриншоте написано «введите цифру», то есть одну цифру, а цифр может быть и две, поэтому нужно исправить на «введите цифры» либо «введите значение».

Б) Слово «совершеннолетний» пишется слитно, а не как на скриншоте «совершенно-летний»

## Задание 2. Составить чек-лист проверок для <https://sbercloud.ru/ru>

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| № | Приоритет | Проверка | Результат |
| 1 | Средний | Зайти на страницу https://sbercloud.ru/ru | Страница открылась |
| 2 | Средний | Поменять язык RU\EN в левом верхнем углу нажав на иконку | Язык меняется с русского на английский и обратно |
| 3 | Средний | Ввести в "поиск" данные для поиска | Данные ввелись, поиск работает |
| 4 | Средний | Нажать на вкладку "о нас" и посмотреть | Вкладка нажимается, появляется информация о проекте |
| 5 | Средний | Проверить вкладку "Enterprise" и все её пункты на работоспособность | Вкладка "Enterprise" открывается, все подпункты в ней открываются |
| 6 | Средний | Проверить вкладку "Advanced" и все её пункты на работоспособность | Вкладка "Advanced" открывается, все подпункты в ней открываются |
| 7 | Средний | Проверить вкладку "ML Space" и все её пункты на работоспособность | Вкладка "ML Space" открывается, все подпункты в ней открываются |
| 8 | Средний | Проверить вкладку "Решения" и все её пункты на работоспособность | Вкладка "Решения" открывается, все подпункты в ней открываются |
| 9 | Средний | Проверить вкладку "Цены" и все её пункты на работоспособность | Вкладка "Цены" открывается, все подпункты в ней открываются |
| 10 | Средний | Проверить вкладку "Консультация" | Вкладка "Консультация" открывается, появляется форма для заполнения данных на консультацию |
| 11 | Средний | Проверить вкладку "Вход" | Вкладка "Вход" открывается, система просит ввести логин и пароль |
| 12 | Средний | Проверить вкладку "Регистрация" | Вкладка "Регистрация" открывается, система просит ввести данные для регистрации |
| 13 | Средний | Проверить вкладку "Архитектурный центр" | Вкладка "Архитектурный центр" открывается, появляется описание об архитектурном центре |
| 14 | Средний | Проверить вкладку "Безопасность" | Вкладка "Безопасность" открывается, появляется описание о безопасности в облаке |
| 15 | Средний | Проверить вкладку "Комьюнити" и все её пункты на работоспособность | Вкладка "Комьюнити" открывается, все подпункты в ней открываются |
| 16 | Средний | Проверить вкладку "Информация" и все её пункты на работоспособность | Вкладка "Информация" открывается, все подпункты в ней открываются |
| 17 | Средний | Проверить вкладку "Документация" | Вкладка "Документация" открывается, появляется информация о сервисах "cloud" |
| 18 | Средний | Проверить информационные вкладки | Вкладки листаются, открываются |
| 19 | Средний | Проверить вкладки категории "Решения, которые нужны прямо сейчас" | Вкладки "Ретейл", "IT-разработка", "Транспорт и логистика", "Медиа и реклама", "Здравоохранение" и "Государство" открываются при нажатии на кнопку "подробнее" |
| 20 | Средний | Проверить вкладку "наши клиенты" | Вкладка открывается при нажатии на кнопку "все кейсы" |
| 21 | Средний | Проверить вкладки "блог" | Вкладки листаются, открываются |
| 22 | Средний | Проверить вкладку "Запросите бесплатную консультацию по вашему проекту" | Вкладка открывается при нажатии на кнопку "оставить заявку" |
| |  | | --- | | 23 | | Средний | Проверка пункта "Подпишитесь на дайджест" | Почта в окно вводится, ставится галочка напротив формы "даю согласие на отправку мне информационных и рекламных сообщений" и при нажатии на кнопку подписаться происходит подписка на уведомления |

## 

Задание 3. **API Яндекс.Погоды позволяет получать погодные данные в автоматизированном режиме. Объем предоставляемых данных зависит от  https://yandex.ru/dev/weather/doc/dg/concepts/pricing.html#about**

**Необходимо с помощью Postman протестировать работу метода со следующими параметрами:**

* тариф Тестовый
* lon 50
* lat - 55
* язык русский

Также создать несколько негативных сценариев и убедиться, что при неправильных данных запрос возвращает соответствующий результат

**РЕШЕНИЕ:**

Приложенный файл «Задание 3. Погода Яндекс.postman\_collection.json» в ответном письме

Задание 4. Есть вот такие вводные данные:



Ответь, используя запросы SQL, на следующие вопросы:

1. сколько всего животных у Вани
2. имена всех кошек
3. Уникальные имена всех хозяев, отсортированные по алфавиту

1.

SELECT animal\_classes.id, animal\_classes.class

COUNT(\*)

FROM animal\_info.name, animal\_info.class, animal\_info.owner

INNER JOIN animal\_info.owner

ON animal\_classes.id = animal\_info.class;

**У Вани 2 животных**

2.

SELECT animal\_classes.id, animal\_classes.class

COUNT(\*)

FROM animal\_info.name, animal\_info.class, animal\_info.owner

WHERE animal\_classes.id = animal\_info.class = 1

GROUP BY animal\_info.class;

**Имена: Кити, Мити**

3.

SELECT animal\_classes.id, animal\_classes.class

COUNT(\*)

FROM animal\_info.name, animal\_info.class, animal\_info.owner

WHERE animal\_classes.id = animal\_info.class

GROUP BY animal\_info.owner

ORDER BY COUNT DESC;

**1.Ваня**

**2.Петя**