МИНИСТЕРСТВО ОБРАЗОВАНИЯ И МОЛОДЕЖНОЙ ПОЛИТИКИ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГАПОУ СО «ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ДОПУЩЕН К ЗАЩИТЕ:

РУКОВОДИТЕЛЬ РАБОЧЕЙ ГРУППЫ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Софьина Н.А.)

ЗАЩИЩЕН: ПРОТОКОЛ\_\_\_\_\_\_ОТ\_\_\_\_\_\_\_ ПРЕДСЕДАТЕЛЬ ГАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Новокрещенов С.А.)

**ДИПЛОМНЫЙ ПРОЕКТ**

ТЕМА: «Создание тест-кейсов на JS»

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 9 | 0 | 2 | 0 | 7 | - ДП - | 4 | 5 | 0 | 3 | – | И | П | 8 | 1 | К | – | 1 | 8 |

ОЦЕНКА: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

РУКОВОДИТЕЛЬ ДИПЛОМНОГО ПРОЕКТА:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Софьина Н.А.)

СТУДЕНТ ГР. ИП-81К

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Кондратьев Н.А)

КОНСУЛЬТАНТ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Грибова А.В.)

РЕЦЕНЗЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

|  |
| --- |
| 2022 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № стр. | Формат | Обозначение | Наименование | Кол.  листов | № экз. | Примеч. |
| 1 | А4 | ЕМК 09.02.07. ДП 4503-ИП-81к-18 ПЗ | Пояснительная записка | 56 | - |  |
| 2 |  | ЕМК 09.02.07.ДП 4503-ИП-81к-18 ИС | « Создание тест-кейсов на JS» | 1 |  | USB Flash |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |
| 21 |  |  |  |  |  |  |
| 22 |  |  |  |  |  |  |
| 23 |  |  |  |  |  |  |
| 24 |  |  |  |  |  |  |
| 25 |  |  |  |  |  |  |
|  | | | | | | |

СОДЕРЖАНИЕ

Оглав

[ВВЕДЕНИЕ 4](#_Toc106021087)

[РАЗДЕЛ 1. АНАЛИТИЧЕСКАЯ ЧАСТЬ 5](#_Toc106021088)

[РАЗДЕЛ 2. ПРОЕКТНАЯ ЧАСТЬ 8](#_Toc106021089)

[РАЗДЕЛ 3. ТЕСТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ 23](#_Toc106021090)

[3.1 Тестирование интерфейса пользователя 23](#_Toc106021091)

[РАЗДЕЛ 4. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 27](#_Toc106021092)

[ЗАКЛЮЧЕНИЕ 47](#_Toc106021093)

[СПИСОК ЛИТЕРАТУРЫ 48](#_Toc106021094)

[ПРИЛОЖЕНИЕ 1 Образец страницы приложения. 50](#_Toc106021095)

[ПРИЛОЖЕНИЕ 2 Список тест-кейсов 51](#_Toc106021096)

# ВВЕДЕНИЕ

На сегодняшний момент модульное тестирования ПО и Web ресурсов играет важную роль. Использование тестирования позволяет контролировать процесс разработки.

Тестирование ПО в настоящее время широко используются во всех звеньях IT сфер. Тест-кейс — это такое описание проверки работы системы, которое может выполнить любой человек из команды, будь то тестировщик, разработчик, аналитик или даже бизнес-заказчик. Тест-кейсы обеспечивают эффективное тестирование заданных или не заданных функций, в которой сбор и переработка информации, необходимой для реализации/исправлений функций работы ПО.

Цель проекта – создание тест-кейсов, а точнее создание «тестовой песочницы», на которой можно будет поупражняться в поиске багов и составлению тест-кейсов. Проект может быть использован при обучении тестировщиков. Разрабатываемая «тестовая песочница» может служить практическим материалом при изучении дисциплины «Тестирование информационных систем» .

Также данный проект может быть использован компаниями как вступительное задание. Человек должен сделать тесты для прохождения всего задания за определённый срок от этого будет зависеть поступит на должность или нет.

В данной работе представлены задачи, предназначенные для поиска тестовых наборов. Реализация проекта выполнена с помощью языка Java Script.

В итоге в информационной системе представлено:

* HTML, CSS разметка сайта.
* JS скрипты.

# АНАЛИТИЧЕСКАЯ ЧАСТЬ

* 1. Общее описание системы

Система была создана для начинающих тестировщиков или людей которые хотят попробовать что-то новенькое. Для того чтобы они упражнялись в данной песочнице. Создавали, тестировали, модифицировали и изменяли свои тесты, делали их более адаптивными и универсальными.

В данной системе представлено 8 тестовых заданий. В которых разное количество кейсов и разная тематика.

* 1. Цели, функции и задачи системы

Функционал заключается в проверке правильности предоставленного ответа.

Цель: Задача состоит в том, чтобы пользователь или программа открыла все кейсы данного теста для перехода на следующий.

Проект состоит из 4 основных модулей:

1. Модуль загрузки данных (Если пользователь до этого проходил тест, но не прошёл его до конца)
2. Модуль проверки (В нём вся логика кейсов)
3. Модуль проверки на прохождения теста (Если пользователь прошёл тест до конца, то кнопка 'Следующий тест' становиться доступной для нажатия)
4. Модуль сохранения данных (В данном модуле сохраняется весь прогресс теста)
   1. Описание предметной области системы

Данный тренажер рассчитан на то, что тестируемый сможет подбирать разные варианты кейсов при работе с заданной формой. Естественно, что при работе формы необходимо учесть не только самые популярные кейсы, но и все варианты ввода данных пользователем. Данная тестовая песочница должна включать все перечисленные в задании тест-кейсы. Разрабатываемые тест-кейсы должны учитывать тот факт, что каждый тест представляет собой класс эквивалентности. Класс эквивалентности – это наборы, с которыми тестируемая форма должна работать одинаково. Например, тест на проверку равносторонний треугольник или нет (тест 4 задание 1) должен одинаково для треугольника со сторонами 3, 3, 4 и для треугольника со сторонами 5,5,2

Количество задач «тестовой песочницы» - 8. Тестовая песочница должна иметь стартовую страницу и 8 страниц для обучения (под каждую задачу – отдельная страница) Образец первой страницы «тестовой песочницы» представлен в ПРИЛОЖЕНИИ 1. Набор заданий для «тестовой песочницы» представлен в ПРИЛОЖЕНИИ 2.

* 1. Модель информационной системы

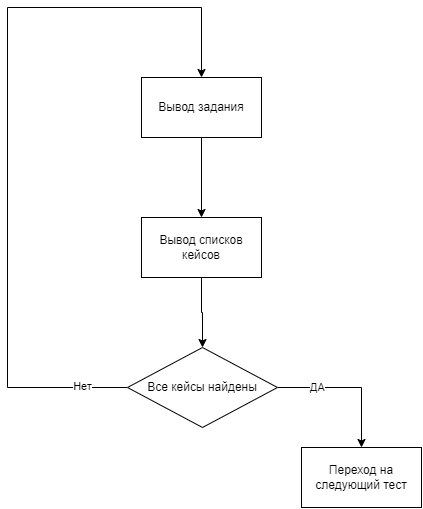


Рисунок 1. Модель информационной системы

* 1. Руководство по стилю



Рисунок 2. Руководство по стилю

РАЗДЕЛ 2. ПРОЕКТНАЯ ЧАСТЬ

Скрипт для открытия тестовых кейсов и очистка

хранилища данных представлена на рис. 3

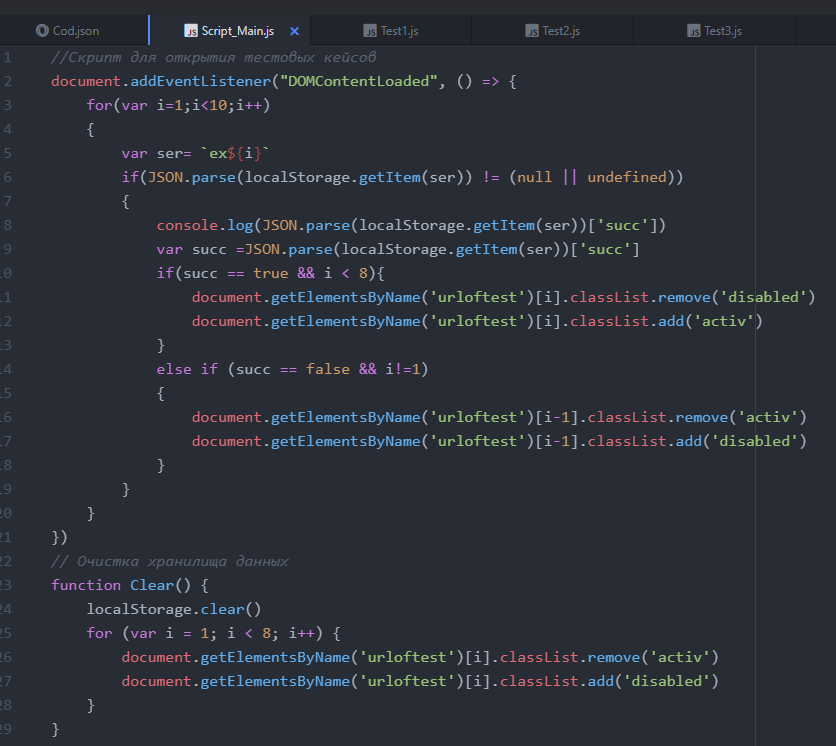


Рисунок 3. Скрипт для открытия тестовых кейсов и очистка

хранилища данных

Загрузка кейсов которые были открыты и основной скрипт проверки представлен на рис. 4,5,6,7,8,9

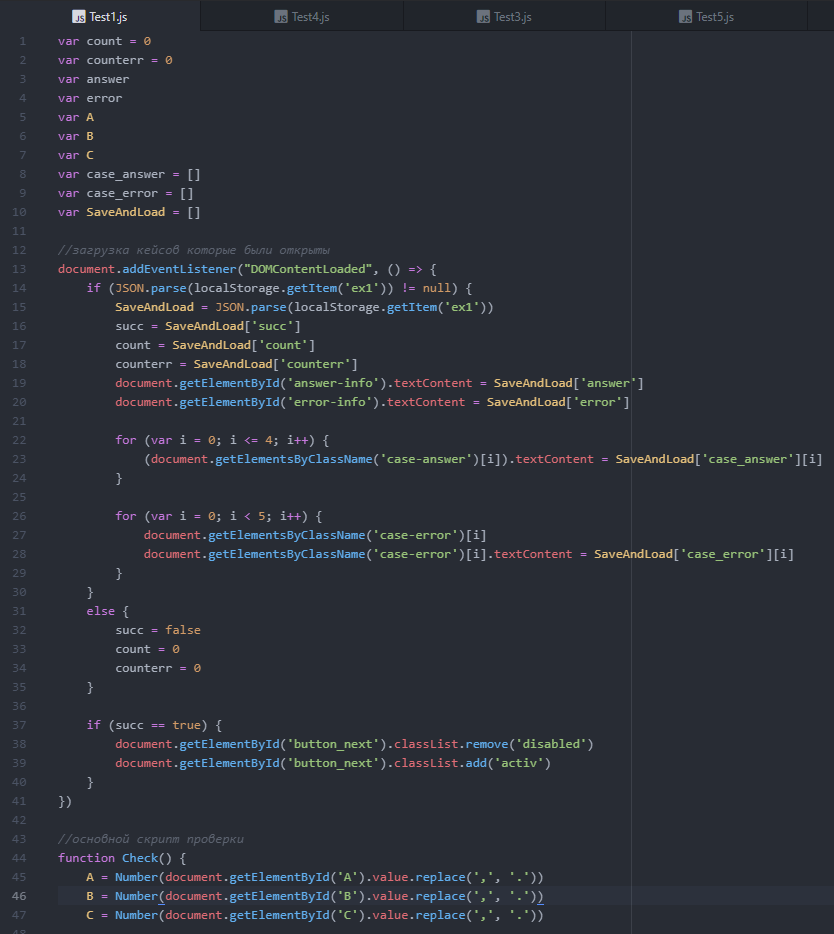


Рисунок 4. Загрузка кейсов которые были открыты и основной скрипт проверки



Рисунок 5. Загрузка кейсов которые были открыты и основной скрипт проверки

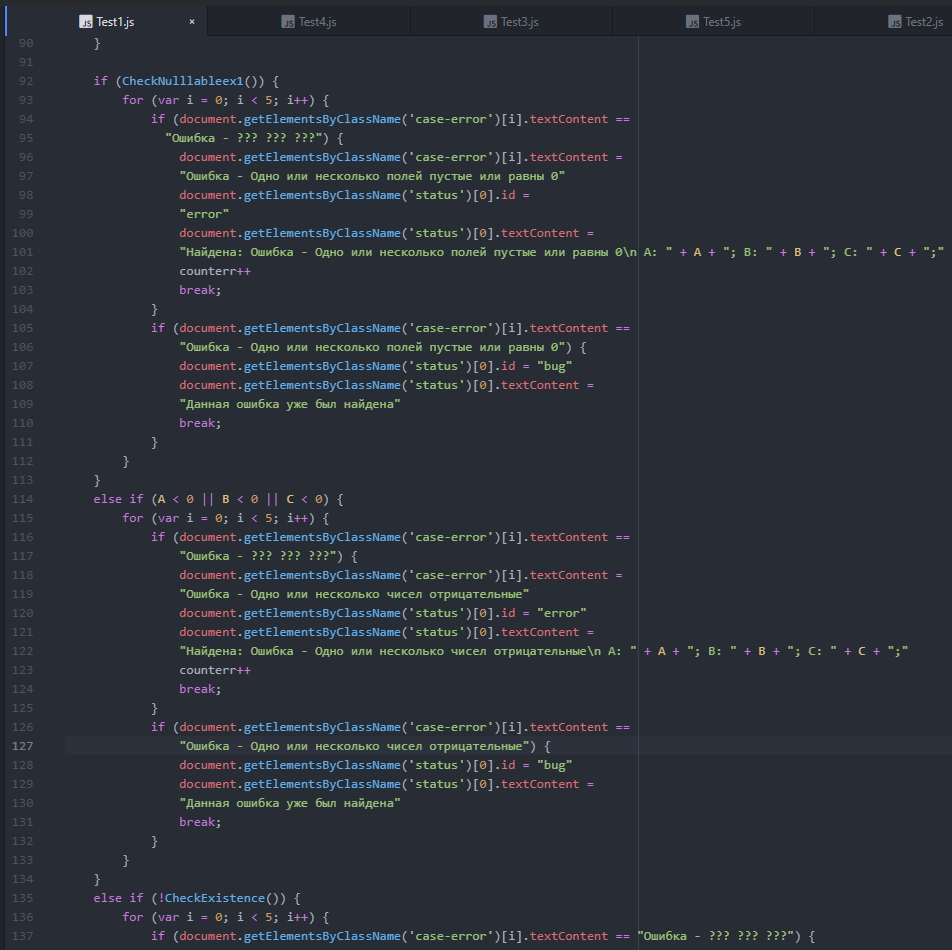


Рисунок 6. Загрузка кейсов которые были открыты и основной скрипт проверки

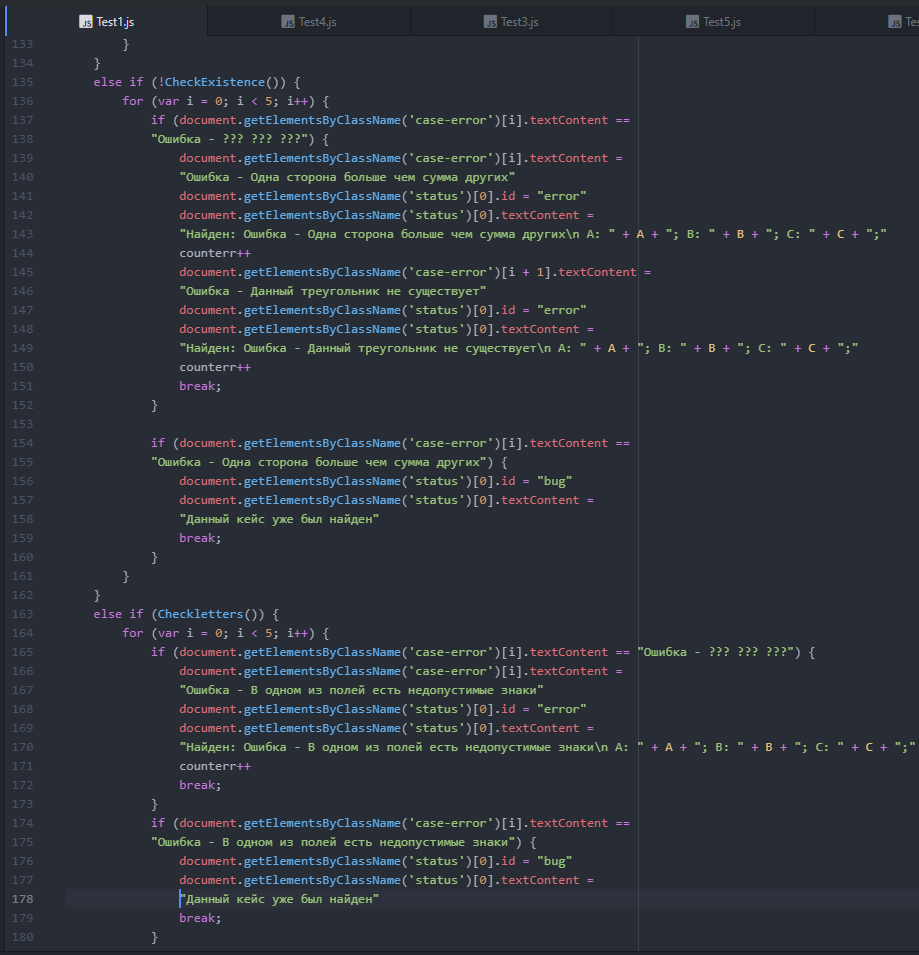


Рисунок 7. Загрузка кейсов которые были открыты и основной скрипт проверки



Рисунок 8. Загрузка кейсов которые были открыты и основной скрипт проверки

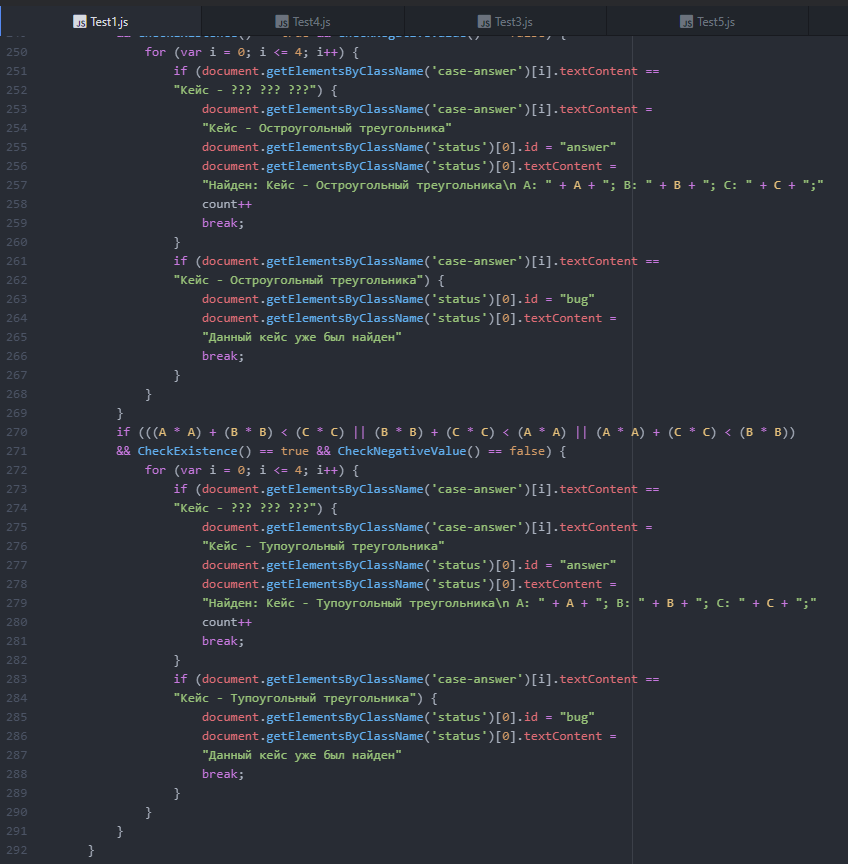


Рисунок 9. Загрузка кейсов которые были открыты и основной скрипт проверки

Сохранение кейсов представлен на рис. 10



Рисунок 10. Сохранение кейсов

Загрузка кейсов которые были открыты представлен на рис. 11

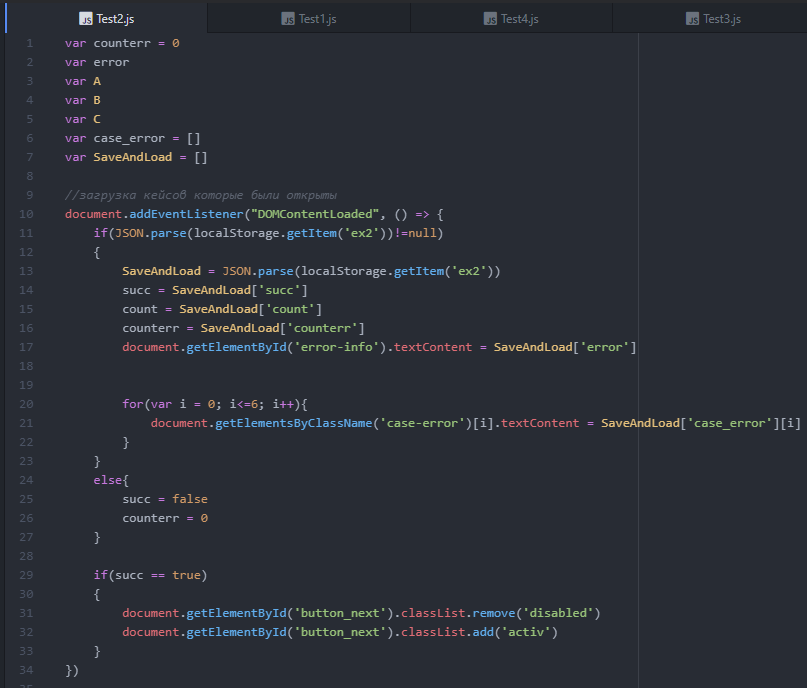


Рисунок 11. Загрузка кейсов которые были открыты

Основной скрипт проверки представлен на рисунках ниже



Рисунок 12. Основной скрипт проверки

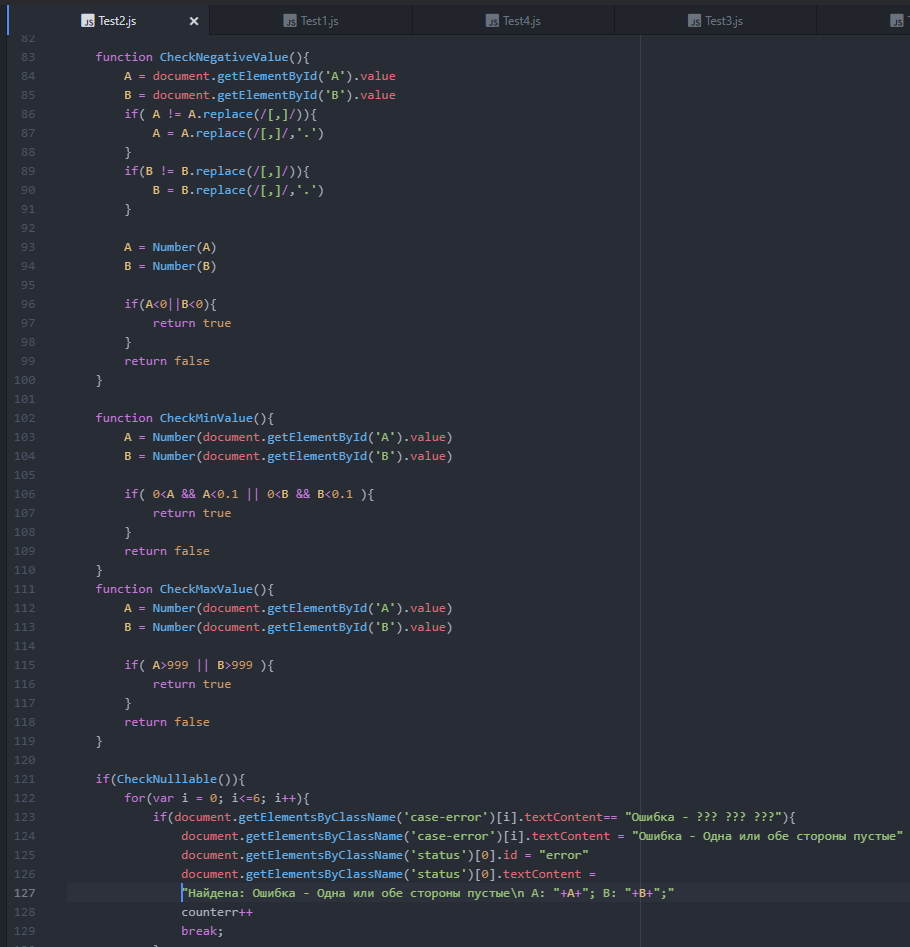


Рисунок 13. Основной скрипт проверки

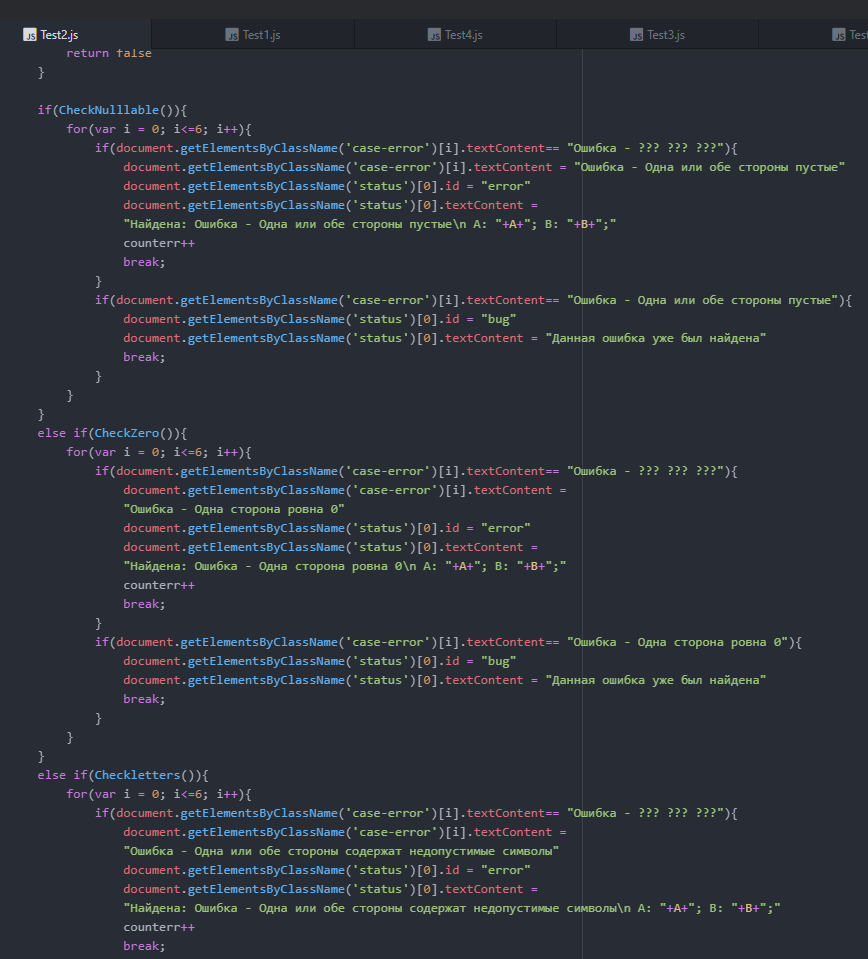


Рисунок 14. Основной скрипт проверки



Рисунок 15. Основной скрипт проверки

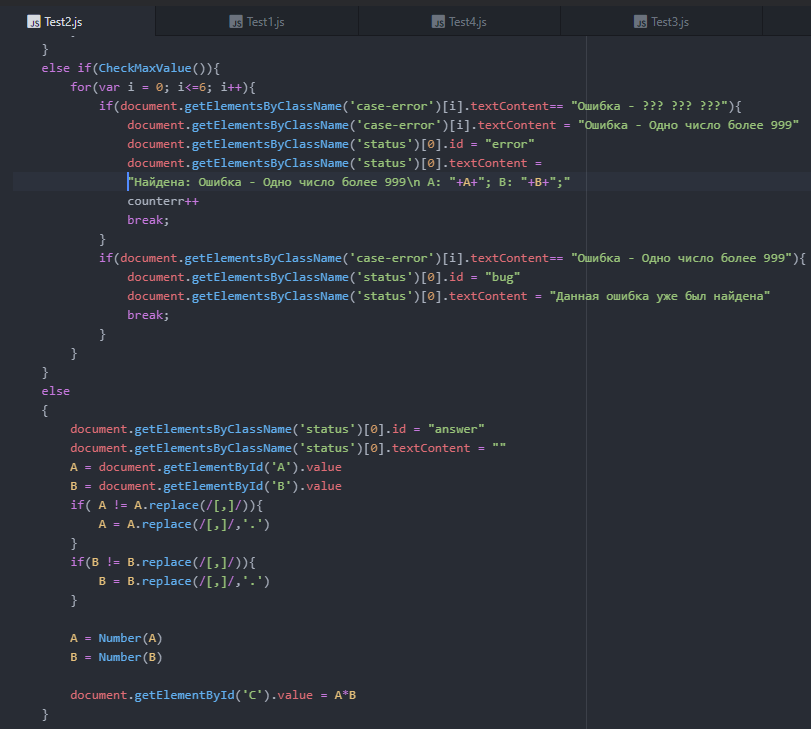


Рисунок 16. Основной скрипт проверки

Сохранение кейсов представлен на рис. 17



Рисунок 17. Сохранение кейсов

РАЗДЕЛ 3. ТЕСТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Тестирование сайта – один из важных жизненных этапов, после которого Исполнитель предоставляет Заказчику готовый проект без ошибок, с хорошей читабельностью, воспринимаемой легкостью, удобством и надежностью.

Тестирование – это отклонение фактического результата от ожидаемого, другими словами – это процесс поиска багов (ошибок).

Основные правила тестирования Web-сайтов – это шаги, которые показывают пользователю, насколько удобен и логичен будет проект, насколько просто и возможно найти ту или иную информацию, хорошо ли воспринимается человеческому взгляду и правильно ли работает весь функционал данного сайта, представленный в техническом задании.

Главная задача тестирования заключается в том, чтобы как Заказчику, так и пользователю, было предоставлено качество продукта.

## 3.1 Тестирование интерфейса пользователя

Тестирование пользовательского интерфейса – это тестирование графического интерфейса пользователя, которое предполагает проверку сайта на соответствие требованиям к графическому интерфейсу, профессионально ли оно выглядит, выполнено ли оно в едином стиле.

Тестирование интерфейса пользователя осуществляется вместе со следующими видами тестирования:

– тестирование на соответствие стандартам графических интерфейсов;

– тестирование с различными разрешениями экрана (адаптивность);

– тестирование кроссбраузерности (совместимости сайта в разных браузерах);

– тестирование графического интерфейса пользователя на различных устройствах.

При тестировании системы была проведена проверка адаптивности всех страниц сайта, одна из которых представлена на рисунках 18-20

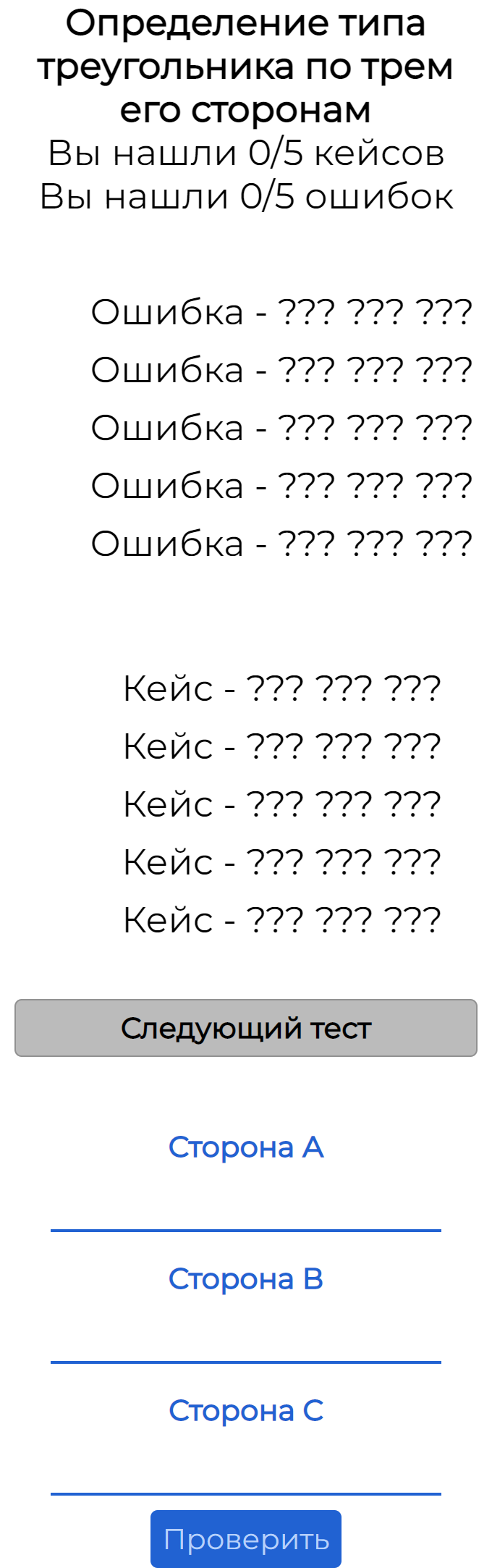


Рисунок 18.

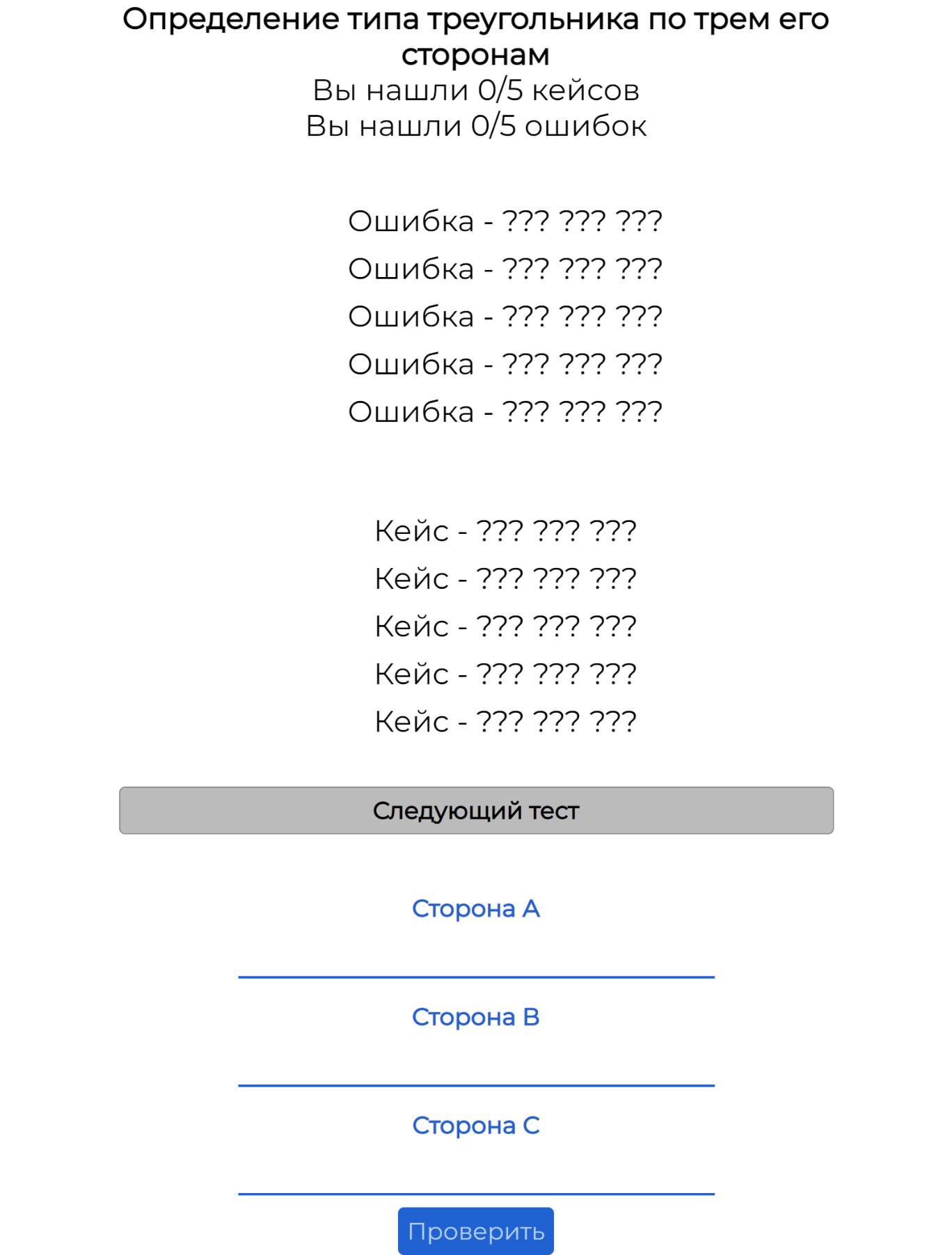


Рисунок 19.

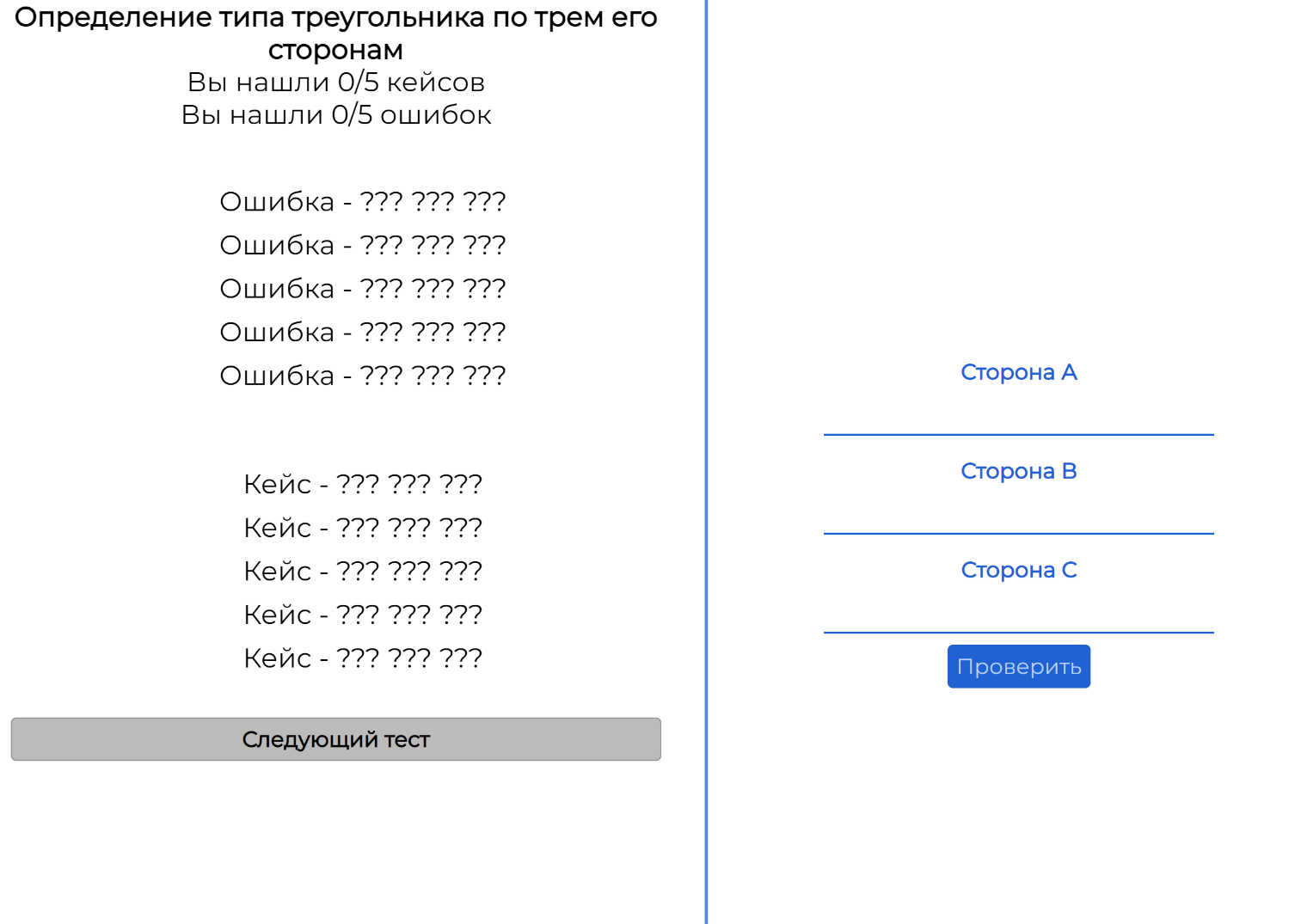


Рисунок 20.

РАЗДЕЛ 4. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Стартовая страница сайта находится по адресу: <http://188.234.244.32/0503tests/>

А также может сам попробовать пройти эти тесты.

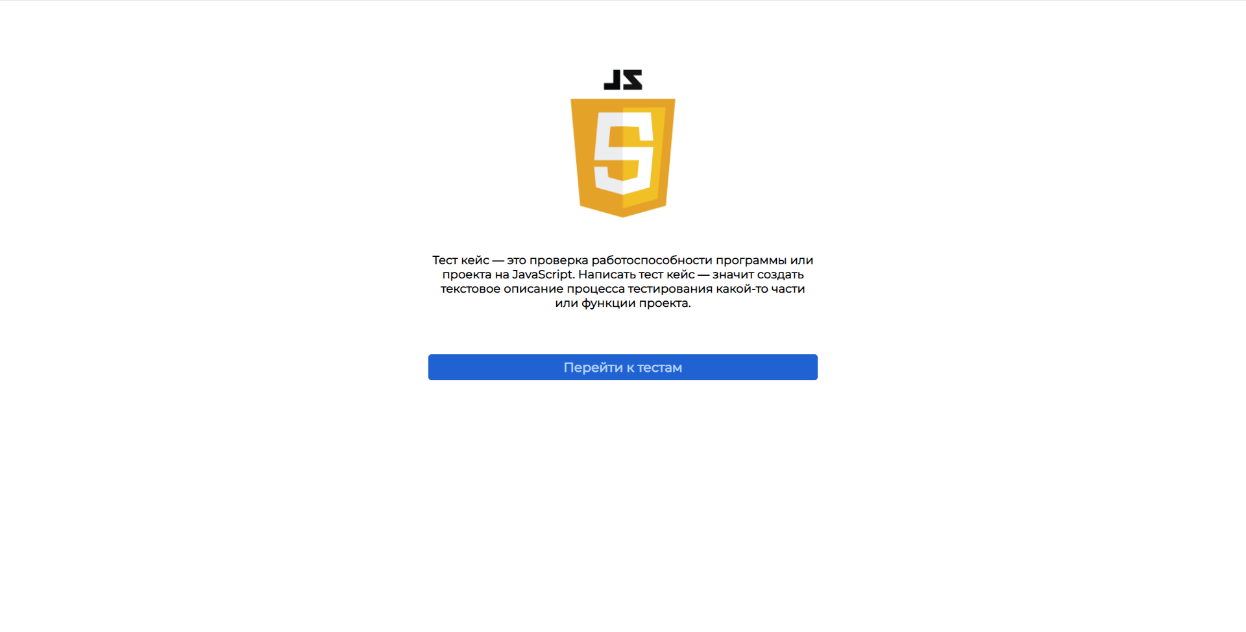


Рисунок 21 Главная страница

После нажатия “Перейти к тестам” пользователь переходит на странице со списком тестов

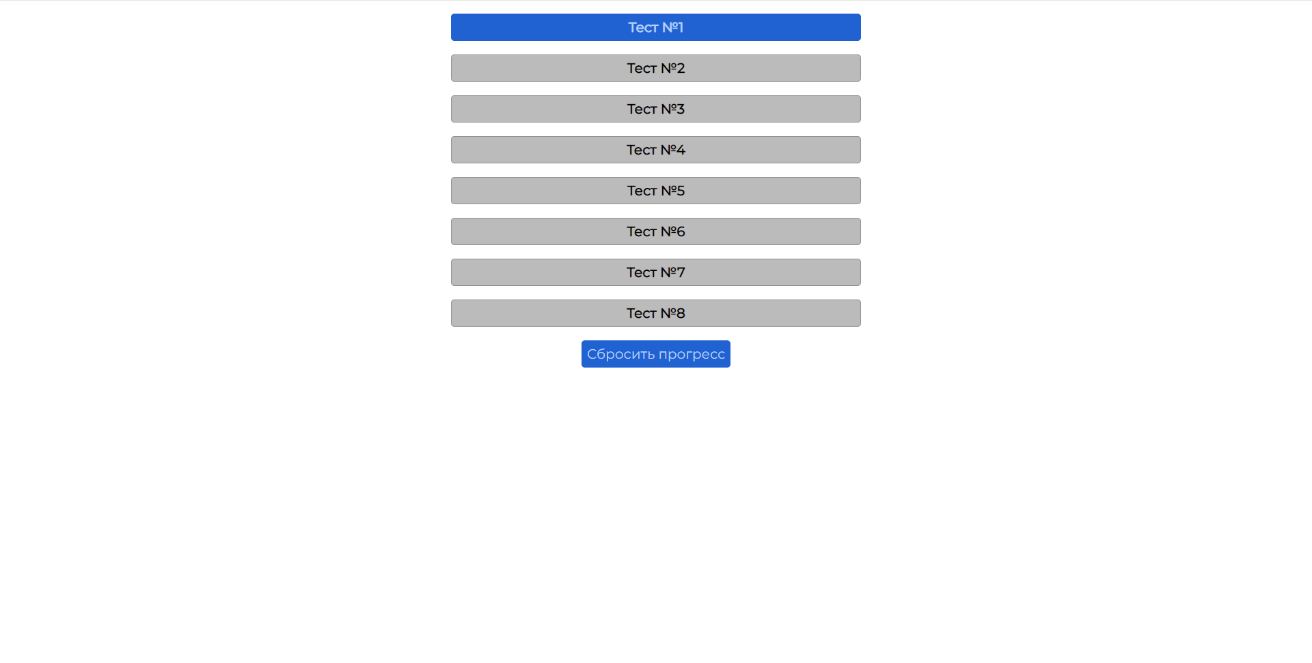


Рисунок 22

На данной странице будет приходить основная навигация по тестовым кейсам, а также взаимодействия с ними.

**Заходя на один из тестов**, в пример берется 1-ый тест-кейс “Определение типа треугольника по трем его сторонам“. (Рис. 23)

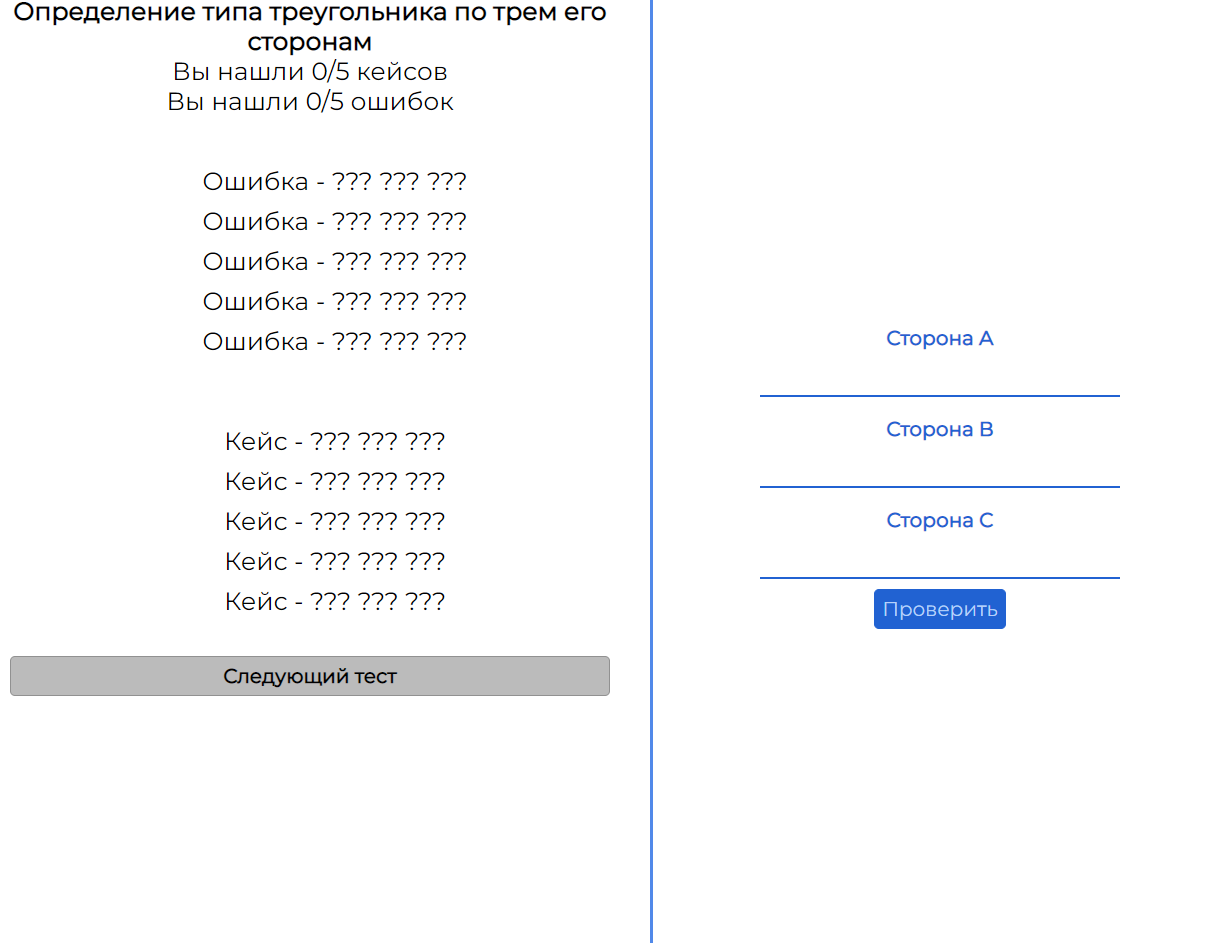


Рисунок 23

К примеру, в данном тест-кейсе пользователь может протестировать:

1. Нахождение определения типа треугольника по введённым пользователем значений.
2. Проверка на ввод именно числовых значений.
3. Проверка на запрещенные/некорректные символы.

Для проверки работы тест-кейса необходимо ввести данные, после ввода каждой нажимать кнопку “Проверить”.

Правильные:

1. 2 , 3 , 4 - Остроугольный треугольник
2. 2 , 3 , 4 - Тупоугольный треугольника
3. 3 , 3 , 4 - Равнобедренный треугольник
4. 3 , 3 , 3 - Равностороннего треугольника
5. 3 , 4 , 5 - Прямоугольный треугольника

Не правильные:

1. 3 , 3 , 1234 - Данный треугольник не существует
2. 3 , 3 , 1234 - Одна сторона больше чем сумма других
3. 3 , -3 , 3 -Одно или несколько чисел отрицательные
4. 3 , 3 , % - В одном из полей есть недопустимые знаки
5. , 3 , 3 - Одно или несколько полей пустые или равны 0

Результат (Рис. 24)

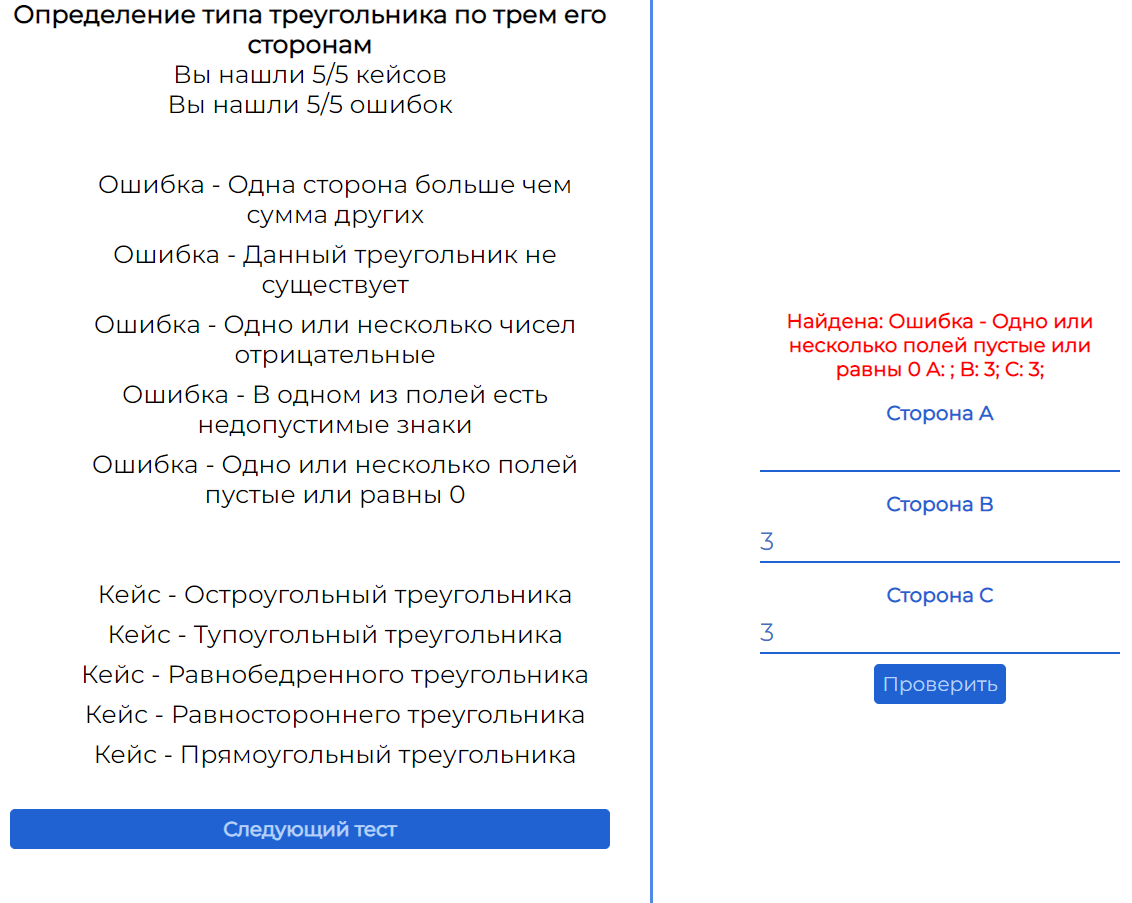


Рисунок 24. Результат

**Второй тест-кейс** “Найти площадь треугольника” (Рис. 25)

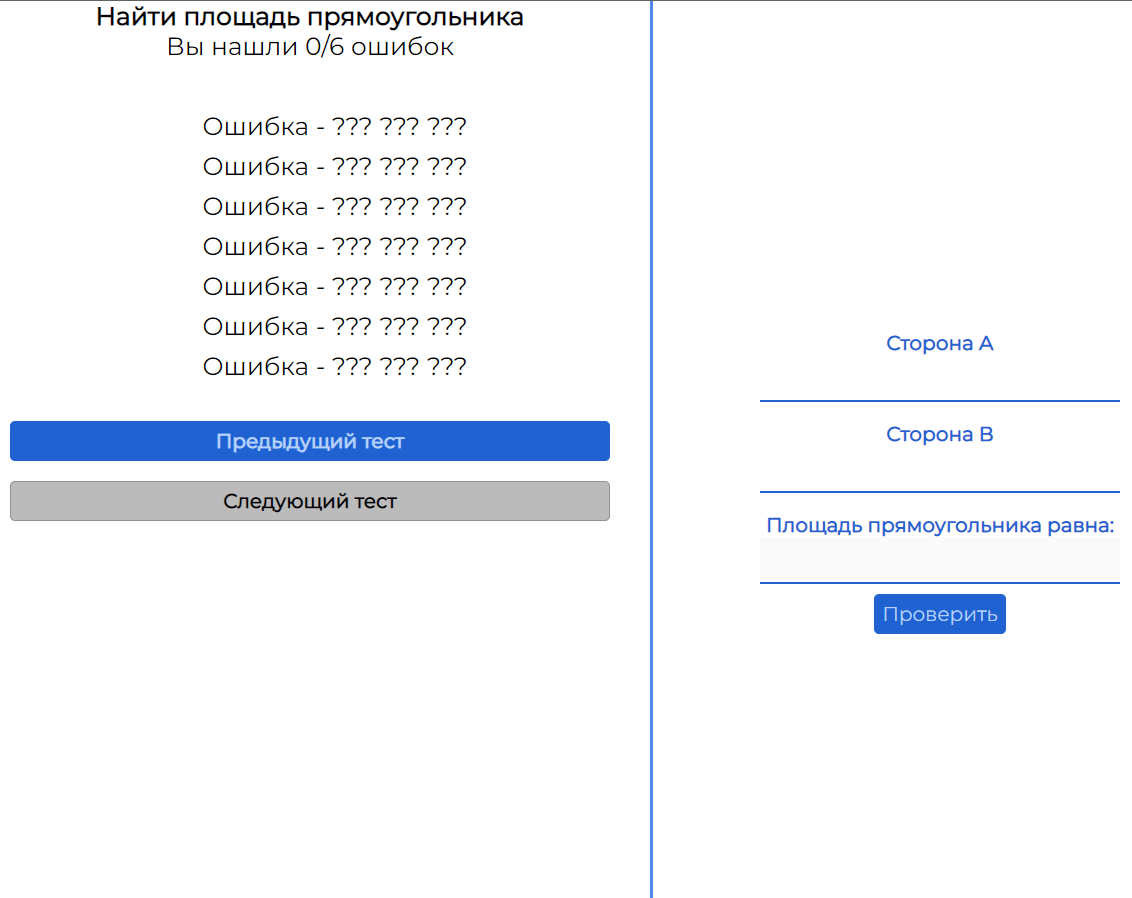


Рисунок 25. Найти площадь треугольника

Для проверки работы тест-кейса необходимо ввести данные для верной проверки

Правильные: “A: 15, B: 20”

Не правильные:

1. 3 , 0.03 - Одно число менее 0.1
2. 3 , 1000 - Одно число более 999
3. 0 , 3 - Одна сторона ровна 0
4. , 3 - Одна или обе стороны пустые
5. -3 , 3 - Одно число отрицательное
6. % , 3 - Одна или обе стороны содержат недопустимые символы
7. -3.3 , 3 - Одно число дробное и отрицательное

Результат правильного (Рис. 26)

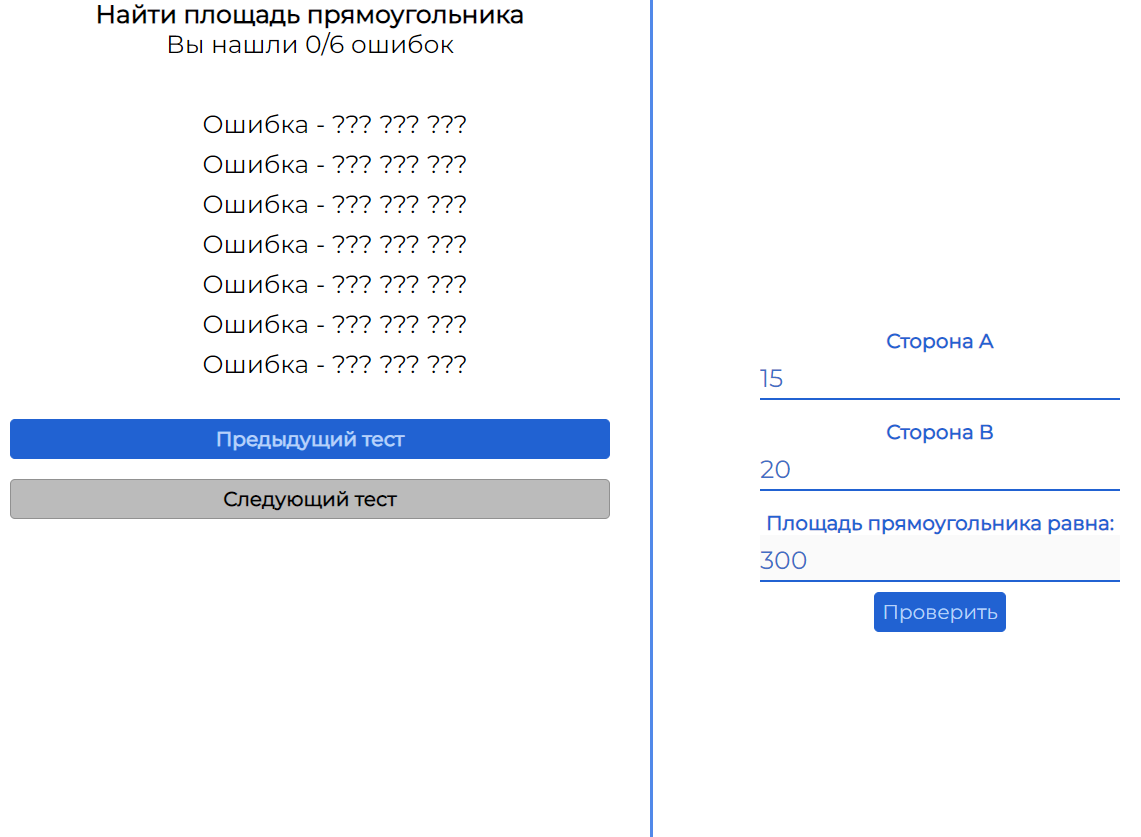


Рисунок 26. Результат правильного

Результат не правильных (Рис. 27)

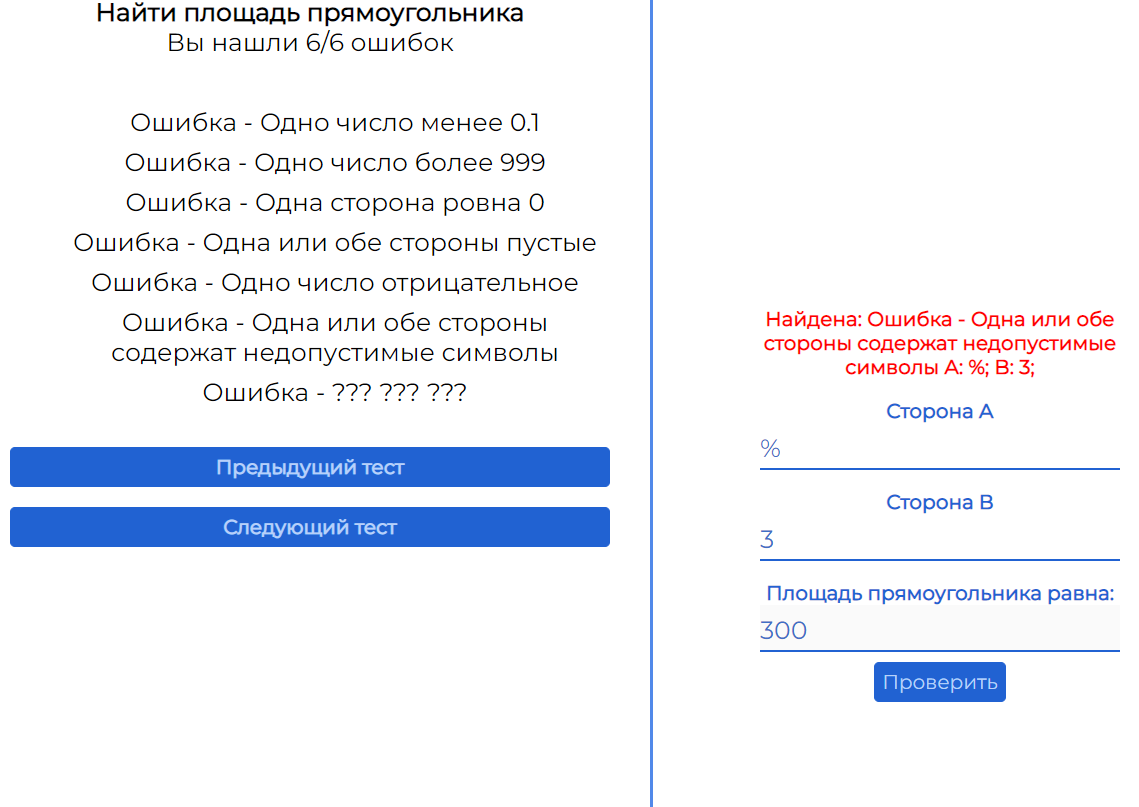


Рисунок 27. Результат не правильных

**Третий тест-кейс.** Строка вида "N/M", где N - первое число, M - второе число.  
Так получилось, что в некоторых ситуациях значение N превышает значение M. Реализуйте метод, который бы в случае, когда N > M поменял бы первое число на второе. (Рис. 28)

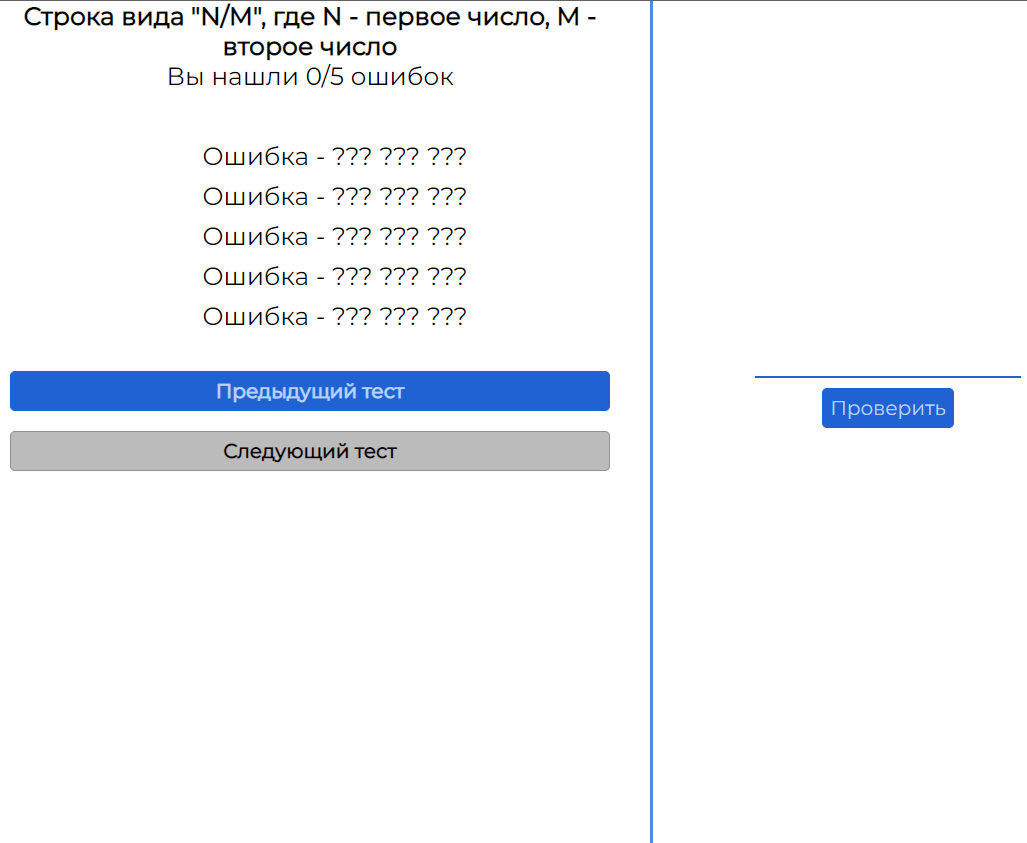


Рисунок 28. Третий тест-кейс

Для проверки работы тест-кейса необходимо ввести данные:

Правильные: 10/20 (Рис. 29)

Не правильные: (Рис. 30)

1. 10/20% - Наличие символов или букв
2. 10/20/ - Лишние разделители
3. - Пустая строка
4. Любой символ
5. Просто число

Результат правильных (Рис. 29)

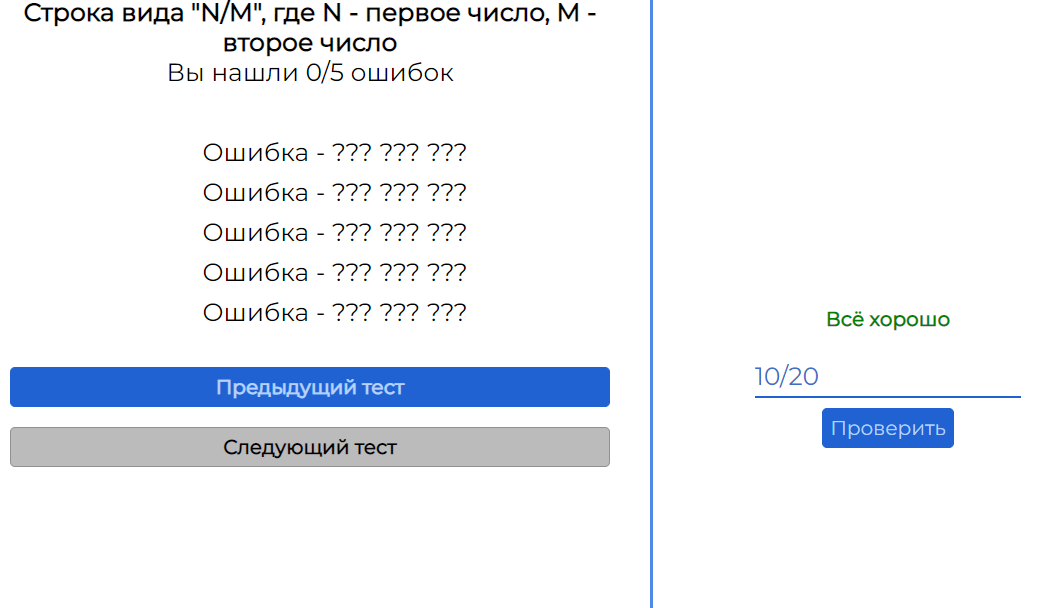


Рисунок 29. Правильные

Результат не правильных (Рис. 30)

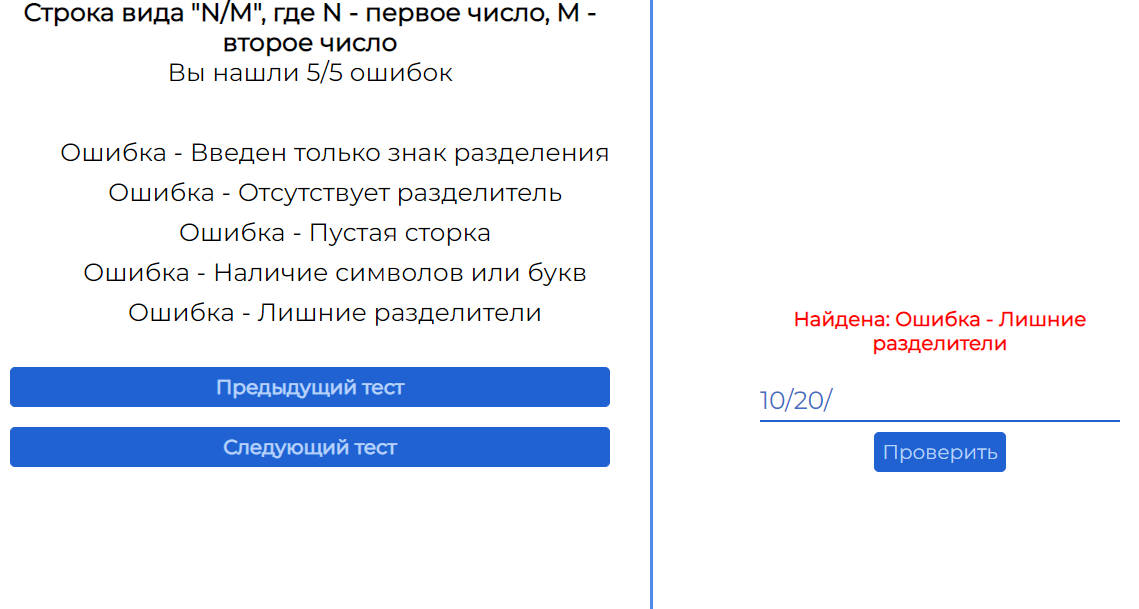


Рисунок 30. Не правильные

**Четвертый тест-кейс**. Необходимо проверить, что в текстовом поле пользователь ввел число. (Рис. 31)

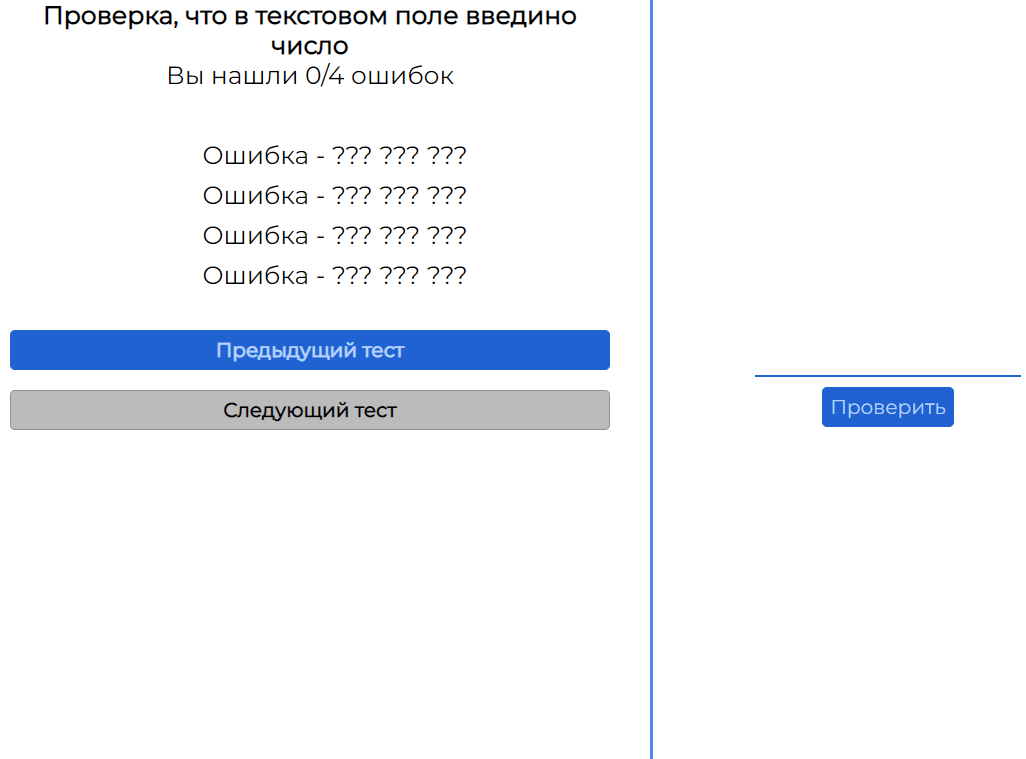


Рисунок 31. Четвертый тест-кейс

Для проверки работы тест-кейса необходимо ввести данные:

Правильные:

1. 123
2. -123
3. 123.45
4. -123.45

Не правильные:

1. 123A - В строке введён недопустимый символ
2. 1234567 - Число слишком большое
3. Пустая строка
4. -9999999 - Число слишком маленькое

Результат (Рис. 32)

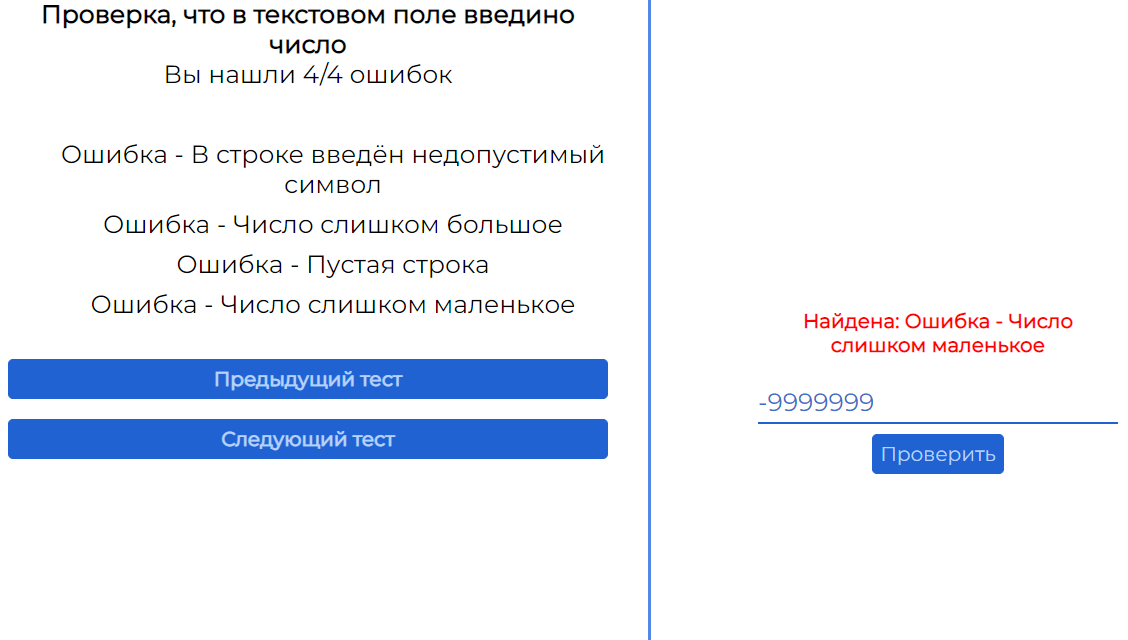


Рисунок 32. Результат

**Пятый тест-кейс**. Проверка, что строка состоит только лишь из букв (русских или латинских)

(Рис. 33)

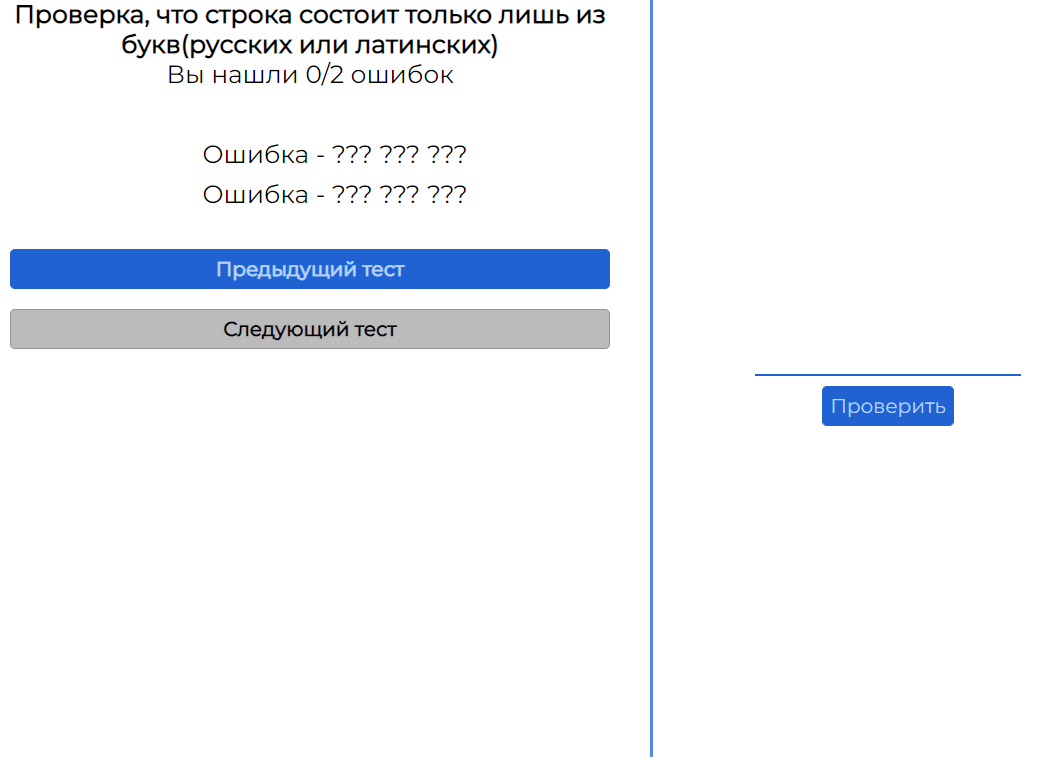


Рисунок 33. Пятый тест-кейс

Для проверки работы тест-кейса необходимо ввести данные:

Правильные: Любые буквы латинского/русского языка

Не правильные:

1. Пустая строка
2. Любую цифру

Результат (Рис. 34)

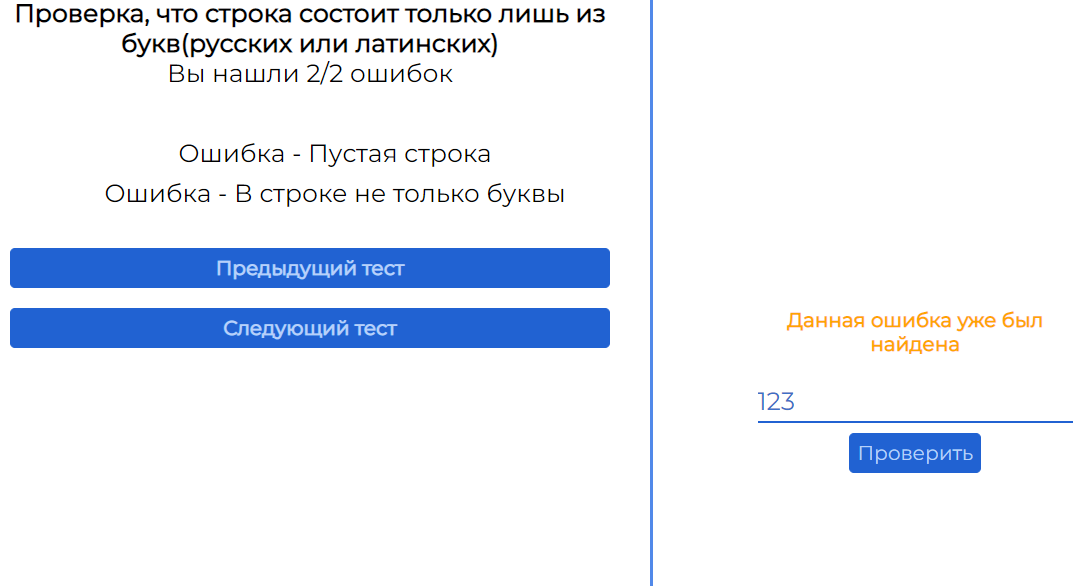


Рисунок 34. Результат

**Шестой тест-кейс**. Проверка на качество пароля (Рис. 35)

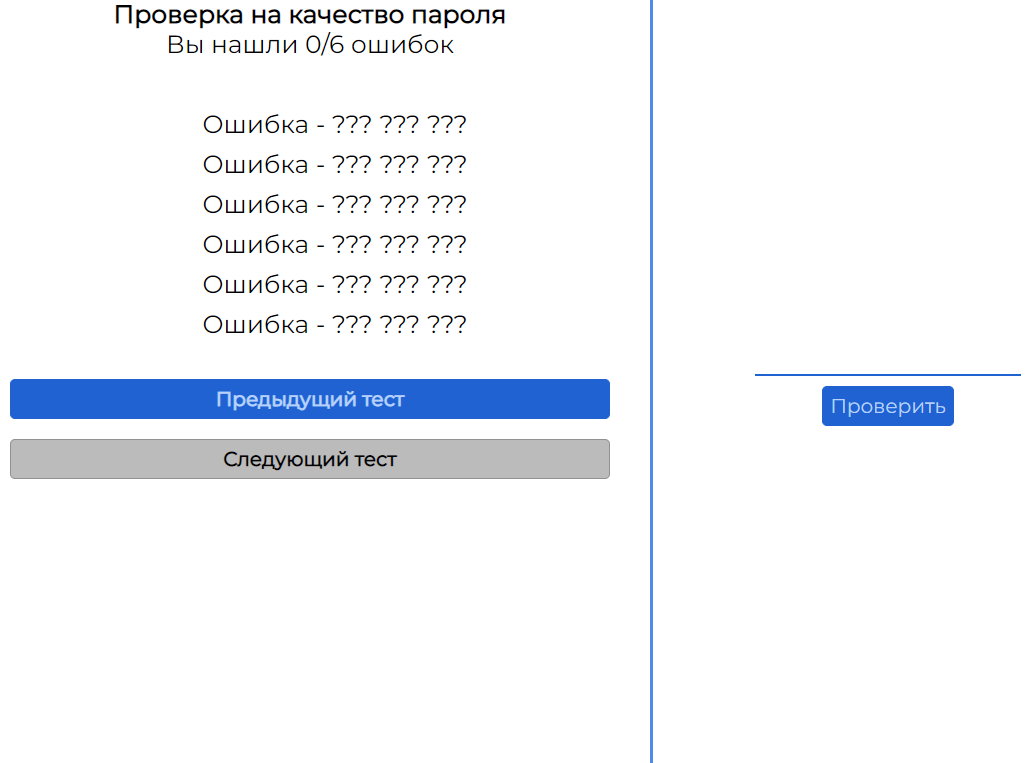


Рисунок 35. Шестой тест-кейс

Для проверки работы тест-кейса необходимо ввести данные:

Правильные

1. 2rY@uPe3
2. R9m%uBe1

Не правильные:

1. Пустая строка

Что должно быть в пароле:

1. Пароль должен содержать минимум одну маленькую латинскую букву
2. Пароль должен содержать минимум одну большую латинскую букву
3. Пароль должен содержать минимум один спец. Символ
4. Пароль должен содержать один символ

Результат (Рис. 34)

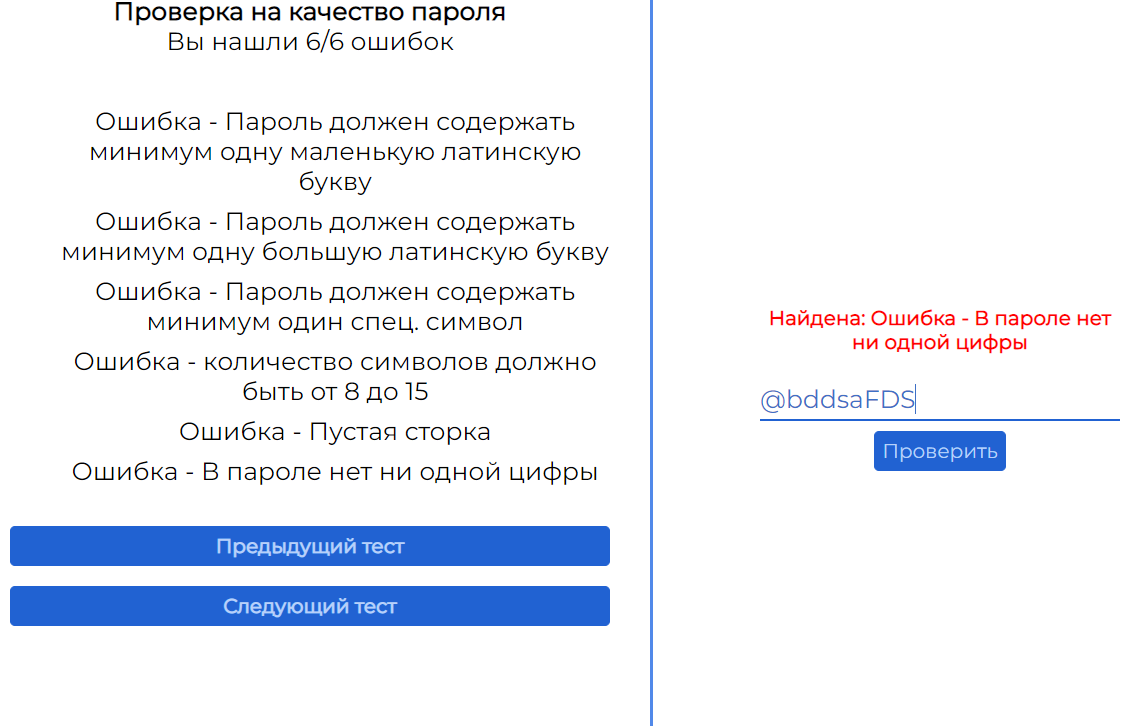


Рисунок 36. Результат

**Седьмой тест-кейс.** Проверка введенного числового кода (Рис. 37)

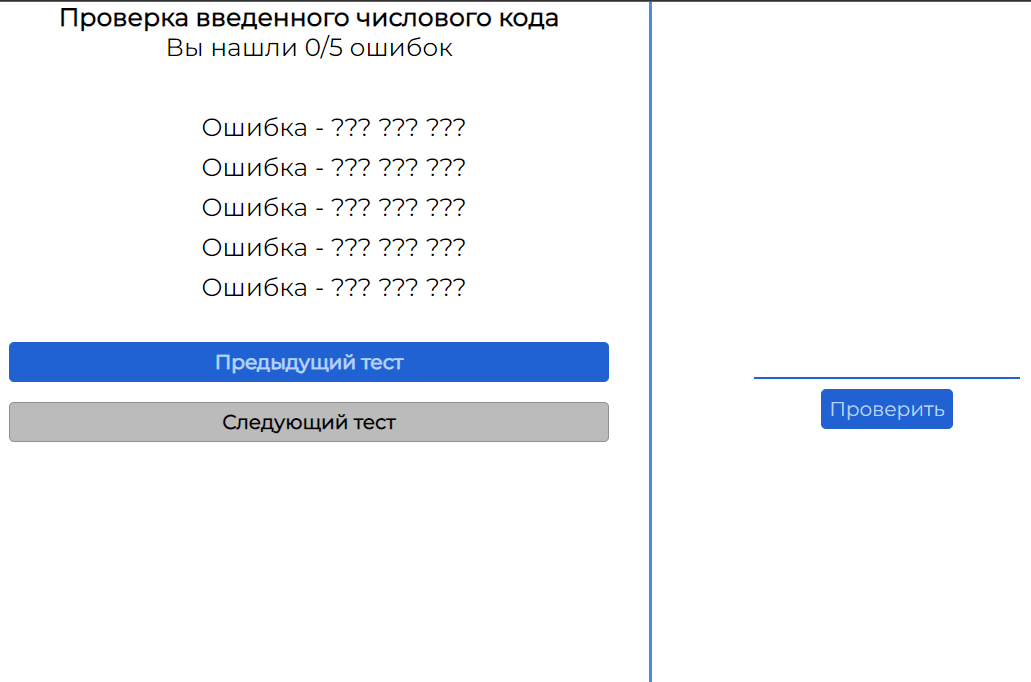


Рисунок 37. Седьмой тест-кейс

Для проверки работы тест-кейса необходимо ввести данные:

1. Правильный код, удовлетворяющий требованию «если сумма первых 9 цифр делится на 3, то в конце (на месте проверочного числа) должна стоять единица 1 (например, "3000030001")
2. Неправильный код, неудовлетворяющий требованию «если сумма первых 9 цифр делится на 3, то в конце (на месте проверочного числа) должна стоять единица 1» (например, "1230000034")
3. Правильный код , удовлетворяющий требованию «во всех остальных случаях в конце кода (на месте проверочного числа) стоит цифра 9» (например, "3000130009")
4. Неправильный код, неудовлетворяющий требованию «во всех остальных случаях в конце кода (на месте проверочного числа) стоит цифра 9» (например, «3000130008»)

Результат (Рис. 38)

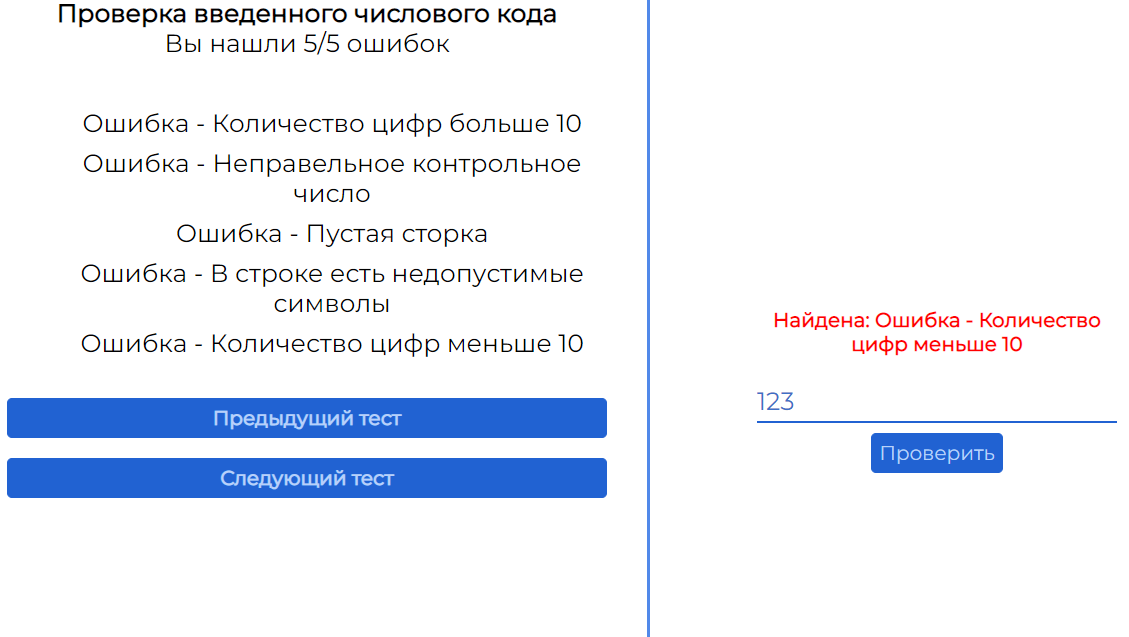


Рисунок 38. Результат

Проверка правильности ввода регистрационного номерного знака автомобиля Российской Федерации. (Рис.39)

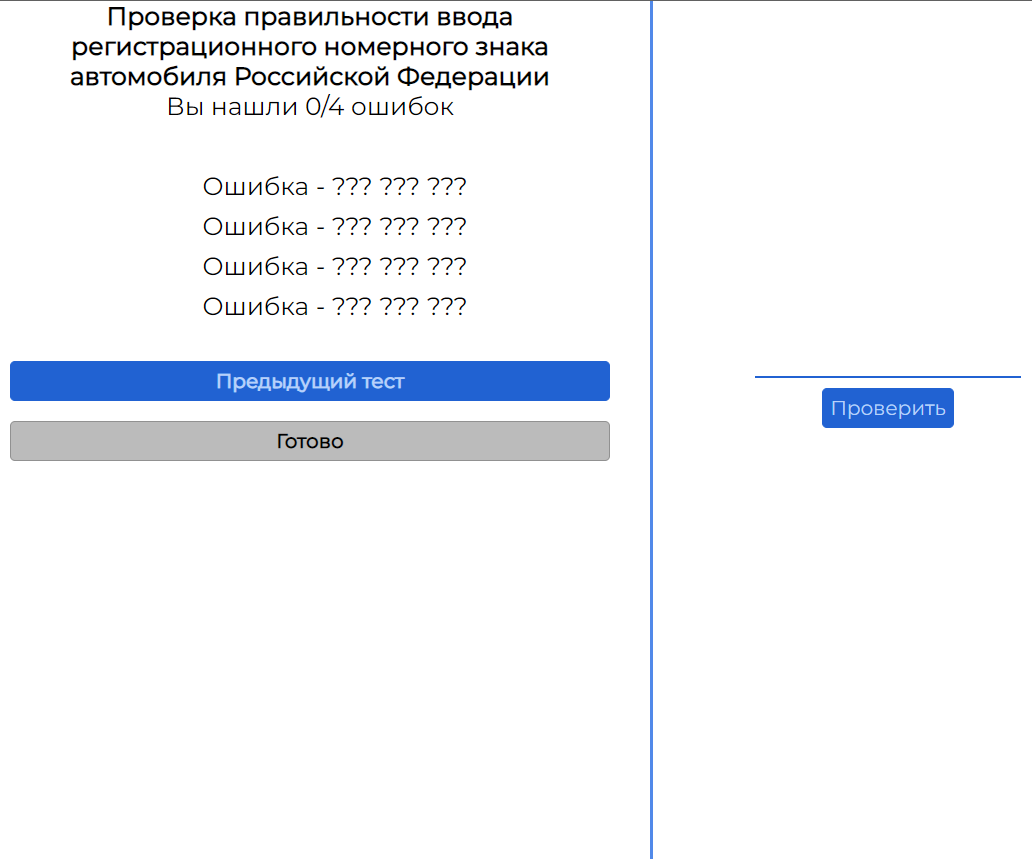


Рисунок 39. Проверка правильности ввода регистрационного номерного знака

Для проверки работы тест-кейса необходимо ввести данные:

Правильные: С065МК78

Не правильные:

1. С000МК78
2. С065МК741
3. U065МК78
4. Пустая строка

Тесты:

1. Пустая строка

2. Правильный гос.номер (использованы А, В, Е, К, М, Н, О, Р, С, Т, У и Х из кириллического /латинского алфавита

3. Номер, в котором присутствуют запрещенные символы

4. Номер, с несуществующим регионом

5. Гос.номер , в котором использованы цифры 000 (диапазон диапазон цифр: 001-999)

Результат (Рис. 40)

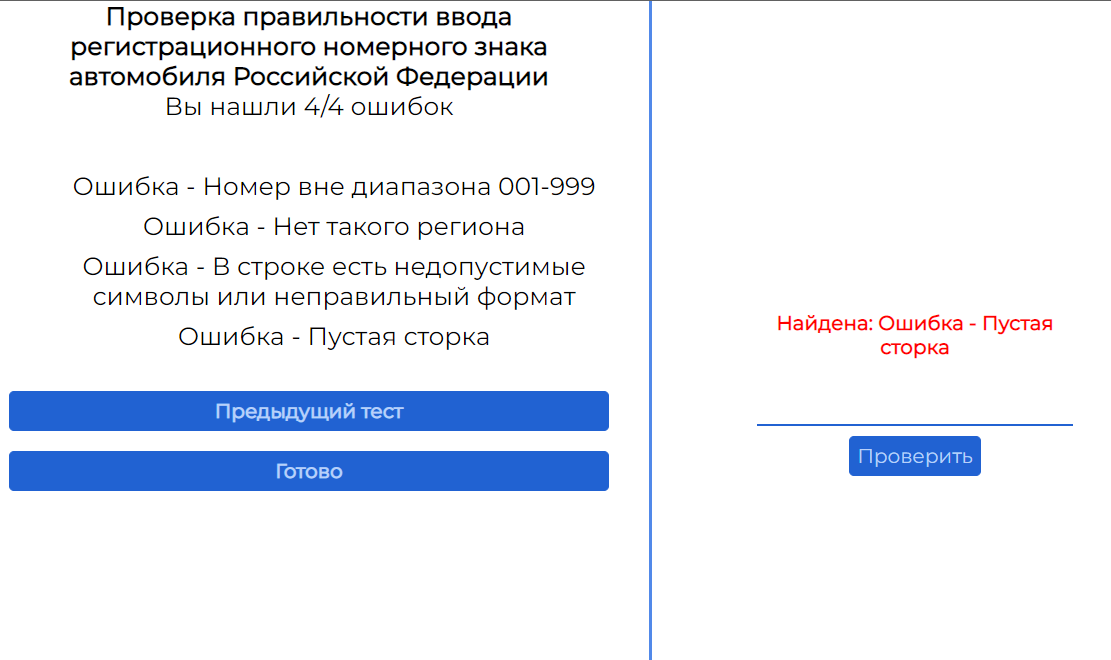


Рисунок 40. Результат

# ЗАКЛЮЧЕНИЕ

Целью дипломного проекта является создание бесплатной программы для начинающих тестировщиков или тех, кто хочет ими стать.

Задачей данного проекта знакомство пользователей с тест-кейсами обеспечение универсальности функций тестирования.

Тема дипломного проекта, является актуальной так как грамотно составленные программы позволяют тестировать сайты, программы и составлять различного рода отчёты по проведённым тестам, что позволяет сэкономить много времени в будущем.

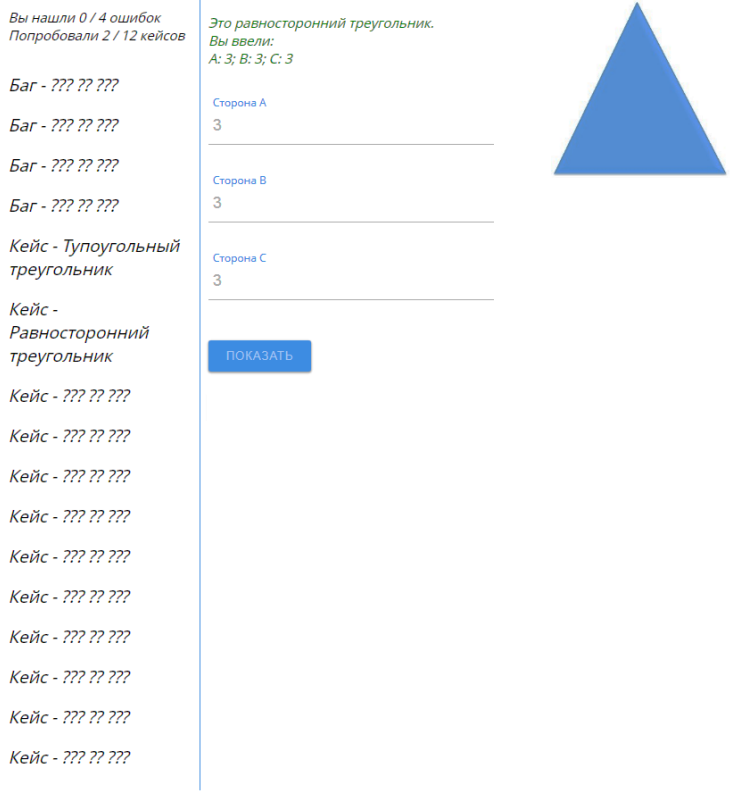
В итоге выполнения дипломной работы были получены следующие основные результаты и выводы:

* Обоснована целесообразность и актуальность разработки системы.
* Изучена предметная область вопроса, рассмотрены существующие решения проблемы.
* Рассмотрены и изучены существующие аналоги системы, выявлены их достоинства и недостатки, конкретизированы данные технического задания
* Разработана структура системы в целом и отдельных модулей в частности.
* Разработаны и описаны рабочие алгоритмы системы

# СПИСОК ЛИТЕРАТУРЫ

1. Основы CSS [Электронный ресурс]. – Режим доступа: <https://html5book.ru/osnovy-css/>.
2. Гарретт, Д. Веб-дизайн. Элементы опыта взаимодействия / Д. Гарретт. – СПб.: Символ-плюс, 2015. – 192 c.
3. Async/await JS [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/async>
4. Json parse JS [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/json>
5. EventListener JS [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/introduction-browser-events>
6. LocalStorage JS [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/localstorage>
7. List CSS element [Электронный ресурс]. – Режим доступа: - <https://www.w3schools.com/css/default.asp>
8. List HTML element [Электронный ресурс]. – Режим доступа: - <https://www.w3schools.com/html/default.asp>
9. Methods of RegExp and String [Электронный ресурс]. – Режим доступа: - <https://javascript.info/regexp-methods>
10. JS Manual [Электронный ресурс]. – Режим доступа: - <https://developer.mozilla.org/ru/docs/Web/JavaScript>
11. CSS Manual [Электронный ресурс]. – Режим доступа: - <http://htmlbook.ru/css>
12. HTML Manual [Электронный ресурс]. – Режим доступа: - <http://htmlbook.ru/html>
13. Regex Manual - <https://learn.javascript.ru/regexp-introduction>
14. JS get element [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/searching-elements-dom>
15. JS dictionary [Электронный ресурс]. – Режим доступа: - <https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide/Keyed_collections>
16. JS OOP [Электронный ресурс]. – Режим доступа: - <https://learn.javascript.ru/about-oop>
17. Создание репозитория на сайте GitHub из Visual Studio [Электронный ресурс] – Режим доступа: <https://docs.microsoft.com/ru-ru/learn/modules/visual-studio-github-push/4-create-repo>
18. CSS Images [Электронный ресурс]. – Режим доступа: - <https://www.w3schools.com/css/css3_images.asp>
19. Dynamic-Pseudo-Classes [Электронный ресурс]. – Режим доступа: -http://htmlbook.ru/index.php/css/hover
20. CSS ID [Электронный ресурс]. – Режим доступа: - http://htmlbook.ru/css/selector/id

# ПРИЛОЖЕНИЕ 1 Образец страницы приложения.



# ПРИЛОЖЕНИЕ 2 Список тест-кейсов

Задание 1

Определение типа треугольника по трем его сторонам. Каждая из сторон задается в отдельном текстовом поле.

Тесты:

1. Не заполнено какое-то одно из полей (или все поля)

2. Тупоугольный треугольник

3. Равнобедренный треугольник

4. Равносторонний треугольник

5. Прямоугольный треугольник

6. Остроугольный треугольник

7. Треугольник, стороны которого заданы неверно (проверка существования треугольника)

8. Отрицательные значения сторон (одной или нескольких)

9. Ввод слишком больших значений сторон

10.Использование нецелых чисел (использована дробная запятая или точка в записи чисел)

11.Ввод в текстовые поля символов, которые недопустимы при вводе чисел (например, букв)

Задание 2

Найти площадь прямоугольника

Тесты:

1. Одна из сторон – пустая строка (Или для обеих сторон не заданы длины)

2. Одна из сторон содержит символы, которые не могут использоваться при написании цифр

3. Первая сторона – положительное целое число Вторая сторона – положительное целое число

4. Одно из чисел – дробное число (должен быть учтен вариант ввода пользователем любого разделителя: или точка, или запятая)

5. Одна из сторон - отрицательное число

6. Одна из сторон – нулевое значение

7. Одна из сторон – отрицательное дробное число (разделитель – либо запятая, либо точка)

8. Тестирование на очень большие числа

9. Тестирование на очень маленькие числа

Задание 3

Пусть имеется строка вида "N/M", где N - первое число, M - второе число. Так получилось, что в некоторых ситуациях значение N превышает значение M. Реализуйте метод, который бы в случае, когда N > M поменял бы первое число на второе(см. примеры)В качестве параметра передается строка.

Тесты:

1. "1/10" => "1/10"

2. "251/100" => "100/100"

3. "-10/10" => "-10/10"

4. Пустая строка

5. Строка, которая содержит разделитель "/" , какая-то из цифр (или все цифры ) отсутствуют

6. Лишние разделители, невозможно вычленить только два числа "35/10/10/25"

7. Наличие символов или букв "10/1b"

8. Отсутствует разделитель "1010"

Задание 4

Необходимо проверить, что в текстовом поле пользователь ввел число.

Тесты:

1. Пустая строка

2. Целые цифры

3. Ноль

4. Отрицательное число

5. Очень большие цифры (например, 768574745959859)

6. Очень маленькие цифры (например -5636345365)

7. Дробное число (разделитель или точка, или запятая)

8. В строке присутствуют символы, отличные от цифр

Задание 5

Необходимо проверить, что строка состоит только лишь из букв (русских или латинских).

Тесты:

1. Пустая строка

2. Наличие символов, отличных от букв (русских или латинских) :цифры, спецсимволы( точки, запятые, пробелы и т.п.)

3. Наличие символов (русских или латинских) в одном регистре

4. Наличие символов (русских или латинских) в разных регистрах

Задание 6

Необходимо сформировать строку, которая будет использоваться в качестве "Пароля"

Требования к паролю:

1. количество символов - от 8 до 15

2. пароль должен содержать цифры

3. пароль должен содержать не менее одной латинской буквы в нижнем регистре

4. пароль должен содержать не менее одной латинской буквы в верхнем регистре

5. пароль должен содержать не менее одного спец.символа

6. Необходимо реализовать метод, который бы создавал строку и выполнял проверку того, что строка удовлетворяет все требованиям к паролю

7. если пароль содержит кириллические символы, то выдается исключение «Кириллические символы запрещены при вводе пароля»

Тесты:

1. Пустая строка

2. Наличие кириллических символов

3. Сложный пароль, который удовлетворяет всем требованиям

4. Слабый пароль: не проходит по длине (маленькая длина)

5. Слабый пароль: не проходит по длине (слишком большая длина)

6. Слабый пароль: не проходит по символам (отсутствуют цифры)

7. Слабый пароль: не проходит по символам (отсутствуют спецсимволы)

8. Слабый пароль: не проходит по символам (отсутствуют маленькие латинские буквы)

9. Слабый пароль: не проходит по символам (отсутствуют заглавные латинские буквы)

Задание 7

Проверка введенного числового кода.

Требования к коду:

1. должен содержать 10 цифр.

2. последняя цифра кода – контрольное (проверочное число)

3. если сумма первых 9 цифр делится на 10, то в конце (на месте проверочного числа) должен стоять 0

4. если сумма первых 9 цифр делится на 3, то в конце (на месте проверочного числа) должна стоять единица 1;

5. во всех остальных случаях в конце кода (на месте проверочного числа) стоит цифра 9.

Тесты:

1. Пустая строка

2. Строка из меньшего количества цифр (меньше 10)

3. Строка из большего количества цифр (больше 10)

4. Строка, содержащая символы, отличные от цифр

5. Правильный код, удовлетворяющий требованию «если сумма первых 9 цифр делится на 10, то в конце (на месте проверочного числа) должен стоять 0» (например, "1234123400")

6. Неправильный код, неудовлетворяющий требованию «если сумма первых 9 цифр делится на 10, то в конце (на месте проверочного числа) должен стоять 0» (например, "1234123401")

7. Правильный код, удовлетворяющий требованию «если сумма первых 9 цифр делится на 3, то в конце (на месте проверочного числа) должна стоять единица 1 (например, "3000030001")

8. Неправильный код, неудовлетворяющий требованию «если сумма первых 9 цифр делится на 3, то в конце (на месте проверочного числа) должна стоять единица 1» (например, "1230000034")

9. Правильный код, удовлетворяющий требованию «во всех остальных случаях в конце кода (на месте проверочного числа) стоит цифра 9» (например, "3000130009") 10. Неправильный код, неудовлетворяющий требованию «во всех остальных случаях в конце кода (на месте проверочного числа) стоит цифра 9» (например, "3000130008")

Задание 8

Проверка правильности ввода регистрационного номерного знака автомобиля Российской Федерации. Комбинации на стандартных номерных знаках автомобилей строятся по принципу — 3 буквы, 3 цифры. Буквы означают серию номерного знака, а цифры — номер. ГОСТом для использования на знаках разрешены 12 букв кириллицы, имеющие графические аналоги в латинском алфавите — А, В, Е, К, М, Н, О, Р, С, Т, У и Х. В правой части номерного знака имеется секция, в которой размещены: в нижней части — флаг РФ и буквенный код RUS, а в верхней — код субъекта РФ, где был зарегистрирован автомобиль.

Тесты:

1. Пустая строка

2. Правильный гос.номер (использованы А, В, Е, К, М, Н, О, Р, С, Т, У и Х из кириллического /латинского алфавита

3. Номер, в котором присутствуют запрещенные символы

4. Номер, с несуществующим регионом

5. Гос.номер, в котором использованы цифры 000 (диапазон диапазон цифр: 001-999