

Научная статья
Статья в открытом доступе
УДК 519:004.5
doi: 10.30987/2658-4026-2022-4-252-261

Разработка автоматизированной системы для создания и тестирования прототипов пользовательских интерфейсов

Дергачев Константин Владимирович^{1✉}, Эссам Мвом Франсис Нозль²

^{1,2} Брянский государственный технический университет; Брянская область, Брянск, Россия

¹ kv.dergachev@gmail.com; <https://orcid.org/0000-0002-2016-8981>

² francis.mvom@mail.ru; <https://orcid.org/0000-0001-5516-5716>

Аннотация.

Описан процесс разработки веб-приложения для создания и тестирования прототипов пользовательских интерфейсов. Приведены результаты анализа процессов разработки и тестирования интерфейсов. Проанализированы аналогичные приложения. На основе анализа выявлены недостатки аналогичных систем и сформулированы требования к разрабатываемой системе. Разработаны математические модели для автоматизированного оценивания качества графического пользовательского интерфейса на основе гармонии цветов, видимости объектов, шрифтов, выравнивания, баланса, пространства между элементами, соблюдения пропорций и единства стиля. Проведено функциональное моделирование системы. Разработана база данных для хранения прототипов и результатов их тестирования. Создан интерфейс автоматизированной системы проектирования и тестирования пользовательских интерфейсов. Сформулирована методология автоматизации тестирования пользовательских интерфейсов и на ее основе созданы алгоритмы для автоматизированного тестирования прототипов по различным критериям. Приложение реализовано на языках Java и Java Script. Оно позволяет: создавать прототипы интерфейсов для всех типов экранов; оценивать визуальную грамотность и структуру прототипов интерфейсов; предлагать решения для исправления эстетических и стилистических ошибок; планировать тесты и добавлять тестовые задания; проводить А/В-тестирование; определять фокус внимания и маршрут передвижения пользователя; определять среднее время выполнения заданий; записывать видео по шагам во время юзабилити-тестирования.

Ключевые слова: автоматизированная система, прототип, пользовательский интерфейс, методика проверки качества, дизайн, тестирование

Для цитирования: Дергачев К. В., Эссам М. Ф. Н. Разработка автоматизированной системы для создания и тестирования прототипов // Эргодизайн. №4 (18). 2022. С. 252-261. <http://dx.doi.org/10.30987/2658-4026-2022-4-252-261>

Original article
Open Access Article

Development of an automated system for creating and testing user interface prototypes

Konstantin V. Dergachev^{1✉}, Essam Mvom Francis Noel²

^{1,2} Bryansk State Technical University; Bryansk region, Bryansk, Russia

¹ kv.dergachev@gmail.com; <https://orcid.org/0000-0002-2016-8981>

² francis.mvom@mail.ru; <https://orcid.org/0000-0001-5516-5716>

Abstract.

The process of developing a web application for creating and testing prototypes of user interfaces is described. The results of analysing the processes of developing and testing interfaces are given. Similar applications are analysed. Based on the analysis, the shortcomings of similar systems are identified and the requirements for the system being developed are formulated. Mathematical models are worked out for automated evaluation of the quality of a graphical user interface based on the harmony of colours, visibility of objects, fonts, alignment, balance, space between elements, compliance with proportions and unity of style. The functional modelling of the system is carried out. A database is developed for storing prototypes and their testing results. An interface for an automated system for designing and testing

user interfaces is created. A methodology for automating testing of user interfaces is formulated and algorithms for automated testing of prototypes according to various criteria are created on its basis. The application is implemented in Java and Java Script languages. It allows one to: create interface prototypes for all types of screens; assess visual literacy and structure of interface prototypes; offer solutions to correct aesthetic and stylistic errors; plan tests and add test tasks; conduct A/B testing; determine the focus of attention and the user movement route; determine the average time to complete tasks; record video step by step during usability testing.

Keywords: automated system, prototype, user interface, quality control method, design, testing

For citation: Dergachev K.V., Essam M.F.N. Development of an automated system for creating and testing user interface prototype // Ergodizayn [Ergodesign], 2022, No. 4 (18). Pp. 252-261. doi:10.30987/2658-4026-2022-4-252-261

Введение

Исследование направлено на разработку математических моделей и автоматизацию ресурсоемких задач процессов тестирования и проектирования пользовательского интерфейса на этапе разработки программного обеспечения. Целью работы является создание автоматизированной системы проектирования и тестирования интерфейсов для повышения качества и скорости их разработки.

С 2000-х годов произошел бурный рост приложений с графическим интерфейсом и каждый год наблюдается появление многих тысяч новых веб-приложений, мобильных и настольных приложений, которые становятся все более доступными для всех пользователей (детей, взрослых, пожилых людей) с помощью компьютеров, планшетов и смартфонов. Разработчикам приходится разрабатывать варианты одного приложения для нескольких устройств с различными форматами экрана или операционных систем, что приводит к огромным затратам на анализ, разработку и тестирование. В пользовательских интерфейсах приложений может возникнуть множество проблем, таких как несовместимость с различными разрешениями экранов, перегруженность элементами управления, неадекватная терминология, отсутствие единого стиля, ошибки в структуре интерфейса, ошибки классификации объектов и несоответствующие пиктограммы, что вызывает недовольство пользователей, замедляет скорость использования и увеличивает количество ошибок [1].

Для решения этих проблем ИТ-компании включают экспертов в команду разработчиков, чтобы обнаружить и исправить недостатки пользовательского интерфейса в графических приложениях до их внедрения [1]. Также тестирование проводится с участием пользователей, чтобы убедиться, что приложения понятны и удобны потребителю. Однако оба эти метода

требуют больших затрат времени и ресурсов для реализации. Использование существующих программных решений и методов не позволяет в полной мере сочетать автоматизацию проектирования интерфейсов с их качественным и быстрым тестированием [2]. Существующие решения ориентированы либо на проектирование, либо на тестирование и не позволяют автоматизировать всестороннюю экспертную оценку качества интерфейса [2, 3]. В работах [4-7] предлагаются методы экспертной оценки качества интерфейсов, которые требуют формализации для построения на их основе автоматизированной системы для создания и тестирования пользовательских интерфейсов.

1. Анализ и формализация моделей оценки пользовательского интерфейса

План исследования включает следующие этапы: анализ предметной области и методов тестирования пользовательских интерфейсов; выбор критериев оценивания и построение математических моделей оценивания качества пользовательского интерфейса; разработка архитектуры и проектных моделей автоматизированной системы; программная реализация системы проектирования и тестирования интерфейсов; оценивание результатов работы созданной системы.

1.1. Анализ текущего процесса проектирования и тестирования графического интерфейса пользователя.

Схема бизнес-процесса проектирования и тестирования пользовательского интерфейса показана на рис. 1. Анализ процесса показал, что процесс является итерационным [1] и во время итераций выполняются практически одни и те же задачи. Существует три уровня итераций: итерация между проектированием основного экрана и тестированием на низком уровне; возврат с проектирования низкого уровня на высокий уровень; возврат с проектирования низкого уровня на этап постановки задачи.

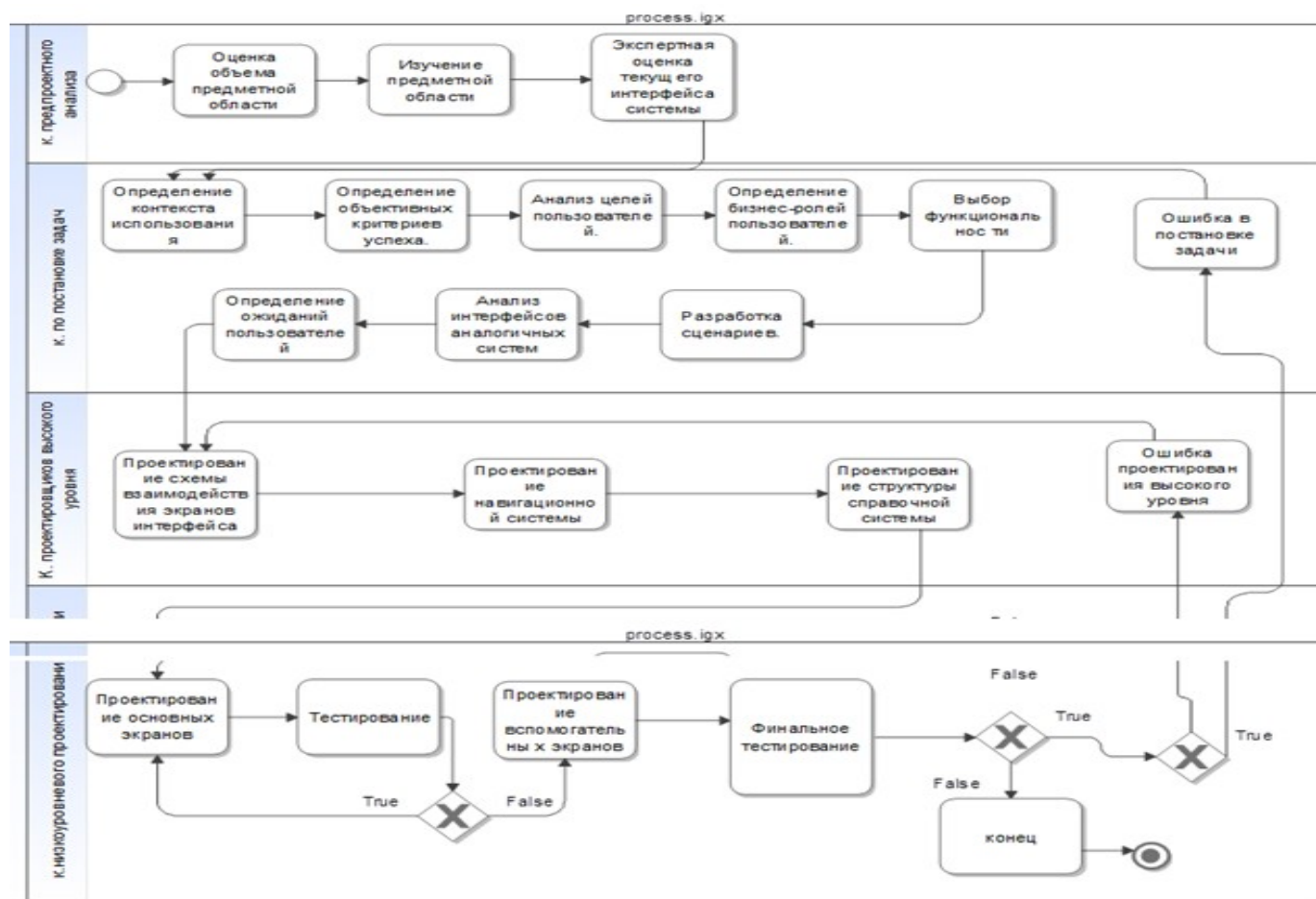


Рис. 1. Схема процесса проектирования и тестирования графического интерфейса пользователя
 Fig. 1. Diagram of the graphical user interface design and testing process

Установлено, что задачи первого уровня итераций, включая анализ элементов интерфейса и результатов после тестирования юзабилити, являются наиболее трудоемкими повторяющимися задачами, поэтому их автоматизация позволит сократить время выполнения процесса проектирования и тестирования пользовательского интерфейса.

1.2. Формализация методов автоматической оценки пользовательского интерфейса

Для достижения поставленной цели необходимо: автоматизировать оценку элементов интерфейса (до юзабилити-тестирования) и нескольких критериев анализа (после юзабилити-тестирования).

Автоматическая оценка интерфейса включает анализ гармонии цветов, видимости объектов, шрифтов, выравнивания, баланса, пространства между элементами, соблюдения пропорций и единства стиля [8]. Определим критерии оценивания пользовательского

интерфейса, методики оценивания качества каждого критерия и весовые коэффициенты.

Гармония цветов

Предлагается цветовую гармонию оценивать от 0 до 12 баллов. Оценка гармонии цветов основывается на трех критериях: количество цветов; совместимость; использование. Каждому из них присвоим равные весовые коэффициенты – 4 балла.

Количество цветов. По правилам дизайна количество цветов должно быть между 2-4 для хорошего восприятия пользователем [9]. В случае соблюдения правила получаем максимальную оценку 4. Формально данное правило выражается следующим образом:

$$S_k(x) = \begin{cases} \frac{4}{x} + 1, & x > 4; \\ \frac{x}{2} + 1,3, & x < 2; \\ 4, & x \in [2; 4]. \end{cases} \quad (1)$$

Где: x — количество цветов в интерфейсе;

S_k — удовлетворение от количества цветов.

Совместимость цветов. Два или четыре цвета совместимы тогда и только тогда, когда они формируют палитру в цветовом круге [8]. Кроме белого, серого и коричневого, каждый цвет представлен в цветовом круге значением. Необходимо проверить, формируют ли цвета интерфейса действительную палитру: комплементарную, сплит-комплементарную, тетрадную, триадичную или аналоговую. Таким образом, совместимость выражается следующим образом

$$S_c(x) = \begin{cases} 4, & x=1; \\ 0, & x=0. \end{cases} \quad (2)$$

Где:

x — индикатор совместимости цветов (1 — совместимые цвета, 0 — несовместимые цвета);

S_c — удовлетворение совместимостью цветов.

Использование цвета. Использование цвета выражается в соответствии с правилом 60-30-10 [6], которое гласит, что: 60% — доминирующий цвет, 30% — вторичный цвет, а 10% — цвет акцента. Применим линейную зависимость для определения степени выполнения данного правила. Общая удовлетворенность использованием цветов является суммой удовлетворенностей количеством использования каждого из трех цветов.

$$S_i = S_{i1} + S_{i2} + S_{i3} \quad (3)$$

Где:

S_i — общая удовлетворенность использованием цветов.

S_{i1} — удовлетворенность колоризацией основного цвета;

S_{i2} — удовлетворенность колоризацией вторичного цвета;

S_{i3} — удовлетворенность колоризацией цвета акцента.

При точном выполнении правила для доминирующего цвета (60%) получим максимальное значение качества 1,34, в остальных случаях используем формулу:

$$S_{i1}(x) = \begin{cases} \frac{1,34 * 100 - s_1}{40}, & s_1 \in [60; 100]; \\ \frac{1,34 * s_1 - 20}{40}, & s_1 \in [20; 60]. \end{cases} \quad (4)$$

При точном выполнении правила для вторичного цвета (30%) получим

максимальное значение качества 1,33, в остальных случаях используем формулу:

$$S_{i2}(x) = \begin{cases} \frac{1,33 * s_2}{30}, & s_2 \in [0; 30]; \\ \frac{1,33 * 60 - s_2}{30}, & s_2 \in [20; 60] \end{cases} \quad (5)$$

При точном выполнении правила для цвета акцента (10%) получим максимальное значение качества 1,33, в остальных случаях используем формулу:

$$S_{i3}(x) = \begin{cases} \frac{1,33 * s_3}{10}, & s_3 \in [0; 10]; \\ \frac{1,33 * 20 - s_3}{10}, & s_3 \in [10; 20] \end{cases} \quad (6)$$

Где:

s_1 — процент использования первого доминирующего цвета;

s_2 — процент использования второго доминирующего цвета;

s_3 — процент использования остальных цветов.

Таким образом, гармония цветов выражается следующей формулой:

$$S = S_k + S_i + S_c \quad (7)$$

Видимость

Видимость зависит от двух типов элементов: ААА для элементов, которые должны привлекать внимание пользователя, и АА для остальных элементов [8]. Видимость определяется контрастом между яркостью цвета элемента и яркостью цвета его контейнера. Контраст выражается по формуле:

$$L = (L_1 + 0,05) / (L_2 + 0,05) \quad (8)$$

Где:

L_1 — яркость цвета элемента [0; 100];

L_2 — яркость цвета контейнера [0; 100].

В табл.1 приведены правила для определения видимости элемента по контрасту с его контейнером.

Критерий видимости оценивается до 20 баллов в соответствии с выражением:

$$C = \frac{N_{f-k-k_c}}{N_f} * 20 \quad (9)$$

Где:

N_f — общее количество элементов в интерфейсе;

k — количество элементов с низкой видимостью;

k_c — количество контейнеров без элементов ААА.

Шрифт

Шрифты интерфейса должны соответствовать следующим правилам [10]:

- Максимальное количество шрифтов 2.
- Высота строки (межстрочный интервал): не менее чем в 1,5 раза больше размера шрифта.
- Расстояние между абзацами не менее чем в два раза превышает размер шрифта.

- Расстояние между буквами (трельяж) должно быть не менее 0,12 размера шрифта.
 - Расстояние между словами должно быть не менее 0,16 размера шрифта.
- Оценка шрифта по этим правилам определяется следующим образом:

$$W = \left(\frac{8}{n_{\phi}} + \frac{n_{WL} - n_{W LX}}{n_{WL}} + 3 \frac{n_{WA} - n_{W AX}}{n_{WA}} + \frac{n_{WB} - n_{W BX}}{n_{WB}} + \frac{2n_M - n_{DLX} - n_{MT}}{n_M} \right) * 20 \quad (10)$$

Где:
 n_{WP} — количество шрифтов в интерфейсе;
 n_{WL} — количество строк;
 $n_{W LX}$ — количество строк с высотой менее 1,5 размера шрифта;
 n_{WA} — количество абзацев;
 $n_{W AX}$ — количество абзацев с пробелами менее двух размеров шрифта;
 n_{WB} — количество букв;

$n_{W BX}$ — количество букв с пробелом менее 0,12 размера шрифта;
 n_M — количество слов;
 n_{MT} — количество слов с размером шрифта менее 10px;
 n_{DLX} — количество слов с расстоянием от соседних менее 0,16 размера шрифта.
 Критерий качества шрифта оценивается из 20 баллов.

Таблица 1.

Правила видимости

Table 1.

Visibility rules

Тип элемента	Категория элемента (размер в пикселях)	Пропорция контрастности (цвет элемента и цвет его контейнера)
AA	размер > 19	3:1
	размер < 19	4:5
	другие	3:1
AAA	размер > 19	4:5
	размер < 19	7:1
	другие	4:5

Выравнивание

Выравнивание создает ощущение единства в графическом интерфейсе пользователя. Оно также позволяет размещать элементы иерархически. Оценка выравнивания основана на двух правилах [8]:

- считается, что каждый элемент расположен в контейнере на позиции x и y, при этом основным контейнером является экран;
- на каждом иерархическом уровне (внутри каждого контейнера) каждый элемент должен быть выровнен по вертикали, горизонтали или центру относительно своих соседей или своего контейнера.

Таким образом, критерий выравнивания выражается следующим образом:

$$E = 15 * (n_A / N) \quad (11)$$

Где:

n_A — количество элементов, не выровненных по вертикальной, горизонтальной или центральной оси;
 N — количество элементов на экране.

Визуальный баланс

Баланс интерфейса зависит от визуального веса элементов в интерфейсе. Воспринимаемый вес элемента – это мера того, что позволяет отличить один элемент от других. Поэтому этот показатель важен, так как помогает узнать, какие элементы могут быстро привлечь внимание пользователей в интерфейсе. Мера визуального веса зависит от трех характеристик [4]:

- цветовой контраст L элемента и его контейнера от 1 до 21;
- положение P элемента по отношению к ориентации считывания ($P = 5$, если элемент находится в зоне F или Z

шаблона Гутенберга [12] и $P = 2$ в противном случае);

- площадь T элемента (здесь каждый элемент рассматривается как прямоугольник)

$$T = \begin{cases} 10, H * W > 999 \\ H * W * 10^{-2}, H * W \leq 999 \end{cases} \quad (12)$$

где:

H — длина элемента;

W — ширина элемента.

Вес определяется в соответствии со следующим выражением:

$$V = L + P + T \quad (13)$$

Критерий визуального баланса оценивается из 20 баллов по следующей формуле:

$$V_B = \frac{N - 2V_S - V_G}{N} * 20 \quad (14)$$

где:

N — количество элементов в интерфейсе;

V_S — общее количество элементов всех групп с $V > 30$ (подсчитывается количество групп включающих в себя более чем 3 элемента);

V_G — количество групп без элементов с $V > 30$.

Пространство

Белое пространство — это пустая область вокруг, внутри или между элементами интерфейса. Без создания достаточного пространства между элементами нельзя спроектировать качественный дизайн. Он усиливает фокус, понимание, баланс, различия, взаимосвязи, визуальную иерархию и ритм дизайна.

Существует несколько способов оценки белого пространства, выбранный в данном случае — нелинейное программирование [5]. Приведенные ниже правила позволяют оценить белое пространство в интерфейсе:

- Для мобильных экранов отступ элемента контейнера должен составлять не менее 16 пикселей.

- Для планшетов и настольных компьютеров отступ элемента контейнера должен составлять не менее 24 пикселей.

- Минимальное расстояние между элементами должно составлять 8 пикселей.

Другой подход, используемый дизайнерами, заключается в том, что размеры пространства определяются золотым сечением.

Пространство оценивается из 20 баллов в соответствии с выражением:

$$H_S = 20 * (H/N) \quad (15)$$

Где:

H — количество элементов, которые не соблюдают не более одного правила пространства;

N — количество элементов на экране.

Пропорции

Пропорции элементов определяют элегантность в дизайне. Наиболее популярным правилом, используемым в графических интерфейсах для определения пропорций элементов, является «золотое» сечение [6]. Оно определяет возможные диапазоны размеров элементов в зависимости от размера экрана. В данной статье размеры ограничены уровнем 10 от размера экрана. Оценка критерия пропорции оценивается из 14 по следующей формуле:

$$P_r = 14 * (N_E - P_{rx}) / N_E \quad (16)$$

где:

N_E — количество элементов и пространств;

P_{rx} — количество элементов и пространств, в которых золотое сечение не представлено.

Единый стиль

Единый стиль — это критерий, который оценивает соблюдение единого стиля на всех экранах приложения. Таким образом, он зависит от трех других критериев, а именно гармонии цветов, шрифтов и пропорций. Единый стиль определяется по следующей формуле:

$$St = 14 * (P_A / 14 + S_A / 12 + W_A / 20) / 3 \quad (17)$$

Где:

P_A — значение критерия пропорции на всех экранах;

S_A — значение критерия гармонии цвета на всех экранах;

W_A — значение критерия шрифта на всех экранах.

Юзабилити

Критерии анализа юзабилити интерфейса [11], предполагающие их автоматическое определение по результатам проведения юзабилити тестирования, выбраны следующие:

- минимальное время для выполнения задания и время между взаимодействиями;

- фокус пользователя;

- щелчки, пропущенные пользователем, и его путешествие по интерфейсу;

- коэффициент ошибок по типам пользователей в каждом интерфейсе;

- видеозапись с субтитрами по заданиям.

После формализации правил дизайна и юзабилити переходим к проектированию автоматизированной системы.

2. Проектирование веб-приложения

2.1. Архитектура программного обеспечения

Архитектура веб-приложения показана на рис. 2. Для реализации была выбрана архитектура клиент-сервер и модель MVC. Серверная часть подразделяется на две части:

- основные контроллеры, состоящие из маршрутов для выполнения процессов приложения;

- аутентификация, авторизация и регистрация пользователя.

Представления также были разделены на две части: модули пользователя, зарегистрированного в приложении; модули участников тестирования.

Для сервера были использованы фреймворки Nestjs 8.4.7 (JavaScript) и Vert.x (Java) 4.2.1, а для части представления – VueJS 2 (JavaScript). Выбранная система управления базами данных – MongoDB версии 4.4.

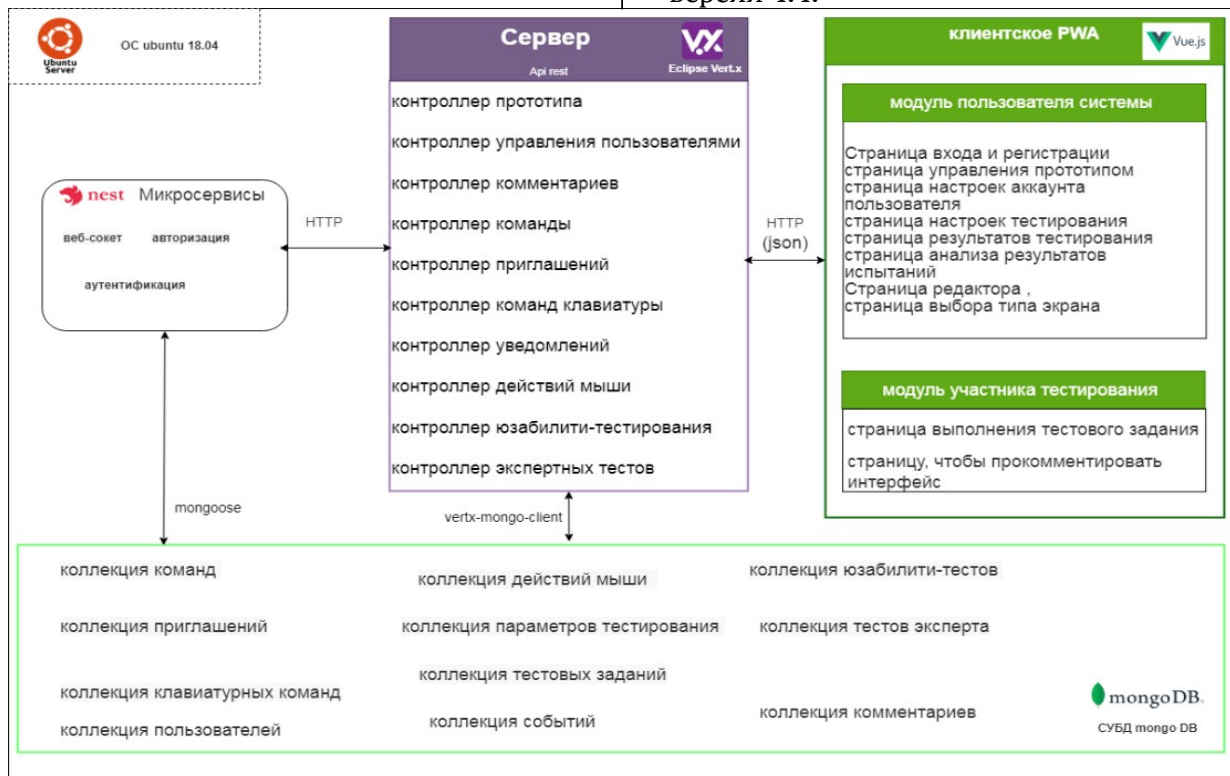


Рис. 2. Архитектура программного обеспечения

Fig. 2. Software architecture

2.2. Функциональное моделирование

Диаграмма состоит из 14 вариантов использования и 3 акторов: дизайнера, эксперта и участника тестирования. Дизайнер занимается редактированием интерфейсов, эксперт – созданием, настройкой тестов и оценка интерфейса, а участник – заполнением анкеты и выполнением тестовых заданий. Эксперт и дизайнер зарегистрированы в системе и имеют доступ по авторизации.

2.3. Разработка веб-приложения

Интерфейс оценивания качества элементов интерфейса представлен на рис. 4. В примере показана ошибка цветовой палитры в наборе интерфейсов, поэтому система предлагает

палитры, сформированные на основе теории цвета.

Интерфейс анализа результатов юзабилити-тестирования представлен на рис.5. Показан процент заданий, выполненных в интерфейсе участником тестирования. Разработанная система позволяет эксперту быстро оценить параметры интерфейса и предлагает рекомендации в случае низкого показателя или несоответствия эргономическим критериям. Для анализа результатов юзабилити-тестирования реализованы следующие функции: определение фокуса внимания пользователя, траектории движения указателя мыши, времени

выполнения задачи, а также определение

экранов, на которых пользователи сталкиваются с наибольшими трудностями.

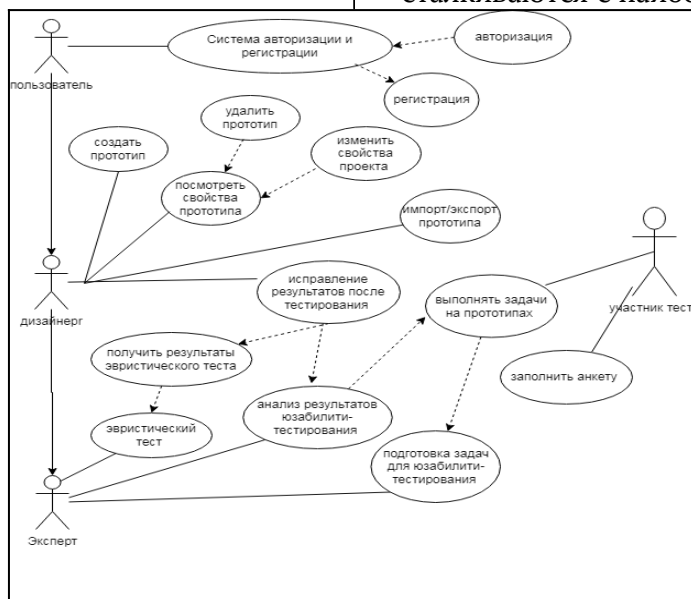


Рис. 3. Диаграмма вариантов использования
Fig. 3. Diagram of use cases

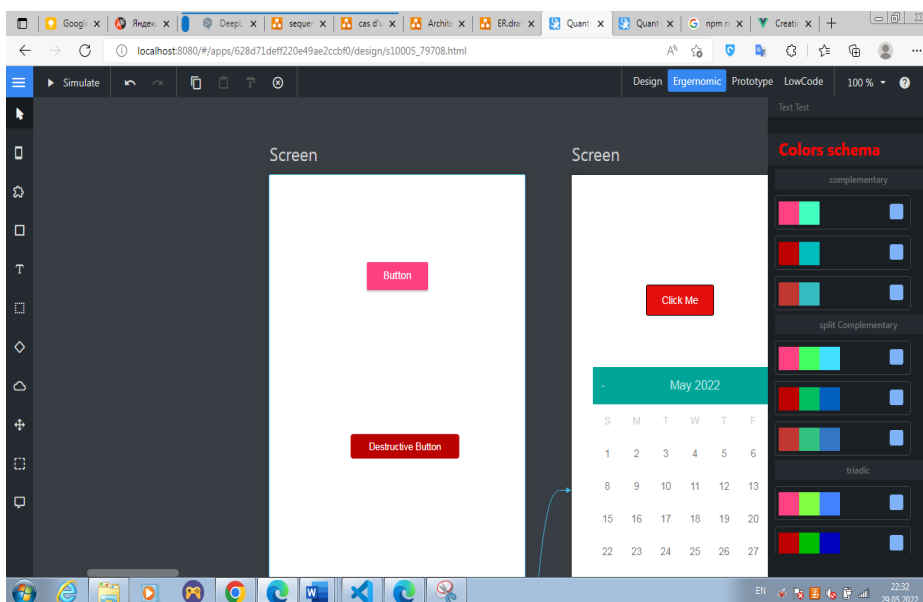


Рис. 4. Интерфейс оценивания элементов
Fig. 4. Element evaluation interface

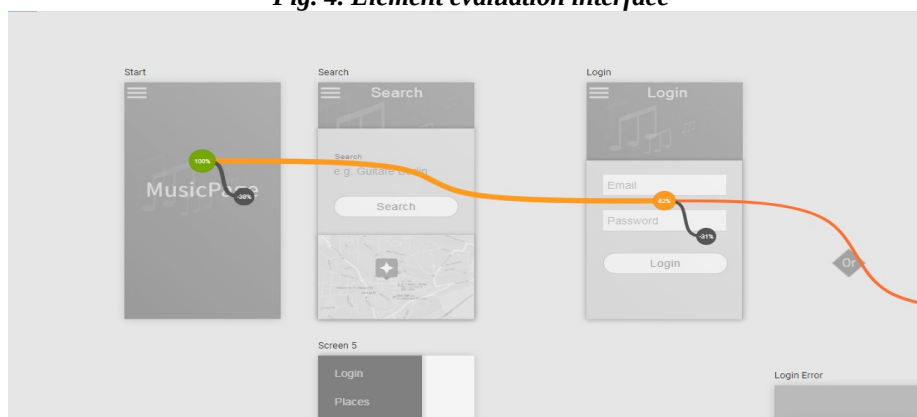


Рис. 5. Интерфейс анализа результатов юзабилити-тестирования
Fig. 5. Interface for analyzing the results of usability testing

Заключение

В результате выполненной работы была разработана автоматизированная система для создания и тестирования прототипов пользовательских интерфейсов.

По результатам анализа предметной области выявлена необходимость автоматизации процессов проектирования и тестирования пользовательских интерфейсов. Анализ аналогичных программных средств, используемых для автоматизированного проектирования и тестирования, выявил практическое отсутствие в них автоматизированной экспертной оценки качества интерфейса и узкую специализацию большинства из них только на проектирование или только на тестирование интерфейсов. По результатам анализа было сформулировано техническое задание.

На основе правил дизайна и эстетики была создана методика оценки элементов интерфейса, состоящая из методов проверки гармонии цветов, видимости элементов интерфейса, баланса интерфейса, соответствия текстов, соблюдения пропорций, пробелов между элементами,

выравнивания элементов и соблюдения уникального стиля в интерфейсе.

На этапе проектирования автоматизированной системы была выбрана клиент-серверная архитектура. Были разработаны алгоритмы для проектирования и проверки качества пользовательского интерфейса. Программная реализация автоматизированной системы осуществлялась на языке Java для серверной части и на JavaScript для клиентской части. Были использованы несколько модулей исходного кода инструментов Quant-UX, в том числе модули для редактирования прототипов, интерфейсных элементов и видеозаписи.

Разработанное веб-приложение способно: автоматизировать создание прототипов интерфейсов для всех типов экранов; планировать тесты и добавлять тестовые задания; проводить А/В-тестирование; определять фокус внимания пользователя на каждом интерфейсе; определять маршруты движения указателя мыши; определить среднее время выполнения заданий; записывать видео по шагам во время юзабилити-тестирования.

СПИСОК ИСТОЧНИКОВ

1. **Эссам М.Ф.Н.** Анализ итеративного проектирования графических пользовательских интерфейсов // III Всероссийская научно-практическая конференция для молодых ученых и студентов «Актуальные проблемы науки и образования в условиях современных вызовов»: сб. науч. тр. Пенза МНИЦ. 2022. С. 282–284. ISBN 978-5-00196-093-5.
2. **Бакаев М.А.** Современные тенденции в автоматизированной оценке юзабилити и поведенческие факторы в алгоритмах поисковых систем // Программные продукты и системы. 2017. №3. С. 447–455.
3. **Морозов А.Н.** Показатели и критерии оценки качества пользовательских интерфейсов систем безопасности: математическая модель и программная реализация // Охрана, безопасность, связь. 2020. № 5-1. С. 123–130.
4. **Дергачев К.В., Кузьменко А.А., Спасенников В.В.** Анализ взаимосвязи объекта и парадигмы исследования в эргономике с использованием информационных технологий // Эргодизайн. 2019. №1(03). С. 12–22. DOI 10.30987/article_5c518d8bd8e3d8.46297271
5. **Introduction to White Space, 2022.** URL:<https://uxengineer.com/principles-of-design/white-space/> (дата обращения: 20.09.2022).
6. **Брусенцова Т.П., Кишкурно Т.В.** Проектирование интерфейсов пользователя : пособие для студентов специальности 1-47 01 02 «Дизайн электронных и веб-изданий». Минск: БГТУ. 2019. 172 с. ISBN 978-985-530-799-1.
7. **Терехин С.Н., Минкин Д.Ю. Вострых А.В.** Алгоритм оценки гармоничности цветовой схемы

REFERENCES

1. **Essam M.F.N.** Analysis of Iterative Design of Graphical User Interfaces. In: Proceedings of the 3d All-Russian Scientific and Practical Conference for Young Scientists and Students: Actual Problems of Science and Education in the Face of Modern Challenges; Penza MNIC: 2022. p. 282-284.
2. **Bakaev M.A.** Modern Trends in Automated Usability Evaluation and Behavioral Factors in Search Engine Algorithms. Software Products and Systems. 2017;3:447-455.
3. **Morozov A.N.** Indicators and Criteria for Assessing User Interface Quality of Security Systems: a Mathematical Model and Software Implementation. Security, Safety, Communications. 2020;5-1:123-130.
4. **Dergachev K.V., Kuzmenko A.A., Spasennikov V.V.** Analysis of the Relationship between the Object and the Research Paradigm in Ergonomics Using Information Technology. Ergodesign. 2019;1(03):12-22. DOI 10.30987/article_5c518d8bd8e3d8.46297271.
5. **Introduction to White Space** [Internet]. 2022 [cited 2022 Aug 20]. Available from: <https://uxengineer.com/principles-of-design/white-space/>.
6. **Brusentsova T.P., Kishkurno T.V.** User Interface Design: a Manual for Students of the Specialty 1-47 01 02 “Design of Electronic and Web Publications”. Minsk: BSTU; 2019. 172 p.
7. **Terekhin S.N., Minkin D.Yu. Vostrykh A.V.** Algorithm for Assessing the Harmony of the Colour Scheme

графических пользовательских интерфейсов информационного обеспечения деятельности подразделений МЧС России // Научно-аналитический журнал Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России. 2021. № 3. С. 66-73.

8. **Lidwell W., Holden K., Butler J.** Universal Principles of Design // Rockport. 2010. 272 p. ISBN: 978-1-59253-587-3.

9. **Dergachev K., Dergacheva E.** Evaluating the ergonomics of online store user interfaces based on visual analytics // CEUR Workshop Proceedings : 31, Nizhny Novgorod, 27–30 сентября 2021 года. Nizhny Novgorod, 2021. P. 872-881.

10. **Understanding Success Criterion 1.4.4: Resize Text**, 2021. URL:<https://www.w3.org/WAI/WCAG21/Understanding/resize-text.html> (дата обращения 20.09.2022).

11. **Quantitative UX Analytics**, 2022. URL:<https://quant-ux.com/analytics.html#/analytics.html> (дата обращения: 20.09.2022).

12. **Visual Hierarchy, Gutenberg Diagram, F & Z Pattern**, 2019. URL:<https://lineindesign.medium.com/be-a-designer-who-can-also-help-with-writing-copy-2f4ea02a5646> (дата обращения: 20.09.2022).

Информация об авторах:

Дергачев Константин Владимирович – доцент, кандидат технических наук, тел. 89208357276, доцент кафедры «Информатика и программное обеспечение» БГТУ, декан факультета информационных технологий, международные идентификационные номера автора: Scopus-Author ID 57188860598, SPIN-код: 9404-3240, AuthorID: 310410.

Эссам Мвом Франсис Ноэль – магистрант направления «Программная инженерия» БГТУ, тел. 89038681211.

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации.

Contribution of the authors: the authors contributed equally to this article.

Авторы заявляют об отсутствии конфликта интересов.

The authors declare no conflicts of interests.

Статья поступила в редакцию 26.08.2022; одобрена после рецензирования 02.09.2022; принята к публикации 06.09.2022. Рецензент – Падерно П.И., доктор технических наук, профессор Санкт-Петербургского государственного технического университета («ЛЭТИ им. В.И. Ульянова-Ленина») член редсовета журнала «Эргодизайн».

The paper was submitted for publication on the 26th of August, 2022; approved after the peer review on the 2nd of September, 2022; accepted for publication on the 6th of September, 2022. Reviewer – Paderno P.I., Doctor of Technical Science, Professor of Saint Petersburg Electrotechnical University “LETI” named after V.I. Lenin, member of the editorial board of the journal “Ergodesign”.

of Graphical User Interfaces for Information Support of the Activities of Units of the Ministry of Emergency Situations of Russia. Bull of Saint-Petersburg University of State Fire Service of EMERCOM of Russia. 2021;3:66-73.

8. **Lidwell W., Holden K., Butler J.** Universal Principles of Design. Rockport Publishers; 2010. 272 p.

9. **Dergachev K., Dergacheva E.** Evaluating the Ergonomics of Online Store User Interfaces Based on Visual Analytics. In: CEUR Workshop Proceedings 31; 2021 Sep 27-30; Nizhny Novgorod; p. 872-881.

10. **Understanding Success Criterion 1.4.4: Resize Text** [Internet]. 2021 [cited 2022 Aug 20]. Available from: <https://www.w3.org/WAI/WCAG21/Understanding/resize-text.html>.

11. **Quantitative UX Analytics** [Internet]. 2022 [cited 2022 Aug 20]. Available from: <https://quant-ux.com/analytics.html#/analytics.html>.

12. **Visual Hierarchy, Gutenberg Diagram, F & Z Pattern** [Internet]. 2019 [cited 2022 Aug 20]. Available from: <https://lineindesign.medium.com/be-a-designer-who-can-also-help-with-writing-copy-2f4ea02a5646>.

Information about authors:

Dergachev Konstantin Vladimirovich – Associate Professor, Candidate of Technical Sciences, ph. 89208357276, Associate Professor of the Department “Computer Science and Software”, BSTU, Dean of the Faculty of Information Technologies, the author’s international identification numbers: Scopus-Author ID: 57188860598, SPIN-code: 9404-3240, AuthorID: 310410

Essam Mvom Francis Noel – master’s student of the direction “Software Engineering” BSTU, ph. 89038681211.