

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б. Н. Ельцина» (УрФУ)

Институт радиоэлектроники и информационных технологий – РТФ

Грибов Илья Николаевич РИ-130912

ОТЧЕТ

о проектной работе

по теме: Digital Portfolio 2024

по дисциплине: Проектный практикум 1A

Команда:Криптоинвесторы

Тимлид: Чеботарев Сергей Дмитриевич РИ-130931

Аналитик: Абдрахманов Даниял Сабырович РИ-130933

Дизайнер: Мешков Матвей Алексеевич РИ-130949

Фронтенд-разработчик: Грибов Илья Николаевич РИ-130912

Бэкенд-разработчик: Спирин Андрей Сергеевич РИ-130949

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc165369900)

[1. Целевая аудитория 4](#_Toc165369901)

[2. Определение проблемы 6](#_Toc165369902)

[3. Подходы к решению проблемы 9](#_Toc165369903)

[4. Анализ аналогов 11](#_Toc165369904)

[5. Календарный план проекта 12](#_Toc165369905)

[6. Сценарии использования 14](#_Toc165369906)

[7. Требования к продукту и к MVP 15](#_Toc165369907)

[8. Стек для разработки 17](#_Toc165369908)

[9. Прототипирование 19](#_Toc165369909)

[10. Проектирование и разработка системы 21](#_Toc165369910)

[ЗАКЛЮЧЕНИЕ 22](#_Toc165369911)

[СПИСОК ЛИТЕРАТУРЫ 23](#_Toc165369912)

[ПРИЛОЖЕНИЕ А 26](#_Toc165369913)

1. ВВЕДЕНИЕ

В эпоху цифровизации искусства и дизайна, когда творческие профессии переживают период значительных трансформаций, проект "Digital portfolio", созданный для предоставления художникам, артистам и дизайнерам инновационного приложения для создания и управления цифровым портфолио. Актуальность данной разработки обусловлена не только стремительным развитием цифровых технологий, но и возросшей потребностью в эффективных инструментах для самопрезентации и продвижения своего творчества в Интернете.

Современный рынок предлагает множество решений для создания цифровых портфолио, однако многие из них либо ограничены в функциональности, либо требуют от пользователей специальных навыков в области веб-дизайна и программирования. Таким образом, существует явная проблема доступности и удобства использования таких инструментов для широкого круга творческих личностей. Проект "Digital portfolio" направлен на устранение этого дисбаланса, предлагая решение, которое будет одинаково удобно как для опытных профессионалов, так и для начинающих артистов.

Цель проекта – разработать приложение "Digital portfolio", которое позволит художникам, артистам и дизайнерам легко создавать, обновлять и делиться своими работами, а также эффективно взаимодействовать с потенциальными заказчиками и поклонниками. На нашем сайте люди смогут спокойно связаться с автором понравившейся им работы, а художники смогут вести свой личный блог, где описывают работу над своими произведениями. Также на нашем сайте будет присутствовать поиск по тегам, где можно будет просмотреть работы из определенного жанра.

Для достижения поставленной цели были определены следующие задачи:

**Изучить** существующие решения для создания цифровых портфолио и выявить их ключевые недостатки и ограничения.

**Описать** требования и пожелания целевой аудитории к функционалу приложения "Digital portfolio".

**Установить** критерии удобства использования и доступности приложения для пользователей с различным уровнем технических навыков.

**Выявить** технологические возможности для интеграции современных инструментов визуализации и интерактивности в структуру приложения.

**Разработать** прототип приложения "Digital portfolio", включающий в себя базовый функционал для создания и управления цифровым портфолио.

Решение этих задач позволит создать приложение, которое станет надежным инструментом для творческих профессионалов и обеспечит им преимущества в продвижении их работ в цифровой среде. В конце работы будет представлено заключение, основанное на результатах решения поставленных задач и достигнутых целях проекта.

1. Целевая аудитория

Анализ целевой аудитории проекта "Digital portfolio" с использованием методики 5W Марка Шеррингтона:

**Что? (What?)**
Проект "Digital portfolio" предлагает художникам, артистам и дизайнерам инструмент для создания и управления цифровым портфолио. Сайт позволяет пользователям легко демонстрировать свои работы, обновлять их и делиться с потенциальными заказчиками и поклонниками.

**Кто? (Who?)**
Целевая аудитория приложения – творческие профессионалы всех возрастных категорий, полов и географических расположений. Особое внимание уделяется художникам, артистам и дизайнерам, которые стремятся к самопрезентации в цифровой среде и нуждаются в удобном инструменте для продвижения своего творчества.

**Почему? (Why?)**
Мотивация потребителей заключается в желании иметь доступный и удобный способ представления своих работ в цифровом формате. Сайт решает проблему отсутствия универсального и легкого в использовании инструмента, который бы соответствовал различным стилям и направлениям творчества.

**Когда? (When?)**
Потребители могут захотеть воспользоваться приложением "Digital portfolio" в любой момент своей творческой карьеры, особенно когда они ищут способы улучшить свою видимость и доступность для широкой аудитории, а также когда они готовы к взаимодействию с заказчиками и поклонниками.

**Где? (Where?)**
Сайт будет доступен для использования в любом месте, где у пользователей есть доступ к Интернету. Основные точки контакта с клиентом – это онлайн-платформы и социальные сети, где творческие профессионалы могут делиться своим портфолио и привлекать внимание к своему творчеству.



Рисунок 1 – Анализ целевой аудитории



Рисунок 2 – Продолжение



Рисунок 3 – Пожелания ЦА

1. Определение проблемы
Определение проблемы, с которой сталкивается проект "Digital portfolio", является ключевым этапом в разработке продукта, который будет максимально отвечать потребностям целевой аудитории. Для выявления основных проблем клиентов можно использовать следующие методы:

**Опросы и интервью** с реальными клиентами, чтобы узнать больше о их потребностях и проблемах.

**Изучение конкурентов** и отзывов их клиентов для выявления общих проблем в отрасли.

**Тематические форумы и социальные сети** для обнаружения вопросов и жалоб потенциальных или реальных клиентов.

**Анализ отзывов** на специальных сайтах и в жалобных книгах для определения болей клиентов.

**Анализ поисковых запросов** через сервисы, такие как Wordstat и Google Trends, для понимания, что ищут потенциальные клиенты.

Проблемы клиентов можно классифицировать по следующим категориям:

**Отсутствие ресурсов** (время, деньги, умения) для решения проблемы.

**Неудачные попытки решения** проблемы в прошлом.

**Недовольство** физическим и эмоциональным состоянием из-за текущей ситуации.

Также можно выделить три уровня "боли" клиентов:

**Технический уровень**: проблемы, вызванные бытовыми неудобствами, например, поломкой товара.

**Финансовый уровень**: проблемы, связанные с отсутствием спроса на продукцию, уменьшением прибыли и неэффективностью продаж.

**Личный интерес**: проблемы, решение которых может принести клиенту значительную экономию времени или другие личные выгоды.

Понимание этих аспектов поможет нам сформулировать предложение, которое будет решать реальные проблемы целевой аудитории и обеспечит успех проекта "Digital portfolio".

1. ПОДХОДЫ К РЕШЕНИЮ ПРОБЛЕМЫ

Старайтесь предлагать как можно больше идей альтернативных решений – лучше откинуть неподходящее решение на этапе анализа, чем упустить выгоду из-за малого количества рассмотренных вариантов. Для этого предлагается проводить «мозговые штурмы» с проектной командой, интервьюировать заказчиков и пользователей и использовать методики ТРИЗ (Теория Решения Изобретательских Задач). Тщательное изучение проблемы, критериев и целей является эффективным началом в выявлении альтернативных решений.

Подготовка нескольких альтернатив на ранних стадиях процесса позволяет повысить вероятность принятия адекватного решения с четким пониманием всех возможных последствий и рисков.

Важно отметить, что зачастую для выбора окончательного решения может понадобиться несколько итераций. На каждой итерации мы получаем все больше информации об анализируемой проблеме и лучше понимаем факторы, влияющие на принятие решения. При этом можно уточнять и изменять аналитические параметры (например, критерии и их веса).

Результат анализа покажет, насколько каждый из выбранных критериев реализован в каждой из рассмотренных альтернатив.

Очевидно, что при использовании структурированного метода анализа предпочтение отдается тому решению, которое наберет наибольшее количество очков по шкале соответствия выбранным критериям. Однако возможны ситуации, при которых результаты «взвешивания» нескольких решений могут оказаться очень близкими. В этом случае стоит провести дополнительный анализ, например, вовлекая экспертов и представителей заинтересованных сторон, выявляя и добавляя новые критерии выбора.

*Например*

*Чтобы сократить процесс согласования счетов, мы можем:*

1. Перераспределить задачи между согласующими. В результате несколько человек могут быть исключены из процесса. Суммарное время процесса сократится за счет периодов передачи данных/ожидания/коммуникации при передаче.
2. Перейти на электронный документооборот — достоверность счетов и данных в них будет подтверждена оператором ЭДО, подтверждение человеком не потребуется.
3. Автоматически распознавать сканы счетов и сравнивать данные с цифрами из системы закупок. Ручная проверка и согласование не потребуются.

Чтобы продумать все варианты, надо разобраться – а что же происходит сейчас? Как устроен процесс без вашей системы, как работают пользователи и заказчики? Даже если процесса еще нет, подробная информация про текущее состояние очень важна. Так мы поймем, какое решение устранит проблему, а не создаст еще одну.

1. Анализ аналогов

Прямые конкуренты – это аналогичное программное обеспечение на аналогичном рынке и работающий с целевой аудиторией, что и разрабатываемый продукт. Косвенные конкуренты – это похожее программное обеспечение с другими характеристиками или абсолютно другой продукт, но работающие с целевой аудиторией, что и разрабатываемый продукт.

Как правильно провести анализ конкурентов?

Рассмотрим шесть базовых шагов, которые надо выполнить при проведении подобной аналитики.

* определить цели анализа конкурентов;
* определить конкурентов;
* определить критерии анализа конкурентов;
* приступить к анализу (заполнение таблиц);
* сопоставить полученные данные;
* сделать выводы.

Используйте аналогичный продукты для того, чтобы заимствовать наиболее удачные приемы и способы решения проблем.

В таблице 2 представлен пример анализа конкурентов по трем критериям.

Таблица 2 – Анализ конкурентов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии сравнения | Разделим чек | Splitwise | Делить чек | Скинемся! |
| Многопользовательский режим | нет | да | да | да |
| Возможность сканировать фото | нет | нет | да | нет |
| Отдельное приложение | да | да | да | нет |

1. Календарный план проекта

Название проекта:

Руководитель проекта:

Таблица 1 – Календарный план

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название** | **Ответственный** | **Длительность** | **Дата начала** | **Временные рамки проекта** |
| 1 нед | 2 нед | 3 нед | 4 нед |
| ***Анализ*** |
| *1.1* | *Определение проблемы* |  |  |  |  |  |  |  |
| *1.2* | *Выявление целевой аудитории* |  |  |  |  |  |  |  |
| *1.3* | *Конкретизация проблемы* |  |  |  |  |  |  |  |
| *1.4* | *Подходы к решению проблемы* |  |  |  |  |  |  |  |
| *1.5* | *Анализ аналогов* |  |  |  |  |  |  |  |
| *1.6* | *Определение платформы и стека для продукта* |  |  |  |  |  |  |  |
| *1.7* | *Формулирование требований к MVP продукта* |  |  |  |  |  |  |  |
| *1.8* | *Определение платформы и стека для MVP* |  |  |  |  |  |  |  |
| *1.9* | *Формулировка цели* |  |  |  |  |  |  |  |
| *1.10* | *Формулирование требований к продукту* |  |  |  |  |  |  |  |
| *1.11* | *Определение задач* |  |  |  |  |  |  |  |
|  | *…* |  |  |  |  |  |  |  |
| ***Проектирование*** |
| *2.1* | *Архитектура системы (компоненты, модули системы)* |  |  |  |  |  |  |  |
| *2.2* | *Разработка сценариев использования системы* |  |  |  |  |  |  |  |
| *2.3* | *Прототипы интерфейсов* |  |  |  |  |  |  |  |
| *2.4* | *Дизайн-макеты* |  |  |  |  |  |  |  |
| *2.5* | *Архитектура системы (компоненты, модули системы)* |  |  |  |  |  |  |  |
|  | *…* |  |  |  |  |  |  |  |
| ***Разработка*** |
| *3.1* | *Написание кода* |  |  |  |  |  |  |  |
| *3.2* | *Тестирование приложения* |  |  |  |  |  |  |  |
|  | *...* |  |  |  |  |  |  |  |
| ***Внедрение*** |
| *4.1* | *Оформление MVP* |  |  |  |  |  |  |  |
| *4.2* | *Внедрение MVP* |  |  |  |  |  |  |  |
| *4.3* | *Написание отчета*  |  |  |  |  |  |  |  |
| *4.4* | *Оформление презентации* |  |  |  |  |  |  |  |
|  | *…* |  |  |  |  |  |  |  |
|  | *Защита проекта* |  |  | 07.06 - 15.06 |  |  |  |  |

1. Сценарии использования

Сценарий использования (use case) — описание поведения системы, когда она взаимодействует с кем-то (или чем-то) из внешней среды. Сценарий использования описывает, «кто» и «что» может сделать с рассматриваемой системой, или что система может сделать с «кем» или «чем». Методика сценариев использования применяется для выявления требований к поведению системы, известных также как пользовательские и функциональные требования.

Каждый сценарий использования сосредотачивается на описании того, как достигнуть цели или задачи. Для большинства программных проектов это означает, что потребуется множество сценариев использования, чтобы определить необходимый набор свойств новой системы.

Сценарий использования определяет взаимодействия между внешними агентами и системой, направленные на достижение цели. Актор представляет собой роль, которую играет человек или вещь, взаимодействуя с системой. Один и тот же человек, использующий систему, может быть представлен как различные акторы, потому что они играют различные роли.

Сценарии использования рассматривают систему как «черный ящик», и взаимодействия с системой, включая системные ответы, описываются с точки зрения внешнего наблюдателя.

Сценарий использования должен:

* описывать, что именно система должна сделать, чтобы актор достиг своей цели;
* не затрагивать деталей реализации;
* иметь достаточный уровень детализации.
1. Требования к продукту и к MVP

Требования систематизируются несколькими способами.

Требования клиентов

Клиенты – это те, кто выполняет основные функции системного проектирования, со специальным акцентом на пользователе системы как ключевом клиенте. Пользовательские требования определят главную цель системы.

Функциональные требования

Функциональные требования объясняют, что должно быть сделано. Они идентифицируют задачи или действия, которые должны быть выполнены. Функциональные требования определяют действия, которые система должна быть способной выполнить, связь входа/выхода в поведении системы.

Нефункциональные требования

Нефункциональные требования – требования, определяющие свойства, которые система должна демонстрировать, или ограничения, которые она должна соблюдать, не относящиеся к поведению системы. Например, производительность, удобство сопровождения, расширяемость, надежность, факторы эксплуатации.

Производные требования

Требования, которые подразумеваются или преобразованы из высокоуровневого требования. Например, требование для большего радиуса действия или высокой скорости может привести к требованию низкого веса.

*Например*

*Анкета должна содержать файл с фото, так как фото необходимо при оформлении документов – это бизнес-требование. А возможно, и бизнес-правило.*

*Из бизнес-требования следует, что у пользователя должна быть возможность прикрепить фото к анкете – это пользовательское требование. То есть требование, описывающее действия пользователя.*

*Получается, что система должна иметь функционал прикрепления фото к анкете – это уже функциональное требование, описывающее поведение системы. Или как должна работать система, чтобы выполнять исходное пользовательское требование.*

*Будем хранить все фото в формате base64 в отдельной таблице в БД – это нефункциональные требования.*

*Фото в очень хорошем качестве нам не нужно, а также мы не хотим покупать много памяти для сервера. Поэтому сделаем ограничение на размер загружаемого фото: не более 10 Мб.*

На каждое бизнес-требование, как правило, приходится несколько пользовательских. Пользовательское требование декомпозируется на какое-то число функциональных. К каждому функциональному требованию нужно продумать нефункциональные требования и ограничения.

Также нефункциональные требования и ограничения могут напрямую вытекать как из пользовательских требований, так и из бизнес-требований и правил.

Таким образом получаются деревья из требований, в вершине каждого из которых – бизнес-требование.

1. Стек для разработки

Стек для разработки приложения Digital Portfolio

Для создания эффективного и функционального приложения Digital Portfolio, предназначенного для художников, артистов и дизайнеров, мы выбрали следующий стек технологий:

Фронтенд:

React: Библиотека JavaScript для создания пользовательских интерфейсов с высокой отзывчивостью и интерактивностью.

Sass: Препроцессор CSS, который позволяет использовать переменные, вложенные правила и миксины для более гибкой стилизации.
Бэкенд:

Node.js: Серверная платформа, работающая на движке V8, позволяет использовать JavaScript и для серверной разработки.

Webpack: Сборщик модулей для современных JavaScript-приложений, который помогает управлять зависимостями и статическими ресурсами.

Этот стек был выбран за его гибкость, масштабируемость и широкое сообщество поддержки, что особенно важно для проектов творческой направленности, где требуется постоянное внедрение новых функций и возможностей.

1. Прототипирование

Процесс создания прототипа состоит из четырёх шагов:

* определение начальных требований;
* разработки первого варианта прототипа (в ПО. например, который содержит только пользовательский интерфейс системы);
* этап изучения прототипа заказчиком и конечным пользователем. Получение обратной связи о необходимых изменениях и дополнениях;
* переработка прототипа с учетом полученных замечаний и предложений.

*Качества, которыми должен обладать эффективный прототип*

Этап создания прототипа не должен быть затяжным.

Эффективные прототипы являются одноразовыми. Они предназначены для того чтобы донести идею до заинтересованного лица. После того как идея была донесена, прототип может быть отвергнут.

Эффективные прототипы являются сфокусированными, это означает что следует обращать внимание на сложные части при создании прототипов. Необходимо найти шаблонные взаимодействия, которые давно известны в теории юзабилити.

Необходимо обращать внимание на элементы взаимодействия, которые принесут пользу вашему продукту.

Даже самые опытные специалисты-разработчики допускают ошибки. Это правило особенно очевидно в коллективной работе. По мере выполнения проекта накапливаются мелкие ошибки, допущения, неверные решения. В результате получается плохой продукт при хорошей исходной идеи. Разработка прототипа – средство, позволяющее проанализировать идеи, прежде чем на них будут потрачены время и деньги. Это распространенный в инженерной практике метод. Основная цель, достигаемая при создании прототипа, – это экономия времени и ресурсов. Ценность прототипа заключается в том, что он является внешней оболочкой-моделью отражающей существенные отношения разрабатываемого продукта. По сравнению с реальным продуктом прототипы просты и недороги в разработке. При минимальном вложении средств можно обнаружить ошибки создателей и юзабилити проблемы, и улучшить пользовательский интерфейс до того, как сделаны значительные инвестиции в окончательную разработку и технологии.

Создание эффективного прототипа интерфейса является чрезвычайно важной задачей. Прототип должен хорошо выглядеть, чтобы понравиться заказчику и не вызвать вопросов у субъектов тестирования, он должен быть максимально дёшев, максимально полон и, что немаловажно, должен с лёгкостью обновляться.

Требования к прототипу изменяются со временем. Сначала наиболее актуальными его свойствами являются скорость создания и простота модификации. Эти свойства позволяют быстро разработать и проверить несколько версий интерфейса, при этом ещё и исправить значительную часть ошибок.

Затем на первый план выходят функциональность и эстетичность, простота же модификации уже не столь важна, поскольку с каждой новой исправленной ошибкой снижается вероятность того, что прототип придётся полностью переделывать при обнаружении новой ошибки.

Поэтому всегда правильно сделать прототип настолько похожим на результирующую систему, насколько позволяет самая поздняя его версия. Первый прототип стоит делать максимально примитивным. Только после того, как тестирование подтверждает его правильность, стоит сделать более детализированный прототип.

Прототипы интерфейсов возможно привести в приложении к отчету.

1. Проектирование и разработка системы

На этом этапе формируется общая структура программного комплекса. В соответствии с технологией нисходящего структурного программирования программный комплекс разбивается на небольшие части – программные модули (блоки). Для каждого программного модуля формулируются требования по реализуемым функциям и разрабатывается алгоритм, реализующий эти функции. Необходимо привести описание этих модулей, а также задач, которые они решают.

Необходимо привести алгоритм работы приложения, т.е. точного предписания и последовательности действий, заложенных в логике продукта.

Также на этом этапе можно привести схему взаимодействия программных модулей, т.е. схема потоков данных программы.

В этой части можно привести описание самого процесса разработки разнотипных сложных блоков программного обеспечения.

ЗАКЛЮЧЕНИЕ

Заключение должно содержать общие выводы, обобщенное изложение основных проблем. Могут быть указаны положительные и отрицательные стороны разработанного решения. Должны быть указаны перспективы дальнейшей разработки темы. Примерный объем заключения 2-3 страницы.

В заключительной части работы должно быть дано последовательное, логическое изложение полученных результатов в соответствие поставленными изначально задачами.

На данный момент сервис готовится к запуску и тестированию на реальных пользователях.

СПИСОК ЛИТЕРАТУРЫ

1. Веб-технологии для разработчиков Обзор протокола HTTP / Веб-технологии для разработчиков. – 2022. – URL: https://developer.mozilla.org/ru/docs/Web/HTTP/Overview (дата обращения: 30.04.2022).
2. Вигерс К. Разработка требований к программному обеспечению / К. Вигерс, Дж. Битти. – 3-е изд. – Москва: Русская редакция, 2014. – 736 с. – ISBN 978-5-9909805-3-2.
3. Евлантьев С.Карта эмпатии клиента / Станислав Евлантьев. – 2022. – URL: https://in-scale.ru/blog/karta-empatii/ (дата обращения: 01.05.2022).
4. Ильяхов М. Пиши, сокращай: как создать сильный текст / М. Ильяхов, Л. Сарычева. – 3-е изд. – Москва: Альпина Паблишер, 2022. – 440 с. – ISBN 978-5-9614-6526-6.
5. Кинзябулатов Р. IDEF0. Знакомство с нотацией и пример использования / Рамиль Кинзябулатов. – 2022. – URL: https://trinion.org/blog/idef0-znakomstvo-s-notaciey-i-primer-ispolzovaniya (дата обращения: 30.04.2022).
6. Коберн А. Современные методы описания функциональных требований к системам / А. Коберн. – Москва: Издательство «Лори», 2012. – 264 с. – ISBN 978-5-85582-326-4.
7. Мидоус Ф. Введение в протоколы HTTP и HTTPS / Филлип Мидоус. – 2022. – URL: https://docs.microsoft.com/ru-ru/azure/rtos/netx-duo/netx-duo-web-http/chapter1 (дата обращения: 05.05.2022).
8. Сырцев В. Использования диаграммы вариантов использования UML при проектировании программного обеспечения / Вадим Сырцев. – 2021. – URL: https://habr.com/ru/post/566218/ (дата обращения:30.04.2022).
9. Фирстов Л. О развитии навыка говорения / Леонид Фирстов. – 2020. – URL: https://habr.com/ru/post/494424/ (дата обращения: 30.04.2022).
10. Фитцпатрик Р. Спроси маму: Как общаться с клиентами и подтвердить и подтвердить правоту своей бизнес-идеи, если все кругом врут? / Р. Фитцпатрик. – Москва : Альпина Паблишер, 2021. – 160 с. – ISBN 978-5-9614-3045-5.
11. Шейнблат В. Естественный подход к освоению языка / Владимир Шейнблат. – 2022. – URL: https://englishsimple.ru/articles/natural-approach-to-language-learning/ (дата обращения: 25.04.2022).
12. Шторкин С. Средневзвешенная система голосования / Семен Шторкин – 2022. – URL: https://habr.com/ru/post/63664/ (дата обращения: 30.04.2022).
13. Apple MacOS / Apple. – 2022. – URL: https://www.apple.com/za/macos/what-is/ (дата обращения: 30.04.2022).
14. Best Programmer Функциональные и нефункциональные требования: полное руководство / Best Programmer. – 2021. – URL: https://bestprogrammer.ru/izuchenie/funktsionalnye-i-nefunktsionalnye-trebovaniya-polnoe-rukovodstvo (дата обращения: 15.05.2022).
15. Google Chrome Браузер от Google / Google Chrome. – 2022. – URL: https://www.google.ru/chrome/ (дата обращения: 30.04.2022).
16. HeyLady Женское сообщество для создания конференций и практики английского языка / HeyLady. – 2022. – URL: https://www.heylady.io/ (дата обращения: 30.04.2022).
17. Lingbe Бесплатные звонки с носителями в любое время / Lingbe. – 2022. – URL: https://lingbe.com/ (дата обращения: 30.04.2022).
18. PostgrePro Что такое PostgreSQL? / PostgrePro. – 2020. – URL: https://postgrespro.ru/docs/postgresql/12/intro-whatis (дата обращения: 30.04.2022).
19. Proglib Об Agile / Denver. – 2022. – URL: https://proglib.io/p/klyuchevye-razlichiya-mezhdu-agile-scrum-i-kanban-2022-02-24 (дата обращения 05.04.2022).
20. SimilarWeb Аналитика трафика Tandem / SimilarWeb. – 2022. – URL: https://www.similarweb.com/ru/website/tandem.net/#traffic (дата обращения: 30.04.2022).
21. Tandem / Tandem.net. – 2022. – URL: https://www.tandem.net/ru (дата обращения: 30.04.2022).
22. JavaScript WebSocket / JavaScript. – 2022. – URL: https://learn.javascript.ru/websocket (дата обращения: 02.05.2022).
23. VC.RU Анализ конкурентов на практике / VC.RU. – 2020. – URL: https://vc.ru/marketing/181297-analiz-konkurentov-na-praktike-10-shagov-shablon (дата обращения: 30.04.2022).
24. VC.RU Как составить портрет клиента (целевой аудитории) / VC.RU. – 2020. – URL: https://vc.ru/marketing/156147-kak-sostavit-portret-klienta-celevoy-auditorii-instrukciya-s-primerami (дата обращения: 30.04.2022).

ПРИЛОЖЕНИЕ А

