

Лабораторная работа №7. Тестирование и оценка пользовательского интерфейса.

Цели работы:

1. Изучить методы тестирования пользовательских интерфейсов.
2. Провести тестирование выбранного интерфейса, используя различные методы.
3. Выявить основные проблемы с удобством использования и интуитивностью интерфейса.

Задания для выполнения:

1. Выберите интерфейс приложения для тестирования (приложение одnogруппника).

Выбранная программа для тестирования была выбрана «numdetect». На рисунке 1 представлена главная страница с интерфейсом приложения.

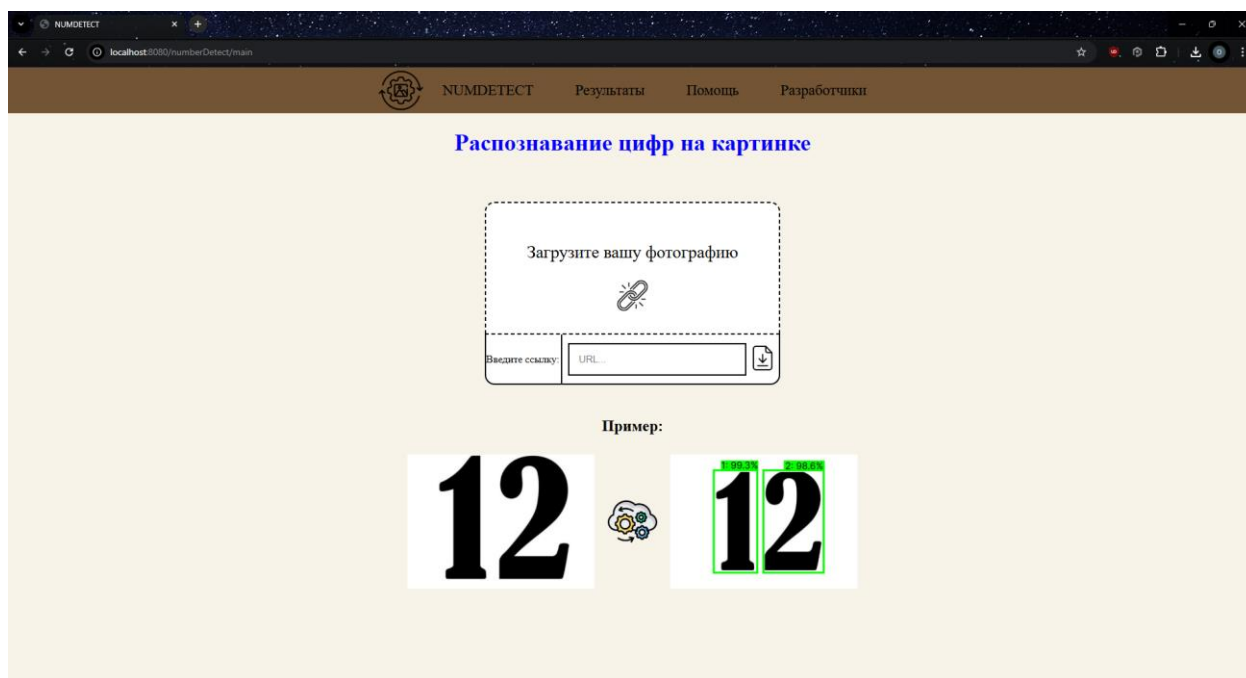


Рисунок 1 – Основная страница приложения «Распознавание рукописных цифр»

2. Проведите тестирование с 5-7 пользователями:

- Попросите пользователей выполнить несколько задач (например, "найдите продукт и добавьте его в корзину" или "переведите деньги на счет").
- Записывайте время, которое пользователи тратят на выполнение каждой задачи, и количество ошибок, которые они допускают.

На таблице 1 представлен подробный отчет по пользователям, выполняемым ими задачи, затраченным временем на задачу и допущенные ошибки во время выполнения.

Таблица 1 – Тестирование интерфейса пользователями

Задача	Пользователь	Время (секунды)	Ошибки
Вставка ссылки в поле для ввода ссылки	1	2.5	-
	2	3.3	Долгое исправление ошибки в связи с небольшим размером URL поля
	3	2.8	-
	4	1.9	-
	5	2.2	-
Вставка изображения из файловой системы	1	7.1	-
	2	6.3	-
	3	4.4	-
	4	8.5	
	5	9.3	После загрузки изображения не вышло уведомление что прошла загрузка изображения
Нажатие на кнопку для выполнения распознавания изображения	1	1.2	-
	2	2.1	Долгое наведение на маленькую кнопку
	3	3.0	Промак по кнопке URL
	4	1.0	-
	5	1.4	-
Перемещение по всем страницам с помощью кнопок меню	1	5.3	-
	2	7.3	Попытка нажать иконку приложения с целью перейти на главную страницу
	3	4.3	-
	4	5.5	Попытка нажать иконку приложения с целью перейти на главную страницу
	5	4.5	-

3. Используйте метод анализа тепловых карт для выявления зон на экране, которые привлекают больше всего внимания пользователей: – Используйте инструменты для создания тепловых карт (например, Hotjar или другие аналоги).

В качестве инструмента для создания тепловых карт был выбран сервис (exproze.app), он позволяет на страницах выделять зоны с повышенным уровнем внимания для пользователя.

На рисунках 2-5 изображены тепловые карты для каждой страницы приложения «Распознавание рукописных цифр».



Рисунок 2 – тепловая карта главной страницы

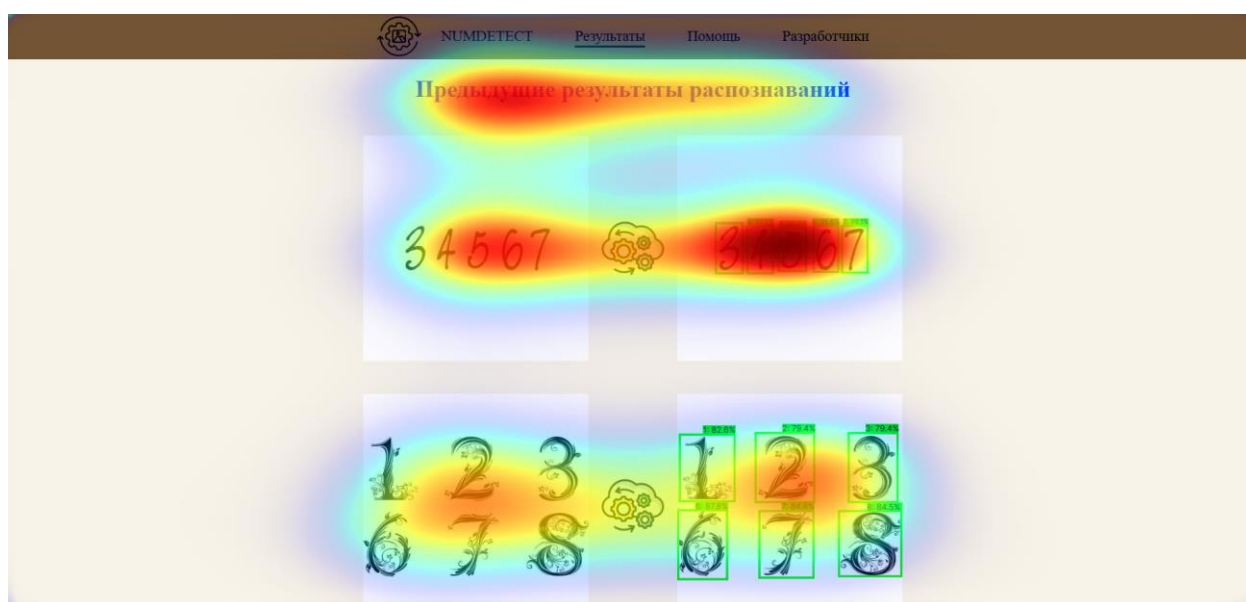


Рисунок 3 – тепловая карта страницы «Результаты»

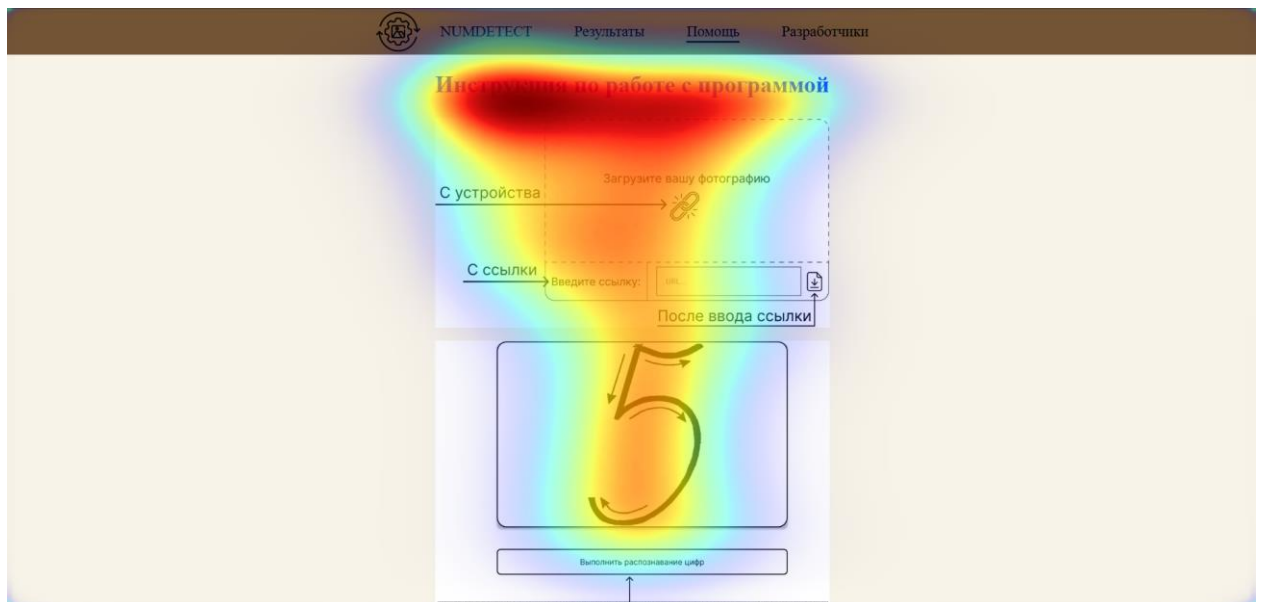


Рисунок 4 – Тепловая карта страницы «Помощь»

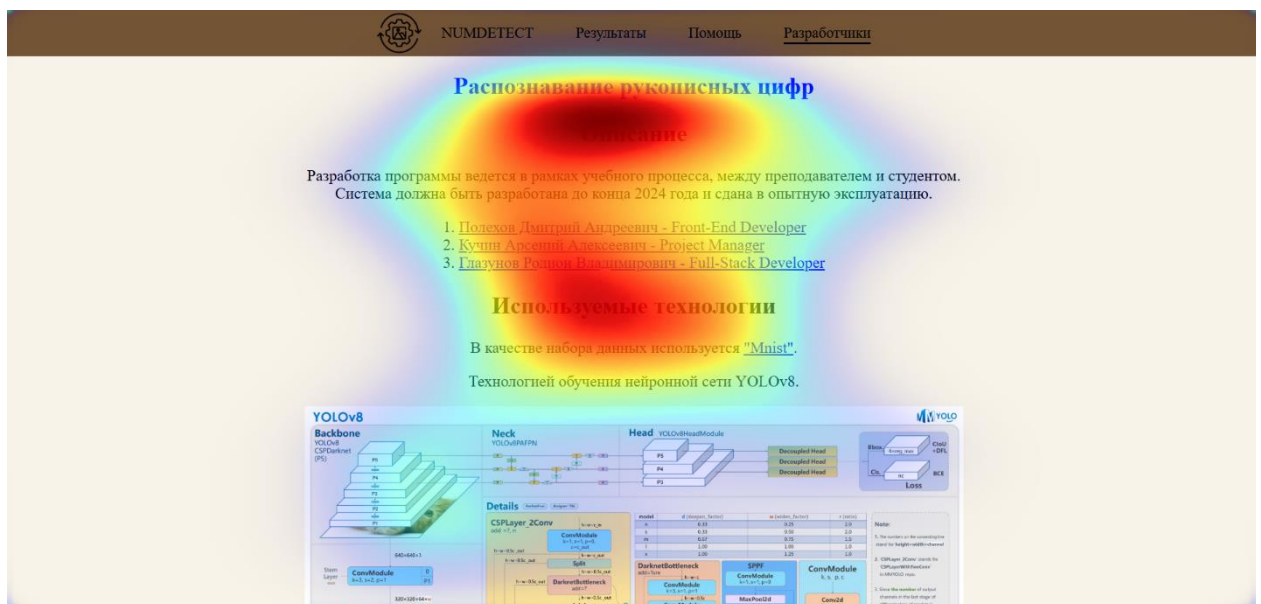


Рисунок 5 – Тепловая карта страницы «Разработчики»

– Определите, какие элементы интерфейса чаще всего используются, а какие игнорируются.

На основе тепловых карт, можно сделать вывод, что в центре внимания всегда верхняя часть сайта. Также меню можно отнести к игнорируемым элементам интерфейса, как и задумывалось разработчиками.

4. Проведите тестирование UI с использованием существующих средств для автоматизированного тестирования (<https://www.geeksforgeeks.org/top-automated-ui-testing-tools/>). Получите отчет о тестировании.

Для проведения тестирования UI воспользуемся сервисом TestCase Studio (selectorhub.com/testcase-studio). Это позволит понять какие запросы отправляются по нажатию на определённые элементы.

На рисунках 6-8 будут отображены основные протестированные функции. Ссылка на полное автоматизированное тестирование

(<https://drive.google.com/file/d/16xTDrer8i68nMaLm2VeWTCpjL37GEFtr/view?usp=sharing>)

На рисунке 6 отображен результат распознавания файла, загруженного из файловой системы.

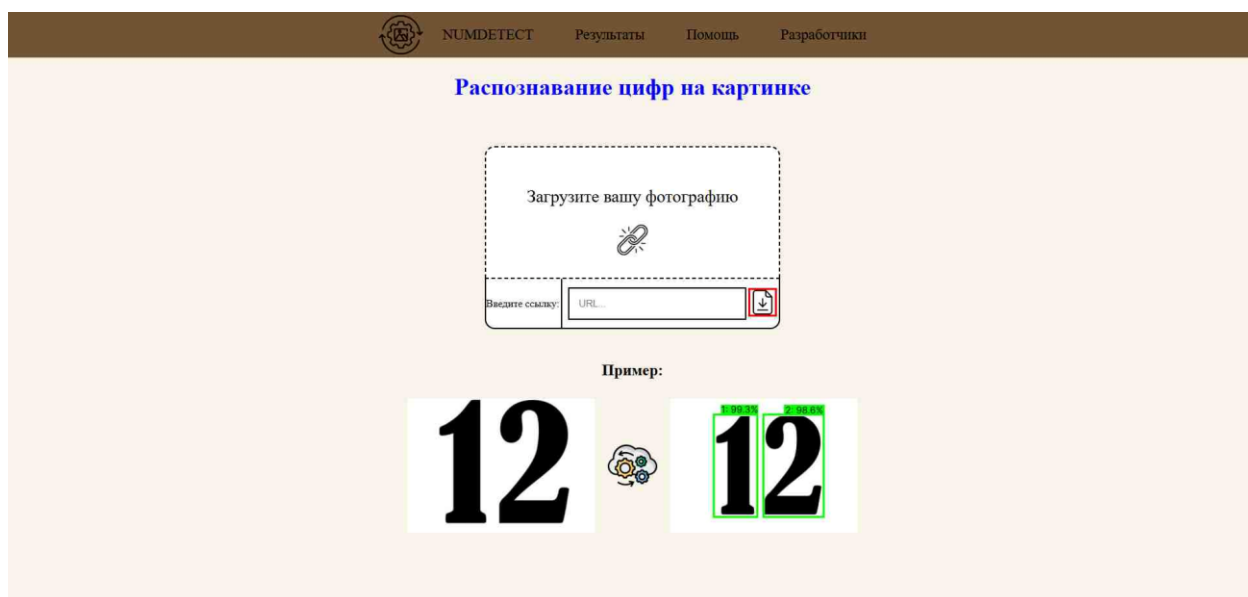


Рисунок 6 – Распознавание картинки из файловой системы

Далее было протестировано распознавание изображения по введённой ссылке, для этого курсор был поставлен в поле с вводом URL (рисунок 7)

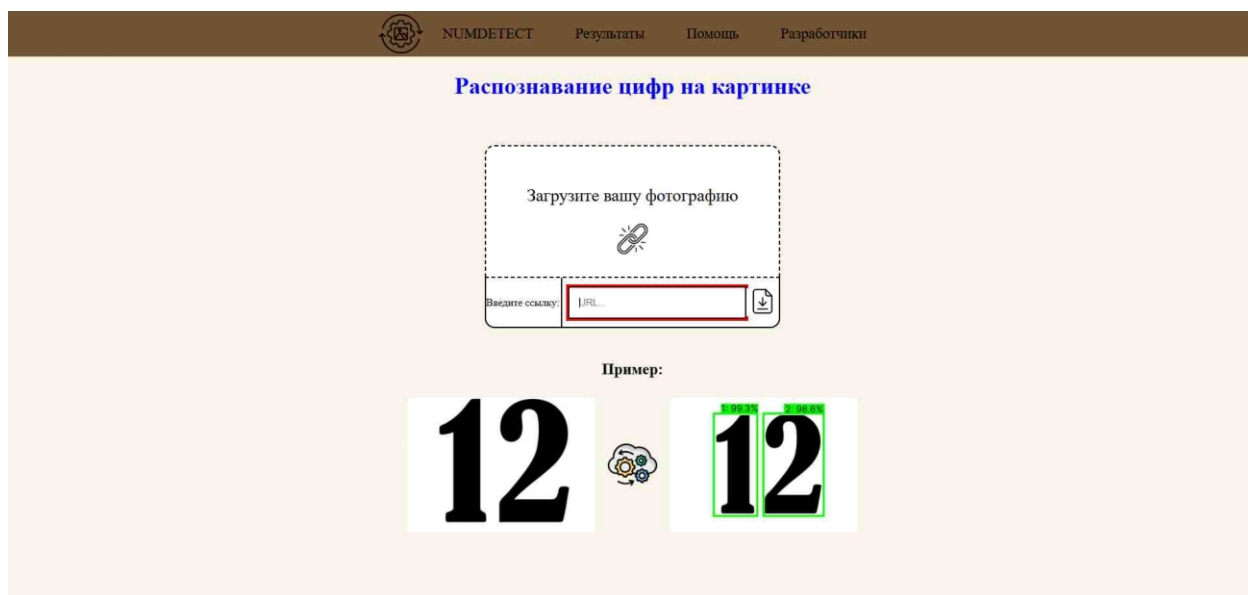


Рисунок 7 – Ввод ссылки в поле «URL»

После ввода ссылки нужно выполнить распознавание, нажав на кнопку файла (рисунок 8).

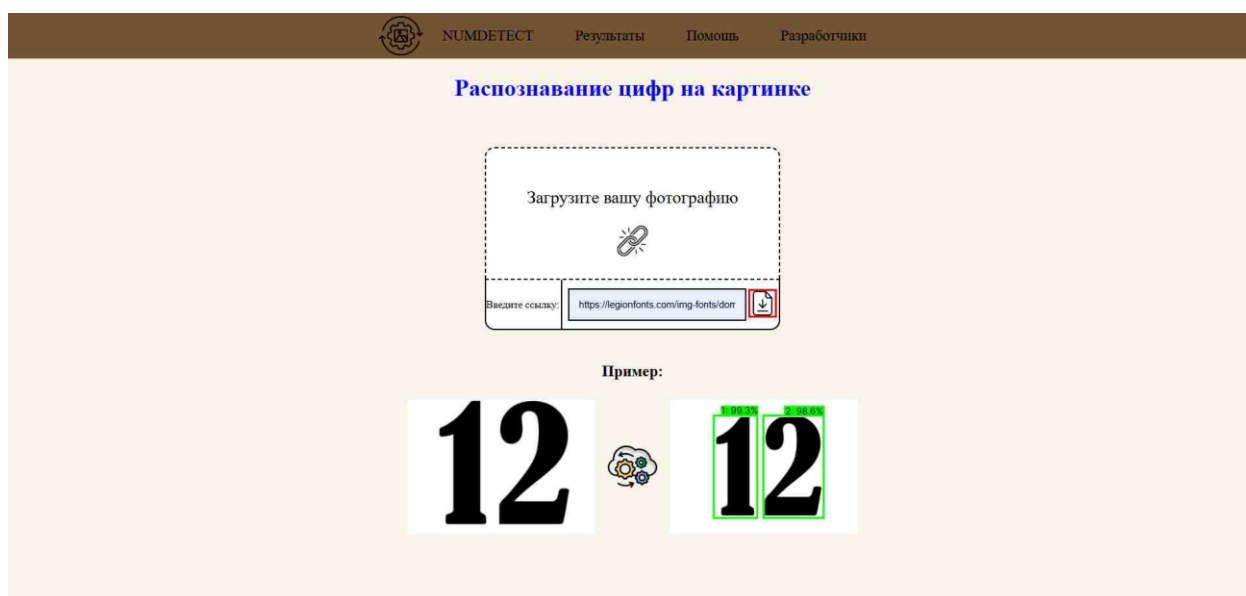


Рисунок 8 – Нажатие на кнопку файла для распознавания изображения

Полный результат тестирования программой «TestCase Studio» представлен на рисунке 9. Она отображает вводимые данные, нажимаемые кнопки и CSS селекторы выбора со всеми скриншотами действий тестирования.

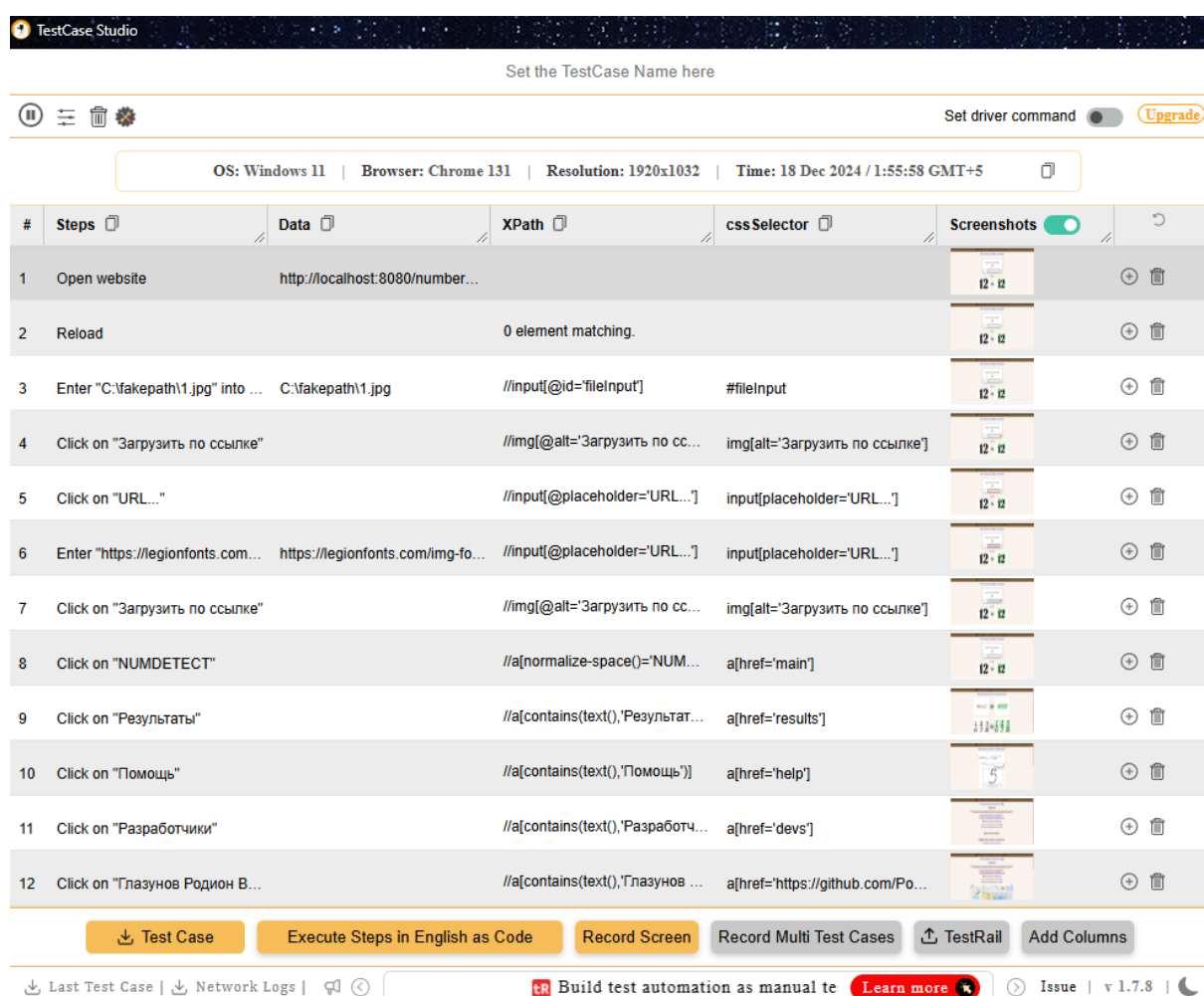


Рисунок 9 – Сервис автоматизированного тестирования «TestCase Studio»

5. Проанализируйте результаты все методов тестирования:

– Какие элементы интерфейса были понятны пользователям, а какие вызвали затруднения?

После тестирования пользователями программы, было выяснено, что после загрузки собственного изображения программа не выводит сообщение о загруженном файле. Остальные элементы тестирования не навязчивые и не вызывали вопросов. Сам интерфейс был спроектирован таким образом, чтобы пользователь не мог ошибочно нажать не на тот элемент интерфейса или промахнуться по элементу интерфейса с целью достижения своей задачи.

– Сколько времени понадобилось для выполнения каждой задачи?

Для выполнения основных задач понадобилось:

1) распознавание чисел на изображении из файловой системы: 8,86 секунды;

2) распознавание чисел на изображении по ссылке: 4,28 секунды.

– Были ли допущены ошибки при навигации по приложению или выполнении задач

При навигации в приложении ошибок допущено не было, верхнее меню не перегружено лишними пунктами, все страницы не содержали лишней или нагружающей пользователя информацией.

6. Подготовьте отчет с описанием:

Метод тестирования.

Для тестирования применялся метод «чёрного ящика», пользователи использовали программу без знаний внутреннего кода программы.

– Багрепорты с описанием выявленных проблем.

Багрепорты представлены в таблице 2.

Предложения по улучшению интерфейса:

1) переделать инструкцию к программе в соответствии с текущими функциями программы;

2) после загрузки файла из файловой системы нет никакого сообщения о успешной или неуспешной загрузке файла;

3) иконка приложения не является ссылкой, на многих сайтах при нажатии она ведёт на главную страницу;

4) шрифт у различных элементов на главной странице имеет разный размер, можно несколько элементов подвести под одинаковый шрифт для более правильного отображения страницы;

5) после выполнения распознавания можно отображать сообщение о завершённости операции;

6) если не прикреплять картинку и не вводить ссылку, то приложение на этот счёт не имеет исключения.

Таблица 2 – багрепорты по приложению

Элемент	Шаги	Ожидаемый результат	Результат	Рекомендация
Кнопка «Файл»	Нажать на кнопку «файл» не загрузив файл или введя ссылку	Получение сообщения о ошибке	HTTP ERROR 400	Настройка исключения
	Ввод ссылки без приставки HTTP/HTTPS	Обработка ссылки без приставки HTTP/HTTPS	Получение сообщения о некорректной ссылки	Проверка ссылки на валидность без приставок
	Ввод ссылки и нажатие на кнопку	После успешного распознания, очистка поля для ссылки	Ссылка не очищается из поля	Очистка поля ссылки после успешного распознания
	Прикрепление файла из файловой системы и нажатие на кнопку	После первого нажатия очистка обработанного файла из поля	Множественная возможность повторной обработки файла	Очистка поля файла после успешного распознания
	Прикрепление файла из файловой системы с одновременным вводом ссылки	Поочерёдная обработка двух файлов	Обработка только файла из файловой системы	Настройка очереди распознания