



Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Хакасский технический институт – филиал ФГАОУ ВО  
«Сибирский федеральный университет»

Кафедра прикладной информатики, математики и естественно-научных  
дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Е. Н. Скуратенко  
подпись  
« \_\_\_\_ » \_\_\_\_\_ 2021 г.

**ЗАДАНИЕ  
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ  
в форме бакалаврской работы**

Студенту Котуленко Василию Сергеевичу

Группа ХБ 17-03

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка Веб-приложения для обучения основам HTML и CSS

Утверждена приказом по институту № 222 от 08.04.2021 г.

Руководитель ВКР: А.Н. Таскин, доцент, канд. физ.-мат. наук, ХТИ – филиал СФУ

Исходные данные для ВКР: заказ ХТИ – филиала СФУ.

Перечень разделов ВКР:

1. Анализ предметной области и выбор средств проектных решений.
2. Описание процесса разработки веб-приложения.
3. Расчет затрат и оценка эффективности реализации проекта «Разработка Веб-приложения для обучения основам HTML и CSS».

Перечень графического материала: нет

Руководитель ВКР

\_\_\_\_\_

подпись

А.Н. Таскин

Задание принял к исполнению

\_\_\_\_\_

подпись

В.С. Котуленко

«08» апреля 2021 г.

## РЕФЕРАТ

Выпускная квалификационная работа (ВКР) на тему «Разработка веб-приложения для обучения основам HTML и CSS» содержит 62 страницы текстового документа, 10 формул, 9 таблиц, 45 рисунков, 19 использованных источников.

ИНФОРМАЦИОННАЯ СИСТЕМА, ВЕБ-ПРИЛОЖЕНИЕ, АНАЛИЗ, IDEF, ЗАТРАТЫ, ЭФФЕКТИВНОСТЬ, РИСК, ОБУЧЕНИЕ, ОНЛАЙН.

Объект выпускной квалификационной работы: процесс разработки веб-приложения.

Предмет выпускной квалификационной работы: разработка веб-приложения для обучения основам HTML и CSS.

Актуальность данной работы заключается в том, что разработка сайтов в настоящее время является перспективной сферой занятости, так как существует большая потребность в квалифицированных разработчиках. Поэтому разработка веб-приложения покажет абитуриентам, поступающим на направление «Прикладная информатика», чему они могут научиться.

Целью написания ВКР является: разработка веб-приложения для привлечения абитуриентов во время профориентационных мероприятий, а также при поступлении в ХТИ – филиал СФУ по направлению «Прикладная информатика».

Основными задачами ВКР являются:

- проанализировать существующие системы для привлечения абитуриентов;
- проанализировать существующие системы для обучения HTML и CSS;
- проанализировать и выбрать средства разработки веб-приложения;
- разработать веб-приложение;
- провести оценку экономической эффективности веб-приложения.

В результате было разработано веб-приложение, которое выполняет функцию обучения основам HTML и CSS, а также привлекает абитуриентов на направление «Прикладная информатика»

## SUMMARY

The theme of the graduation thesis is «Web Application Development for Teaching the Basics of HTML and CSS». It contains 62 pages of a text document, 10 formulae, 9 tables, 45 figures, 19 reference items.

IT SYSTEM, WEB APPLICATION. ANALYSIS, IDEF, COST, EFFICIENCY, RISKS, TRAINING, ONLINE.

The object of the graduation paper: the process of developing a web application.

The subject of the graduation paper: web application development for teaching the basics of HTML and CSS.

The relevance of the thesis: website development is currently a promising area of employment, since there is a great need for qualified developers, so the development of a web application will show applicants entering the “Applied Informatics” Training Program what they can learn.

The purpose of the thesis: to develop a web application to attract applicants to apply for KhTI – branch of SibFU, the “Applied Informatics” Training Program.

The objectives of the graduation paper are:

- analysis of existing systems for attracting applicants;
- analysis of existing systems for learning HTML and CSS;
- to analyze and to select web application development tools;
- to develop a web application;
- to calculate economic efficiency of the web application.

Results: the web application has been developed that performs the function of teaching the basics of HTML and CSS, and helps to attract applicants to study at the “Applied Informatics” Training Program as well.

English language supervisor

\_\_\_\_\_

N.V. Chezybaeva

signature, date

## СОДЕРЖАНИЕ

Введение.....	8
1 Анализ предметной области и выбор средств проектных решений	10
1.1 Организационно-экономическая характеристика предметной области	10
1.2 Принципы функционирования реализуемой информационной системы	12
1.3 Анализ предметной области проектируемой информационной системы	12
1.4 Описание работы веб-приложения .....	18
1.5 Обоснование и выбор средств проектных решений .....	20
1.6 Моделирование бизнес-процесса «Разработка веб-приложения для обучения основам HTML и CSS» .....	21
1.7 Выводы по разделу «Анализ предметной области и выбор средств проектных решений» .....	23
2 Описание процесса разработки веб-приложения.....	24
2.1 Описание работы пользователя с информационной системой в нотации IDEF3 .....	24
2.2 Установка и настройка ПО .....	25
2.3 Создание главного меню и меню выбора задания .....	26
2.4 Создание прототипа заданий .....	30
2.5 Реализация функции перемещения элементов .....	31
2.6 Реализация стилизации элементов.....	35
2.7 Разработка остальных заданий .....	38
2.8 Выводы по проектному разделу.....	45
3 Расчет затрат и оценка эффективности реализации проекта «Разработка Веб-приложения для обучения основам HTML и CSS» .....	45
3.1 Капитальные затраты .....	46
3.2 Расчет эксплуатационных затрат .....	51

3.3	Оценка риска реализации проекта создания ИС .....	54
3.4	Определение доходов и экономической эффективности реализации проекта «Разработка веб-приложения для обучения языкам программирования HTML и CSS».....	55
3.4.1	Анализ рынка продуктов-аналогов .....	56
3.4.2	Источники получения доходов.....	56
3.5	Вывод по разделу «Расчет затрат и оценка эффективности реализации проекта «Разработка веб-приложения для обучения основам HTML и CSS»	56
	Заключение .....	58
	Список сокращений .....	59
	Список использованных источников .....	60



## ВВЕДЕНИЕ

Frontend-разработчик – одна из самых популярных и востребованных профессий, в данный момент времени, в IT сфере. Это именно тот человек, который картинку макета превратит в код для браузера. Сделает его привлекательным на вид и удобным для пользователя. Сфера IT расширяется, из-за этого растет востребованность frontend-разработчиков.

В настоящее время происходит активный процесс информатизации общества. Все больше людей переводят, например, свой бизнес в IT сферу, создают онлайн-каталоги товаров, и даже целые магазины, что значительно упрощает процесс поиска и покупки необходимого.

Целью написания ВКР является: разработка веб-приложения для привлечения абитуриентов во время профориентационных мероприятий, а также при поступлении в ХТИ – филиал СФУ по направлению «Прикладная информатика».

Для достижения этой цели были поставлены следующие задачи:

- проанализировать существующие системы для привлечения абитуриентов;
- проанализировать существующие системы для обучения HTML и CSS;
- проанализировать и выбрать средства разработки веб-приложения;
- разработать веб-приложение;
- провести оценку экономической эффективности веб-приложения.

В первом разделе будет проведен анализ предметной области и обоснован выбор программных средств.

Во втором разделе подробно описан процесс создания веб-приложения.

В третьем разделе будут рассчитаны затраты на разработку, а также экономическая эффективность от реализации проекта.

В итоге будет разработано веб-приложение, которое будет обучать основам HTMLи CSS, а также может повысить интерес к направлению «Прикладная информатика» в ХТИ – филиале СФУ.

# **1 Анализ предметной области и выбор средств проектных решений**

## **1.1 Организационно-экономическая характеристика предметной области**

Заказчиком является Хакасский технический институт – филиал федерального государственного автономного образовательного учреждения высшего образования «Сибирский федеральный университет». Заказчику требуется программный продукт, который будет привлекать внимание абитуриентов, которые пришли на различные мероприятия, организованные в ХТИ – филиале СФУ, путем выполнения нескольких заданий, направленных на получение основных знаний HTML и CSS, тем самым привлекая абитуриентов, которым интересна разработка сайтов.

Краткое наименование: ХТИ – филиал СФУ.

Адрес: Республика Хакасия, г. Абакан, ул. Щетинкина 27.

Контактный телефон: +7 (3902) 22-53-55.

Основной задачей ХТИ – филиала СФУ является создание передовой образовательной, научно-исследовательской и инновационной инфраструктуры, продвижение новых знаний и технологий для решения задач социально-экономического развития Сибирского федерального округа, а также формирование кадрового потенциала — конкурентоспособных специалистов по приоритетным направлениям развития Сибири и Российской Федерации, соответствующих современным интеллектуальным требованиям и отвечающих мировым стандартам [13].

Учреждение было создано 30 декабря 1967 года по приказу №550 Министерства высшего и среднего специального образования РСФСР. Учредитель – Российская Федерация. Функции и полномочия учредителя осуществляет Министерство науки и высшего образования Российской Федерации.

Организационная структура ХТИ – филиала СФУ представлена на рисунке 1.

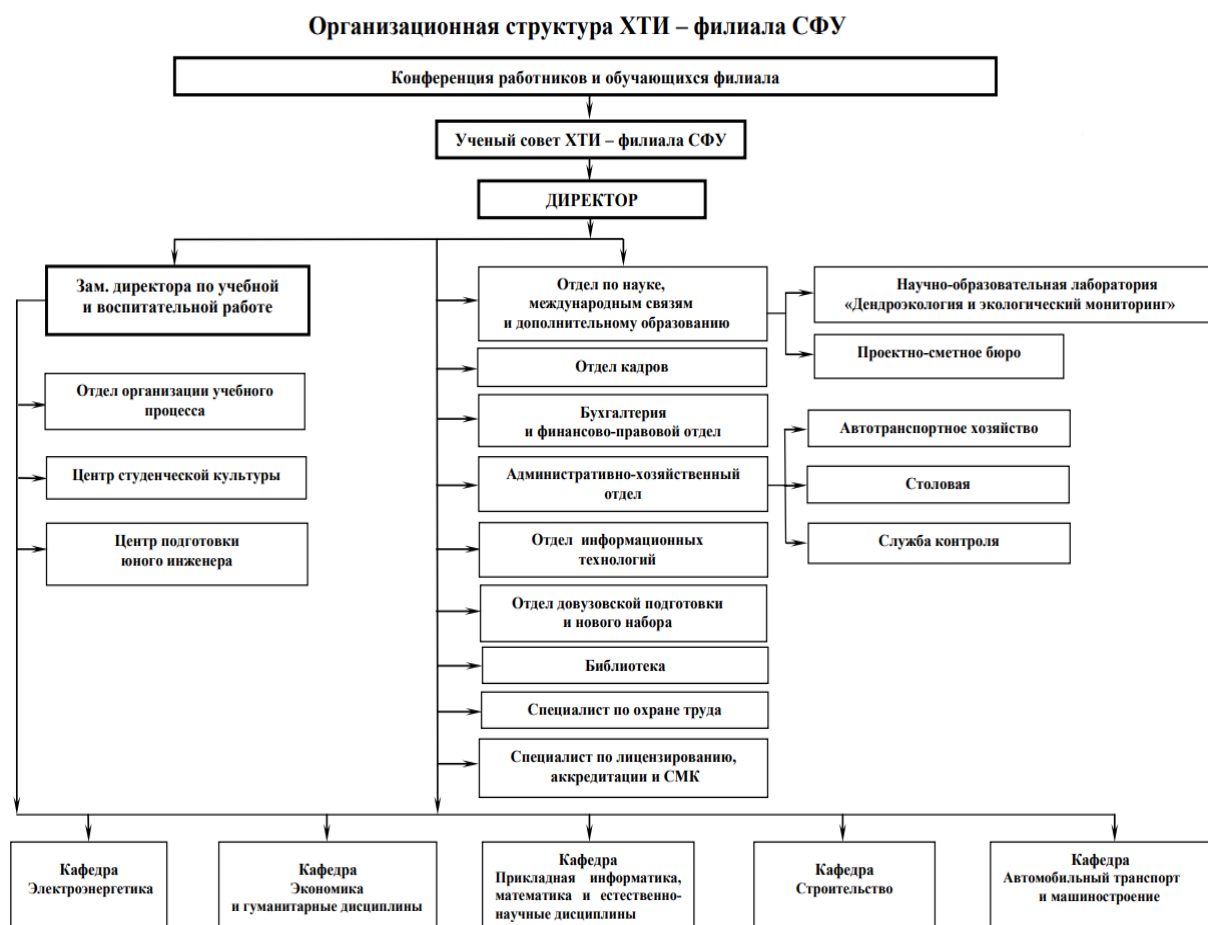


Рисунок 1 – Организационная структура ХТИ – филиала СФУ

Разрабатываемое веб-приложение направлено на повышение интереса к направлению «Прикладная информатика». Обучение проводится в очной форме. Ведется подготовка будущих специалистов, которые будут выполнять работу по созданию, модификации, внедрению и сопровождению информационных систем и управление этими работами, разрабатывать проекты автоматизации и информатизации прикладных процессов и создавать информационные системы в прикладных областях [14].

Также институт проводит мероприятия, направленные на привлечение будущих абитуриентов. Этим занимается отдел довузовской подготовки и

нового набора. Этот отдел организует проведение рекламных компаний по набору на первый курс, разрабатывает мероприятия и многое другое.

В настоящее время разработчики сайтов и веб-приложений очень востребованы, но поступающие абитуриенты могут не знать об этом, думать, что обучаться этому придется самому. Поэтому данное веб-приложение привлечет абитуриентов, которые заинтересованы в этом.

## **1.2 Принципы функционирования реализуемой информационной системы**

Веб-приложение – это клиент-серверное приложение, которое доступно с любого ПК с доступом в интернет, а также в любом браузере. Веб-приложение не занимает место на ПК, а также не требует скачивания.

Реализовано 5 легких заданий, которые для незнающих абитуриентов имеют сжатые теоретические материалы, которые направлены на обучение основам HTML и CSS. А также для закрепления теоретического материала имеется практическая часть, в которой абитуриент может решить задание, которое проверится, и, если задание правильно сделано, даст допуск к следующему заданию.

Задания сформированы так, что с каждым следующим заданием уровень сложности повышается, но задания остаются простыми для быстрого их решения, чтобы больше абитуриентов могли попробовать свои знания, либо получить новые, во время мероприятий для привлечения абитуриентов.

## **1.3 Анализ предметной области проектируемой информационной системы**

Практика привлечения абитуриентов в ВУЗ, является одной из основных задач ВУЗа. Для этого разрабатываются системы, которые показывают направления обучения, а также знакомят абитуриента с ВУЗом. Одним из

самых популярных ВУЗов, в котором используются интерактивные игры, является ТПУ. На сайте интернет-лица ТПУ имеется три игры, направленные на привлечение будущих абитуриентов, это:

– Интерактивный обучающий курс «Экспедиция» (рис. 2) – игра, предоставляющая информацию о полезных ископаемых, их свойствах и особенностях.

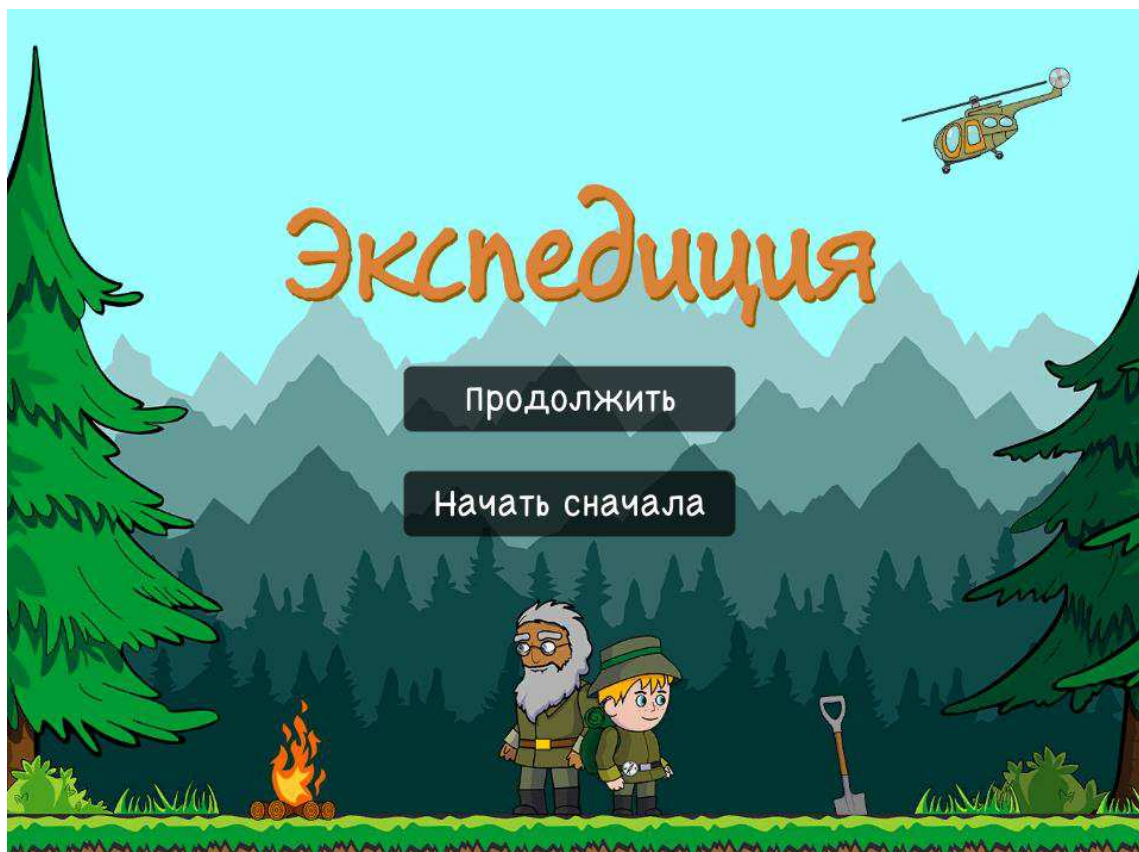


Рисунок 2 – Главное меню приложения «Экспедиция»

– Научная игра «Пилигримы» (рис. 3) – игра, показывающая геологический, биологический и химический мир прошлого времени.

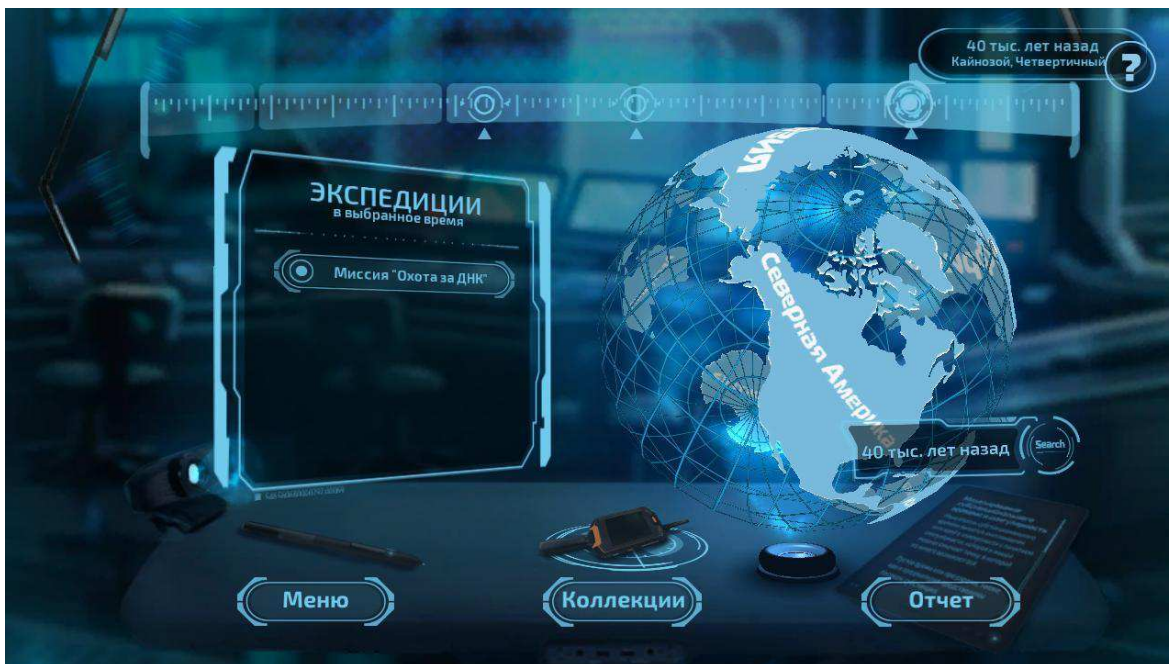


Рисунок 3 – Главное меню приложения «Пилигримы»

– Join.TPU (рис. 4) – показывает то, что ждет абитуриента в стенах ВУЗа, проводит знакомство с ВУЗом [11].

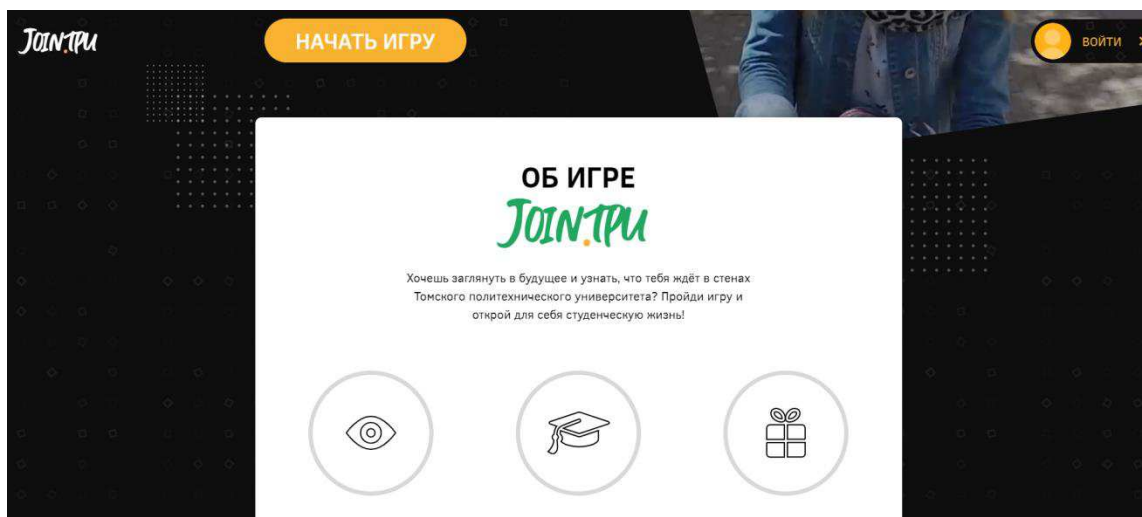


Рисунок 4 – Главное меню приложения «Join.TPU»

Эти приложения объединяет то, что все они относятся к жанру квестов. Обладают интерактивными свойствами. Благодаря этому жанру, пользователя данных приложений постоянно привлекает следующий шаг, ему становится интересно продолжить выполнять поставленные задания. Для большего

повышения интереса по итогам игры – при поступлении в ТПУ, будут начислены до 10 дополнительных баллов. Именно поэтому данные системы отлично справляются с привлечением абитуриентов в ВУЗ. Например, к Join.TPU присоединилось уже 7840 абитуриентов со всей России.

Также, было выделено три веб-приложения, которые направлены на обучение HTMLи CSS.

– Flexboxfroggy (рис. 5) – игра, которая направлена на изучение CSS Flexbox, в которой игрок, учится применять flexbox свойства для позиционирования объектов на странице.

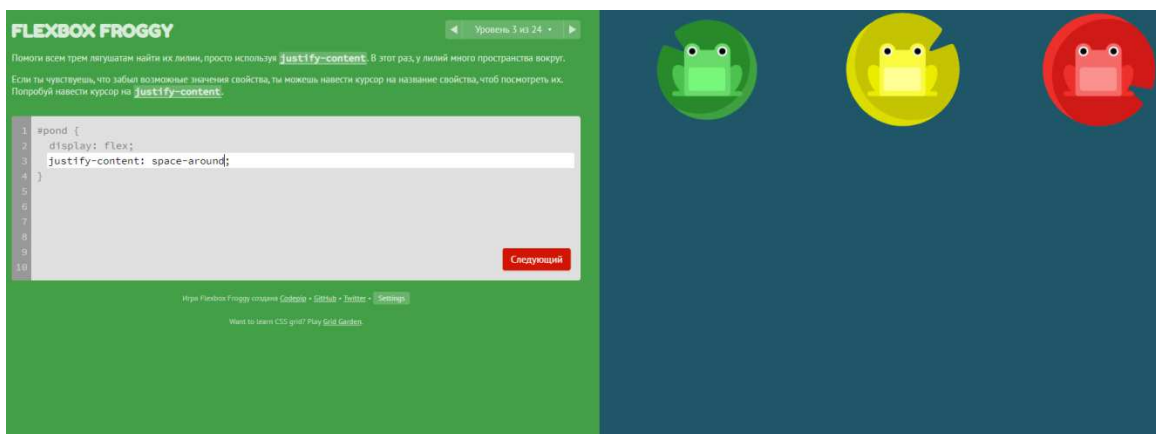


Рисунок 5 – Окно выполнения задания «Flexboxfroggy»

– CSS Dinner (рис. 6) – простая игра, которая поможет освоить CSS и освежить в памяти базовые правила верстки. В CSS Dinner игрок с помощью правильной команды – игра помогает инструкциями – должен расположить на столе посуду и продукты [1].



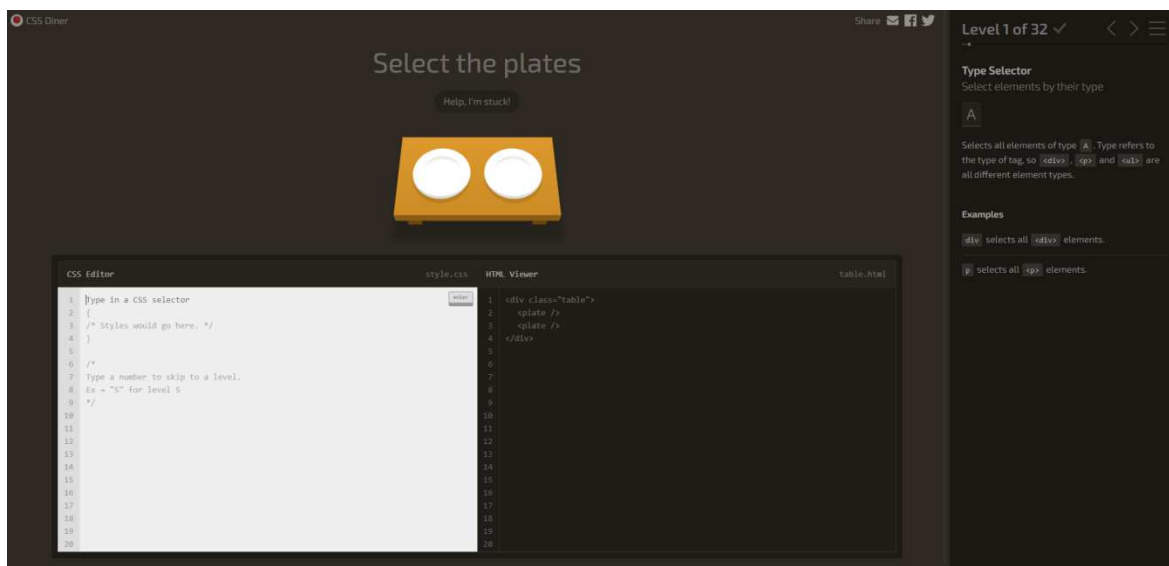


Рисунок 6 – Окно выполнения задания «CSSDinner»

– Яндекс.практикум (рис. 7) – веб-сервис, в котором предоставлены курсы на различные направления в IT сфере, имеются бесплатные курсы, которые показывают основы, а также платные, в которых более глубоко можно изучить нужное направление.

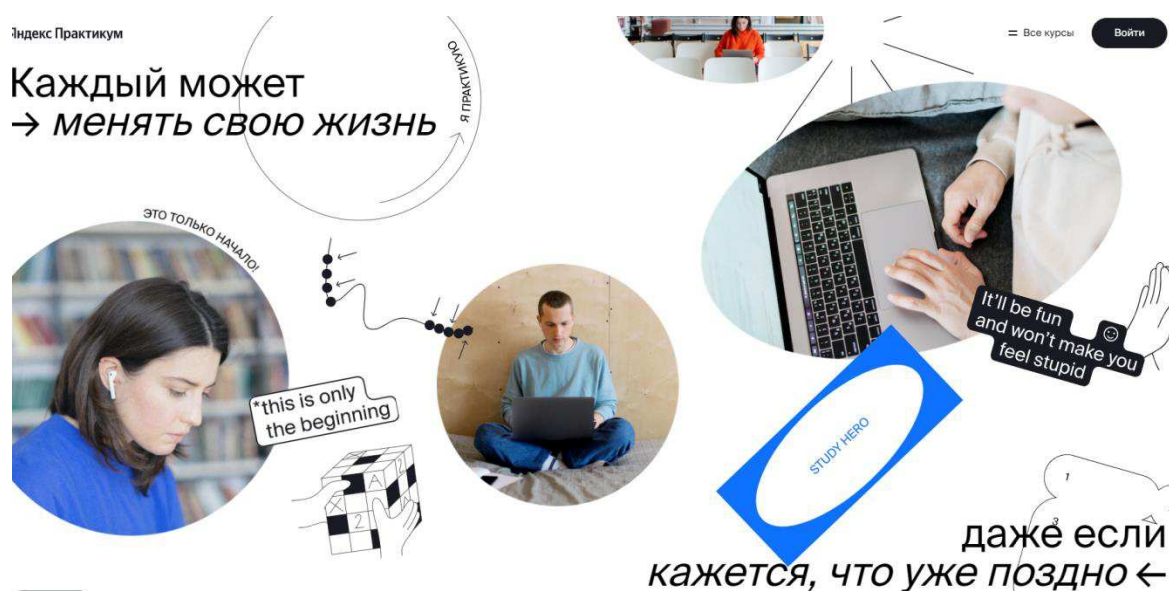


Рисунок 7 – Главное меню «Яндекс.Практикум»

Данные веб-приложения направлены на обучение HTML и CSS, задания поставлены так, чтобы с каждым последующим заданием, пользователю было все сложнее справиться с ним. Первые два приложения имеют бесплатную

основу, но они узко направлены на изучение CSS. Последнее приложение имеет платную основу, но при этом оно имеет множество курсов и заданий, а также пробные задания для того, чтобы понять какое направление пользователю более интересно.

Таблица 1 – Сравнение приложений

Приложение	Привлечение абитуриентов	Обучение	Изучение кодирования	Интерактивность	Цена
Экспедиция	+	+	-	+	бесплатно
Пилигримы	+	+	-	+	бесплатно
Join.TPU	+	+	-	+	бесплатно
Flexboxfroggy	-	+	+	+	бесплатно
CSSDinner	-	+	+	+	бесплатно
Яндекс.практикум	-	+	+	+	от 100т.р

Из таблицы 1 становится ясно, что приложения, созданные для привлечения внимания абитуриентов, не обучают кодированию. А приложения, созданные для обучения кодированию, не могут привлечь абитуриентов, так как они были созданы не для этой цели.

Также все приложения обладают интерактивным свойством, так как пользователь взаимодействует с системой, от выбора действий человека происходят определенные действия.

Поэтому создаваемое приложение должно объединить в себе, как привлечение абитуриентов, так и изучение кодирования, а также обладать интерактивностью.

Но во всех приложениях, которые обучают кодированию отсутствует визуальное (графическое) кодирование. Данный вид кодирования избавит пользователя от ввода кода, пользователь будет взаимодействовать с доступными элементами.

Визуальное кодирование – это способ создания программы для ЭВМ путём манипулирования графическими объектами вместо написания её текста. Некоторые авторы представляют визуальное программирование как следующий этап развития языков программирования [8].

Одним из самых популярных приложений, основанных на визуальном кодировании, является Scratch. Scratch – визуальная событийно-ориентированная среда программирования, созданная для детей и подростков [9]. Используется для обучения детей. Главным его преимуществом является визуальное кодирование. Благодаря этому, программирование становится доступным для новичков и для тех, кто вообще не связан с информатикой. Окно с визуальным программированием показано на рисунке 8.

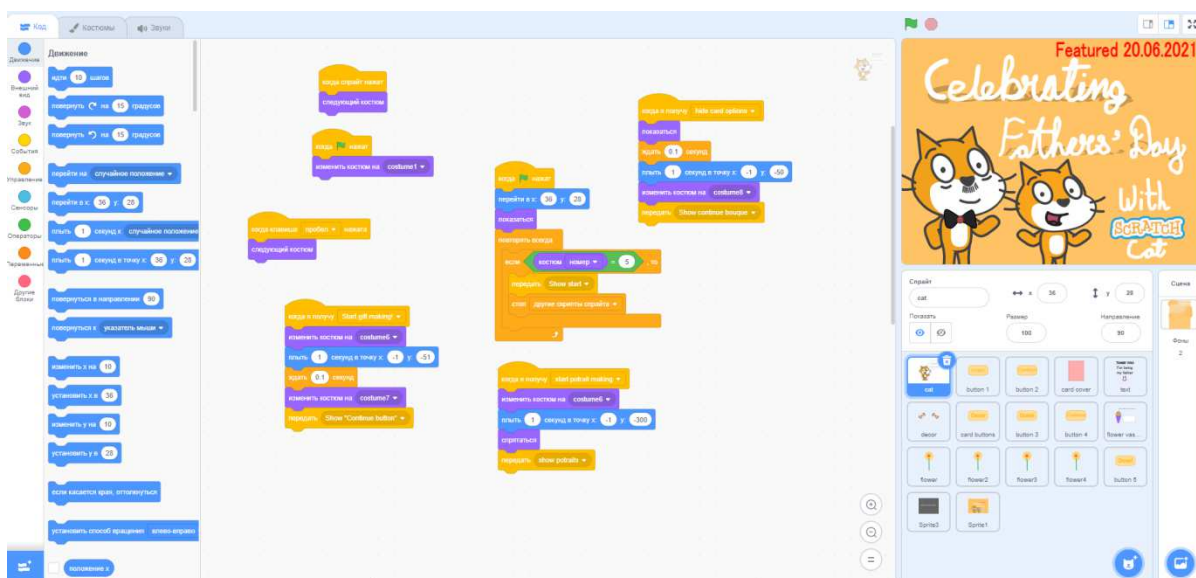


Рисунок 8 – Визуальная событийно-ориентированная среда программирования «Scratch»

Выводы, которые получились после анализа приложений в таблице 1, стоит дополнить тем, что разрабатываемая система должна иметь функцию визуального кодирования, так как исходя из опыта Scratch, это гораздо сильнее упростит кодирование и обучение языкам программирования для тех, кто никогда с этим не сталкивался.

#### 1.4 Описание работы веб-приложения

Разрабатываемое приложение «HTML/CSS» разработано в виде веб-приложения, так как, веб-приложения не требуют скачивания на носители

информации, тем самым освобождая пользователя от лишних файлов. Все необходимые файлы хранятся на сервере. Для доступа к веб-приложению необходим любой браузер и выход в интернет.

Главной особенностью приложения будет визуальное (графическое) кодирование. Суть веб-приложения заключается в том, что пользователю предлагается 5 заданий различной сложности, которые он может выполнять с помощью взаимодействия с элементами, которые содержатся на странице. При каждом взаимодействии с элементом показывается код, который соответствует выполненным действиям. Исходя из этого получается, что пользователь не пишет сам код и саму структуру веб-страницы, она строится за него, но, чтобы выполнить задание, необходимо сделать это по всем правилам HTML и CSS. Перед каждым заданием появляется окно с теоретическим материалом, а также с самим заданием, которое нужно выполнить. Общий вид одного из заданий представлен на рисунке 9.

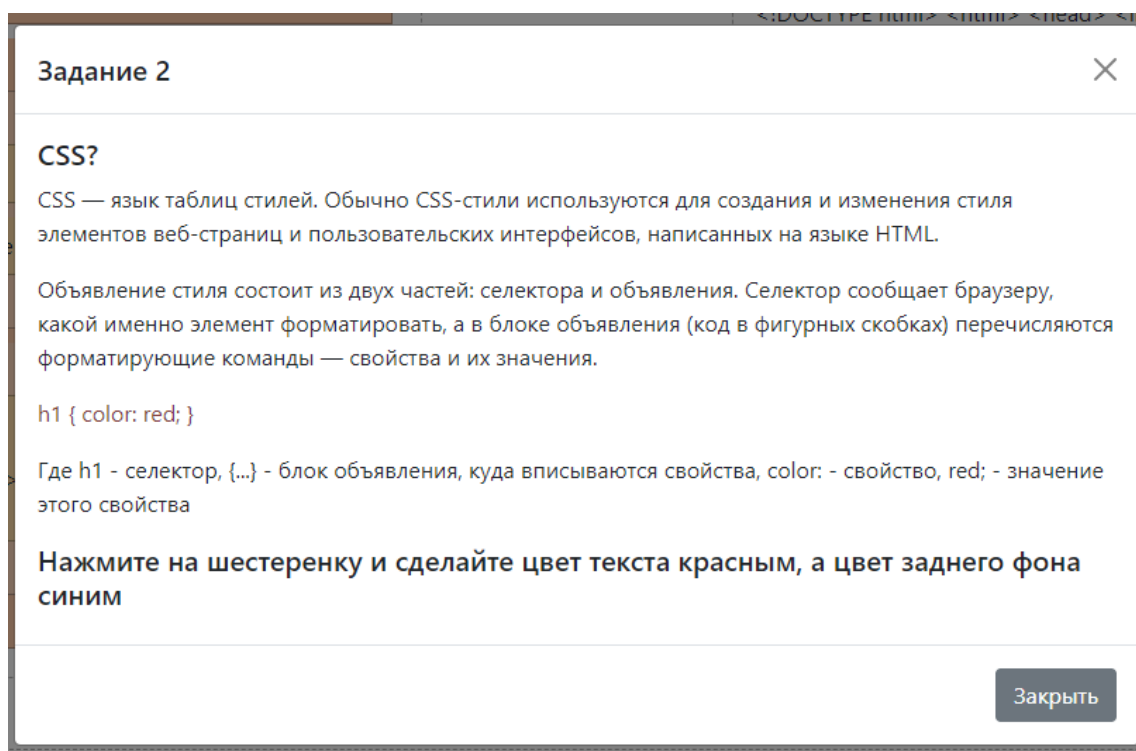


Рисунок 9 – Общий вид задания

## 1.5 Обоснование и выбор средств проектных решений

После анализа веб-приложений, описанных в разделе 1.3, изучив их структуру, были выбраны следующие средства для разработки. Для создания веб-приложения будут использоваться такие языки как, HTML5, CSS3 и JavaScript. HTML и CSS – основные языки разметки и стилизации страницы в сети интернет.

HTML5 – стандартизированный язык разметки веб-страниц в сети интернет. Код HTML интерпретируется браузерами; полученная в результате интерпретации страница отображается на экране монитора компьютера или мобильного устройства [3].

Для описания внешнего вида страниц используется CSS (Cascading Style Sheets). CSS (Cascading Style Sheets) – является формальным языком, используемым для описания, оформления документов, созданных с использованием HTML страниц [7].

Для описания сценария используются такие языки как: Java, JavaScript, Python и другие.

Таблица 2 – Сравнение языков программирования

Свойство	Java	JavaScript	Python
Кроссплатформенность	+	+	+
Сложность (1-5)	3	2	4
Frontend разработка	+	+	-

Так как в веб-приложении не будет баз данных, Python не подходит для разработки клиентской части. Java используется для создания апплетов, которые интегрируются в веб-страницу, программирование на данном языке усложняет разработку. JavaScript используется для программирования клиентской части веб-страницы, описывает сценарии, происходящие на странице, и придает интерактивность сайту.

JavaScript используют большая часть сайтов в сети интернет, он отлично дополняет два вышеописанных языка [2]. Для разработки веб-приложения

используется фреймворк JQuery с библиотекой SortableJS, которая расширяет функции JQueryUISortable, улучшает рабочую механику и упрощает работу.

SortableJS – библиотека, разработанная для организации сортировки внутри списка и между списками [6].

## **1.6 Моделирование бизнес-процесса «Разработка веб-приложения для обучения основам HTML и CSS»**

Для наглядности взаимосвязей и работы разрабатываемой системы была составлена диаграмма IDEF0, показанная на рисунке 10. Для этого были определены следующие данные:

Входы:

– требования – требования к проекту, необходимые для составления технического задания и самой разработки проекта. Полностью регулирует процесс разработки.

Механизмы:

– Разработчик – выполняет все действия, связанные с разработкой проекта.

Используемое ПО – ПО, которое используется для разработки проекта.

Управление:

– ГОСТ Р 51904-2002 – стандарт разработки ПО;

– ISO/IEC 29148:2018 – программная и системная инженерия.

Процессы жизненного цикла. Разработка требований.

Выход:

– Готовое веб-приложение – полностью выполненное веб-приложение, удовлетворяющее требованиям заказчика, которое готово к внедрению.

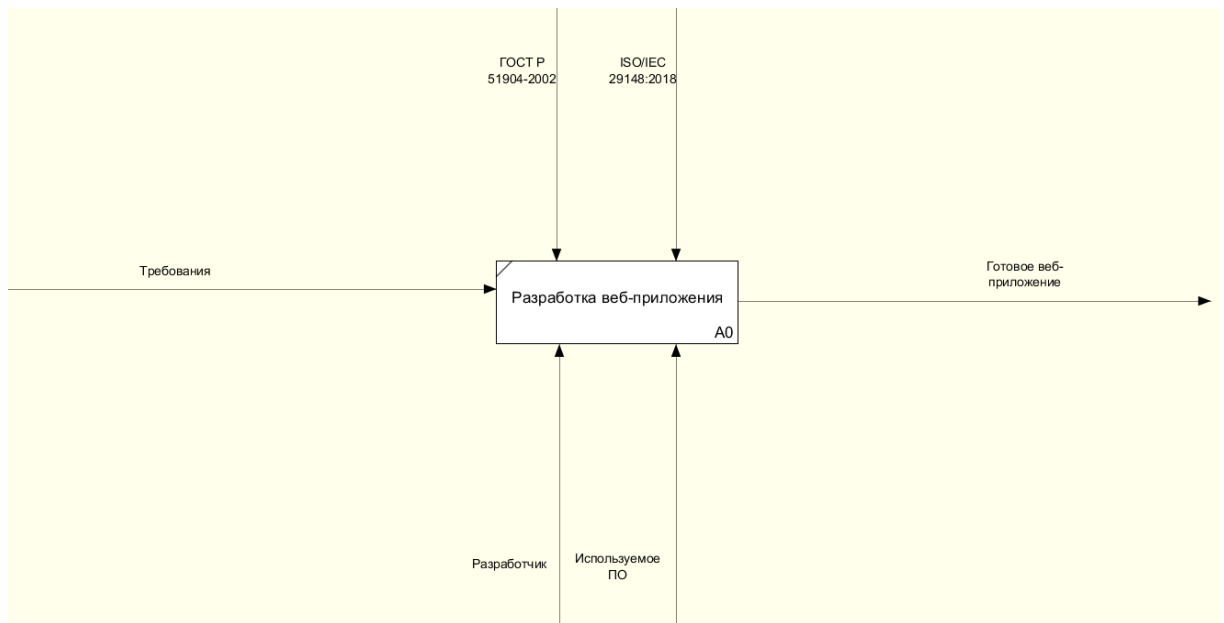


Рисунок 10 – Функциональная модель проекта

Выполним декомпозицию функционального блока, которая показана на рисунке 11.

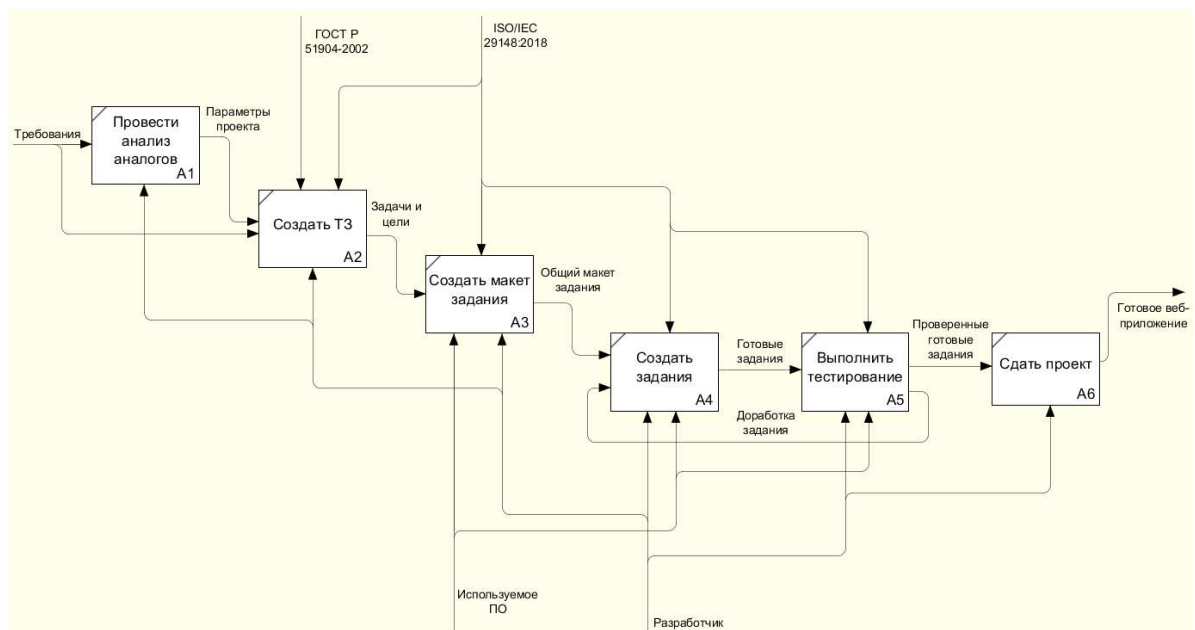


Рисунок 11 – Декомпозиция блока «Разработка веб-приложения»

Далее выполним декомпозицию блока «Создать макет задания», декомпозиция показана на рисунке 12.

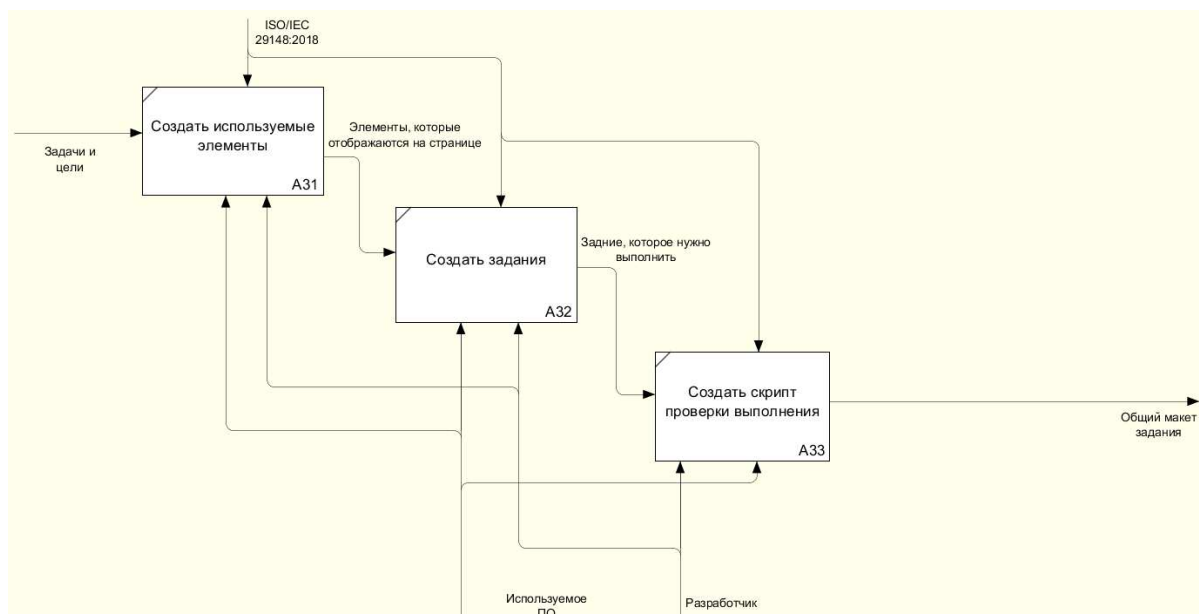


Рисунок 12 – Декомпозиция блока «Создать макет задания»

Из рисунка 12 видно, что нужно сделать для создания одного из заданий в веб-приложении. Разработка начинается с создания элементов, которые будут использоваться на странице, а также для выполнения задания. Далее эти элементы располагаются на странице. Потом создается скрипт, который проверяет выполнение задания и, при правильном выполнении задания, дает доступ к следующему заданию.

### 1.7 Выводы по разделу «Анализ предметной области и выбор средств проектных решений»

В данном разделе было выполнено следующее:

- выполнен анализ предметной области, были выявлены отделы ХТИ – филиала СФУ, которые выполняют функцию привлечения абитуриентов;
- проанализированы подобные системы в других ВУЗах, а также подобные проекты, выявлены их особенности;
- выбраны такие средства разработки как: HTML и CSS, так как это основа веб-страниц в сети интернет. Выбран язык JavaScriptс подключенной библиотекой SortbaleJS, для улучшения взаимодействия с элементами на



странице;

– построена бизнес-модель процесса «Разработка веб-приложения для обучения основам HTML и CSS» в нотации IDEF0, а также выполнена ее декомпозиция.

## 2 Описание процесса разработки веб-приложения

### 2.1 Описание работы пользователя с информационной системой в нотации IDEF3

IDEF3 — методология моделирования и стандарт документирования процессов, происходящих в системе. Метод документирования технологических процессов представляет собой механизм документирования и сбора информации о процессах. С его помощью можно смоделировать технологические процессы, происходящие на предприятии, то есть описать возможные сценарии реализации процессов, в рамках которых происходит последовательное изменение свойств объекта [17].

На рисунке 13 представлен процесс использования сайта, когда пользователь только вошел на главную страницу.

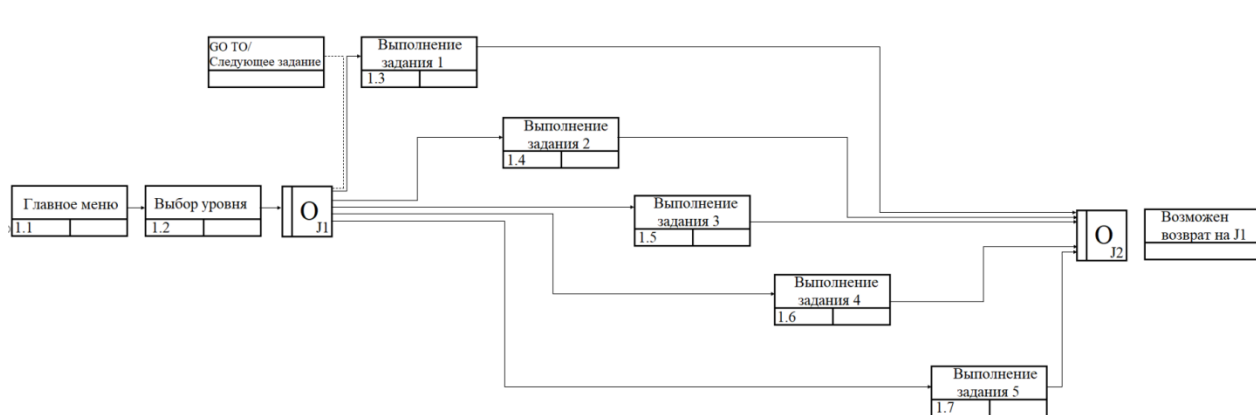


Рисунок 13 – Работа пользователя с веб-приложением

При запуске веб-приложения пользователь заходит в выбор заданий, можно выбрать только одно задание для выполнения. После выполнения задания, пользователь может приступить к выполнению следующего или вернуться на страницу с выбором задания.

## 2.2 Установка и настройка ПО

Так как браузер уже имеется на рабочем компьютере, необходимо только установить программу для программирования на языках HTML, CSS и JS, необходим редактор исходного кода. В данный момент их много, одними из самых популярных является: Visual Studio Code (VSC), Sublime3 и Atom. Сравнение редакторов исходного кода представлено в таблице 3.

Таблица 3 – Сравнение редакторов исходного кода

Свойство	VSC	Sublime 3	Atom
Кроссплатформенность	+	+	+
Дополнительные плагины	+	+	+
Цена	Бесплатный	100\$	Бесплатный

Большинство редакторов исходного кода одинаковы, отличительными особенностями которых являются, дизайн, разработанные плагины и цена.

Среди указанных редакторов Visual Studio Code – разработанный Microsoft, самый распространенный, постоянно обновляемый, большое количество пользователей, в связи с этим большое количество новых плагинов.

После установки Microsoft Visual Studio Code необходимо установить плагины, которые будут помогать в написании кода (рис. 14). Найти их можно в самом приложении, устанавливаются и подключаются автоматически.

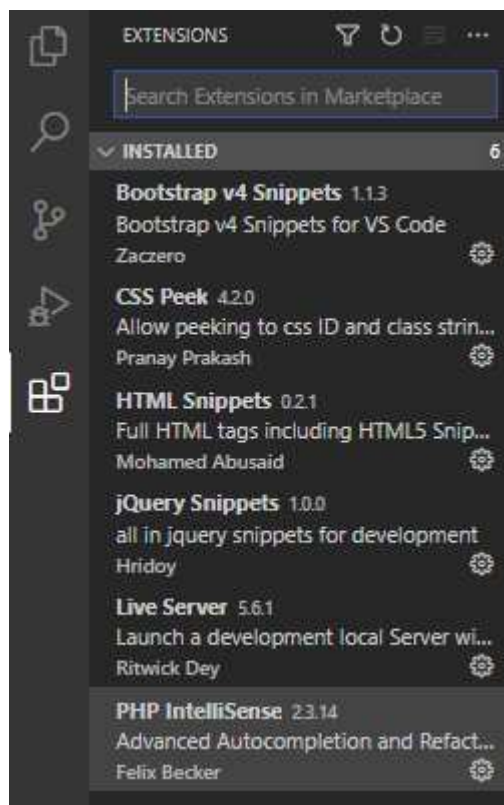


Рисунок 14 – Установленные плагины

Так как визуальная составляющая веб-приложения будет писаться на HTMLи CSS, был установлен плагин BootstrapSnippets, который облегчает работу с библиотекой Bootstrap, предлагая краткое написание исходного кода, который преобразуется в обычный.

Также был установлен плагин JQuerySnippet, создан для такой же функции что и BootstrapSnippets, только для библиотекиJQuery.

LiveServer, для лучшего отображения во время написания кода, при сохранении кода, в автоматическом порядке перезагружает страницу, показывая изменение, избавляет от постоянной перезагрузки страницы вручную.

### 2.3 Создание главного меню и меню выбора задания

Разработка веб-приложения начинается с самого простого, с главного меню, на котором будет отображено название веб-приложения, кнопка,ведущая

на выбор задания, а также небольшое описание веб-приложения, которое подогреет интерес к выполнению заданий.

Для этого была создана стандартная разметка HTML документа, подключен фреймворк Bootstrap для упрощения верстки некоторых элементов. Код главного меню представлен на рисунке 15.

```
<header style="margin-bottom: 350px;">
  <div class="container-fluid">
    <nav class="navbar navbar-light bg-light">
      <a class="navbar-brand" href="#">
        
          HTML/CSS
        </a>
      </nav>
    </div>
  </header>
  <div class="main">
    <a href="zadaniya.html" style="color: black; text-decoration:none;"><h2 style="text-align: center; border: 1px solid black;">Начать выполнять задания</h2></a>
    <h5 style="text-align: center;">Благодаря 5 легким заданиям, Вы сможете получить базовые основы программирования HTML и CSS </h5>
  </div>
```

Рисунок 15 – Код главного меню

Далее было создано меню выбора задания, на котором представлены 5 заданий. Код меню выбора задания представлен на рисунке 16.

```
<div class="main">
  <div class="container-sm text-center">
    <div class="list-group">
      <a href="first.html" class="list-group-item list-group-item-action">
        <b>1 задание.</b> Введение в HTML
      </a>
      <a href="second.html" class="list-group-item list-group-item-action">
        <b>2 задание.</b> Введение в CSS
      </a>
      <a href="third.html" class="list-group-item list-group-item-action">
        <b>3 задание.</b> ID и классы
      </a>
      <a href="four.html" class="list-group-item list-group-item-action">
        <b>4 задание.</b> Работа со стилями
      </a>
      <a href="five.html" class="list-group-item list-group-item-action">
        <b>5 задание.</b> Легкое подведение итогов
      </a>
    </div>
  </div>
</div>
```

Рисунок 16 – Меню выбора задания

Веб-приложение будет состоять из 5 заданий различной сложности:

1. Введение в HTML. В данном задании, пользователю необходимо

правильно расположить элементы HTML-кода. Теоретический материал показывается перед началом выполнения задания, а также его можно вызвать во время выполнения (рис. 17).

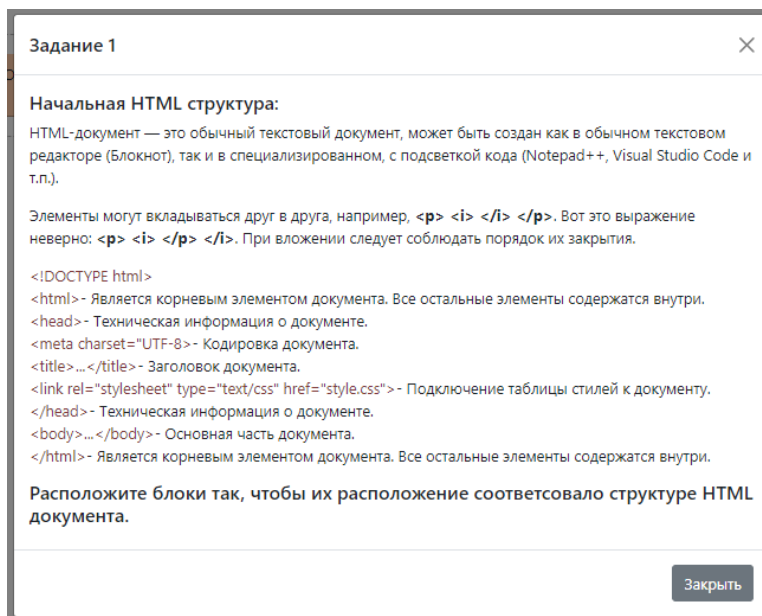


Рисунок 17 – Пример теории перед заданием

2. Введение в CSS. В задании пользователь сталкивается со стилизацией текста (рис. 18). Код, который прописывается при выборе какого-либо стиля, выводится в соответствующее окно (рис. 19).

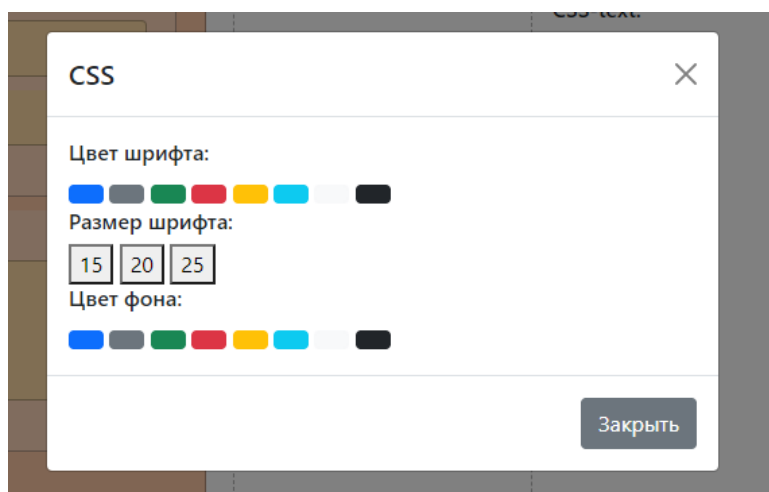


Рисунок 18 – Меню стилизации

```
Text-zone:
HTML-text:
<!DOCTYPE html> <html> <head> <link> <title> Заголовок </title>
</head> <body> <div> <h1> Hello World! </h1> </div> </body>
</html>

CSS-text:
div{
background-color: blue;
}

h1{
color: red;
}
```

Рисунок 19 – Код выбранных стилей

3. ID и классы. Некоторые элементы представлены без своих классов и ID, у них описана стилизация, но из-за этого она не работает. Пользователь должен ввести ID и классы элементам.

4. Работа с блоками. Каждая веб-страница состоит из блоков, которые необходимо правильно стилизовать, во-первых, для правильного отображения на странице, во-вторых, для красоты.

5. Итоговое задание. Это задание объединяет то, что пользователь изучил в прошлых заданиях. Пользователь столкнется со стилизацией текста и блоков, расположения на странице. Задание представлено на рисунке 20.

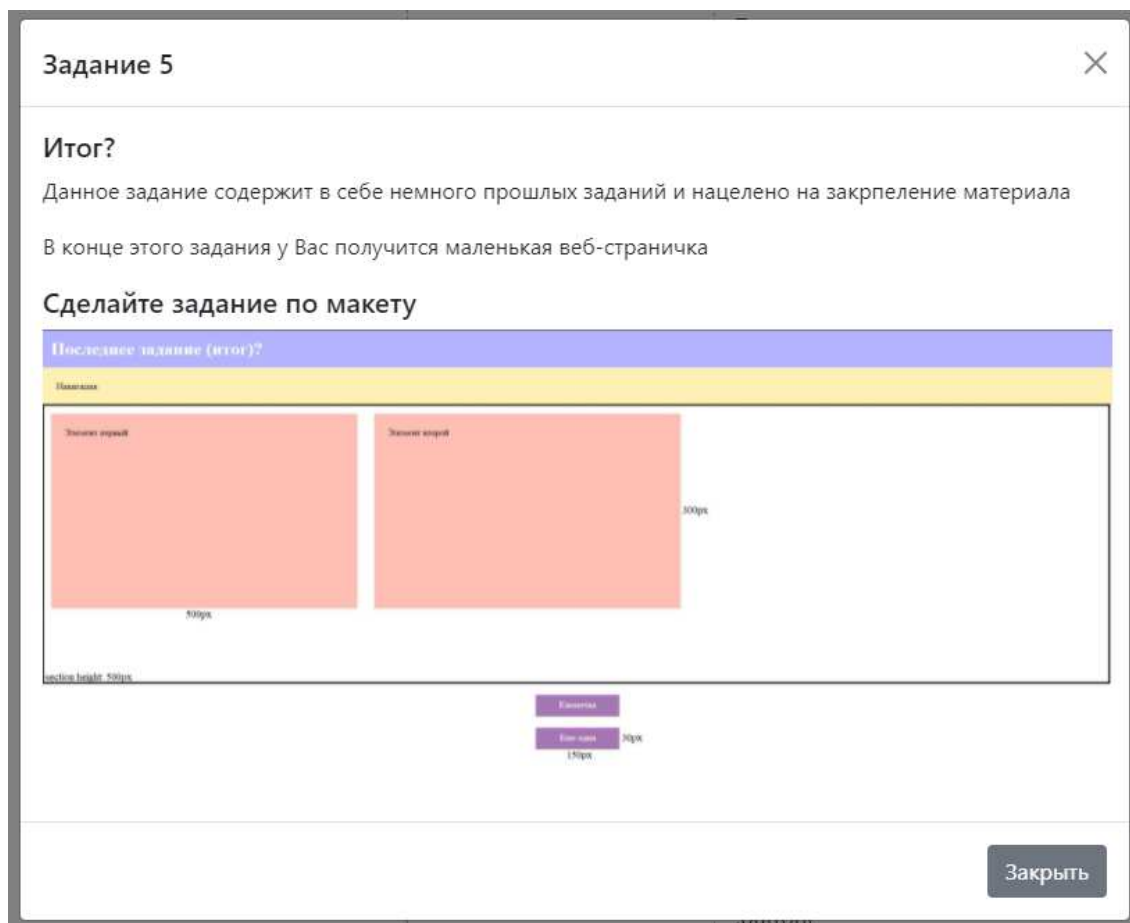


Рисунок 20 – Задание 5

## 2.4 Создание прототипа заданий

Для всех заданий была сформирована одна структура HTML страницы, которая содержит в себе два рабочих блока, с которыми происходит взаимодействие пользователя: «HTML» и «Элементы». Блок, в котором дублируется HTML и CSS код, который формируется в результате манипуляций с рабочими блоками. А также, кнопка, которая проверяет результат выполнения задания; при правильности выполнения появляется кнопка, которая ведет к следующему заданию. Результат создания макета задания представлен на рисунке 21.



Рисунок 21 – Макет задания

Далее добавляются блоки в соответствии с заданиями. С этими блоками в последствии пользователь будет взаимодействовать. Результат добавления блоков в первом задании показан на рисунке 22.



Рисунок 22 – Рабочее поле с добавленными блоками

## 2.5 Реализация функции перемещения элементов

Для упрощения взаимодействия пользователя с веб-приложением, в результате анализа подобных систем (пункт 1.3), было принято решение реализовать функцию перемещения рабочих элементов. Данное решение освободит пользователя от монотонного ввода текста для размещения элементов, добавит интерактивности, что ускорит время выполнения заданий, тем самым больше абитуриентов смогут опробовать данное веб-приложение.



Вышеперечисленное может привести к повышению интереса поступить в ХТИ – филиал СФУ.

В JQueryUI имеется функция Sortable, но на нее содержится очень мало документации, из-за чего повышается сложность разработки веб-приложения. Поэтому для реализации эффекта визуального кодирования была выбрана сторонняя библиотека SortableJS для JQueryUI, она имеет улучшенное взаимодействие перемещаемых и сортируемых элементов страницы, а также обширную документацию.

Особенность работы данной библиотеки состоит в том, что каждому блоку, в котором будет работать функция «сортировки и перемещения» элементов, присваивается класс sort.

Далее необходимо написать скрипт, который позволит перетаскивать блоки, а также их “сортировать”. Каждому родительскому блоку, в котором должны происходить перемещение и сортировка, прописываем класс sort. Далее в JS создаем переменную sort, в которую считываются все элементы с классом sort. Далее создается цикл, который каждой переменной sort дает свой индекс. Код, выполняющий данные действия, показан на рисунке 23.

```
var sort = [].slice.call(document.querySelectorAll('.sort'));
for (var i = 0; i < sort.length; i++) {
  new Sortable(sort[i], {
    group: 'sort',
    animation: 150,
    swapThreshold: 0.2,
    emptyInsertThreshold: 20,
    fallbackOnBody: true,
    onEnd: function (Event) {
      var text = $("#drop").text();
      $("#html-text").text(text);
    }
  });
};
```

Рисунок 23 – Код для перемещения и сортировки элементов

Далее создается функция sortable для всех элементов из цикла. В этой функции происходит группировка всех элементов в одну группу, чтобы они

могли сортировать в себя, задается скорость анимации и чувствительность на перемещение и вложение, а также настройка так называемого призрачного блока для того, чтобы показывать, куда вкладывается блок. А также была настроена функция onEnd, для отображения текста, который получается от выполнения действий с блоками. Результат работы скрипта представлен на рисунке 24.

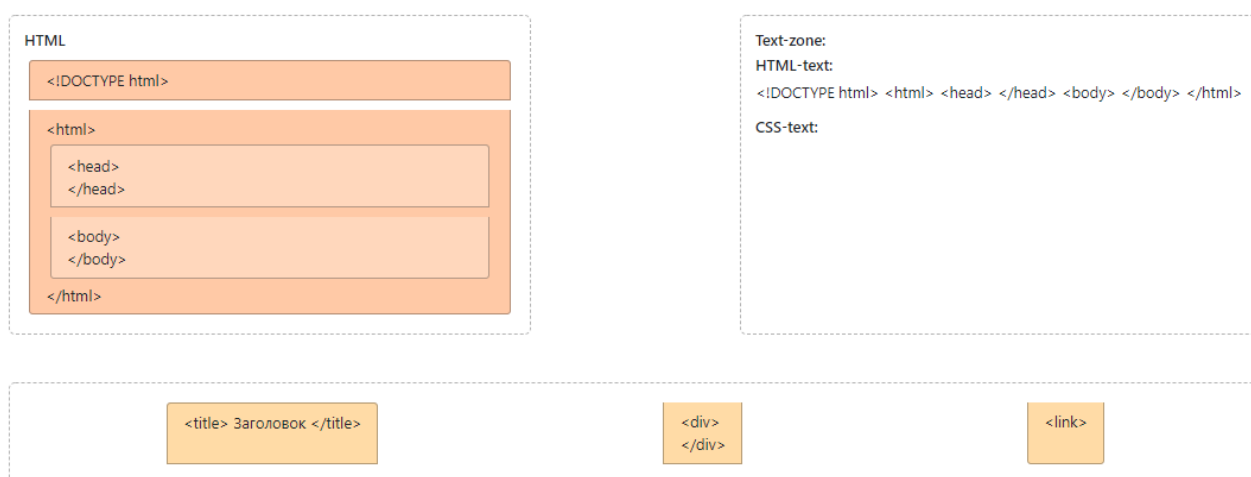


Рисунок 24– Результат работы скрипта

После выполнения скрипта пользователь видит HTML код в виде текста, который он сделал обычным перетаскиванием блоков. Чтобы увидеть результат построения кода блоками, пользователю необходимо нажать кнопку «Проверить»; после нажатия кнопки создается фрейм, в котором показан результат, как будто пользователь вручную написал код. Код для создания фрейма показан на рисунке 25.

```

$("#go").click(function(){
var text2 = $("#html-text").text();
var text3 = $("#css-text").text();
function createFrame() {
    var frame = document.createElement("iframe");
    document.body.appendChild(frame);
    frame.height = "500px";
    frame.width = "100%";
    frame.contentDocument.head.innerHTML = '<style>' + text3 + '</style>';
    frame.contentDocument.body.innerHTML = text2;
}
createFrame();

```

Рисунок 25 – Код создания фрейма

Также эта кнопка сравнивает код, полученный в результате выполнения задания, и правильный код. Исходя из этого, если задание выполнено правильно, появляется кнопка, ведущая к следующему заданию. Код проверки выполнения первого задания представлен на рисунке 26.

```

$("#check").click(function(){
    var rightText = "<!DOCTYPE html> <html> <head>
    <link> <title> Заголовок </title> </head> <body>
    <div> </div> </body> </html>".replace(/\s+/g, '').
    toLowerCase();
    var value = $("#html-text").text().replace(/\s+/g,
    '').toLowerCase();
    var result = document.getElementById('result');
    if (value === rightText)
    {
        result.innerHTML = "Задание выполнено верно.";
        $("#next").slideToggle(300)
    }
    else{
        result.innerHTML = "Что-то не так!";
    }
})

```

Рисунок 26– Код проверки выполнения первого задания

В конечном итоге выполненное первое задание представлено на рисунке 27.

HTML

```
<!DOCTYPE html>

<html>
  <head>
    <link>
    <title> Заголовок </title>
  </head>
  <body>
    <div>
    </div>
  </body>
</html>
```

Text-zone:

HTML-text:

```
<!DOCTYPE html> <html> <head> <link> <title> Заголовок </title>
</head> <body> <div> </div> </body> </html>
```

CSS-text:

Проверить Далее

Результат:  
Задание выполнено верно.

Рисунок 27 – Выполненное первое задание

## 2.6 Реализация стилизации элементов

Для реализации стилизации элементов каждому элементу, для которого выполняется стилизация добавлена кнопка, открывающая меню заготовленных стилей, что также упрощает взаимодействие пользователя с системой, избавляя от ввода текста. Результат добавления кнопки показан на рисунке 28.

```
<!DOCTYPE html>

<html>
  <head>
    <link>
    <title> Заголовок </title>
  </head>
  <body>
    <div> ⚙️
    <h1> Hello World! </h1>
    </div>
  </body>
</html>
```

Рисунок 28 – Кнопка стилизации

Нажатие на кнопку показывает меню со стилями, которые можно применить для данного блока; результат на рисунке 29.

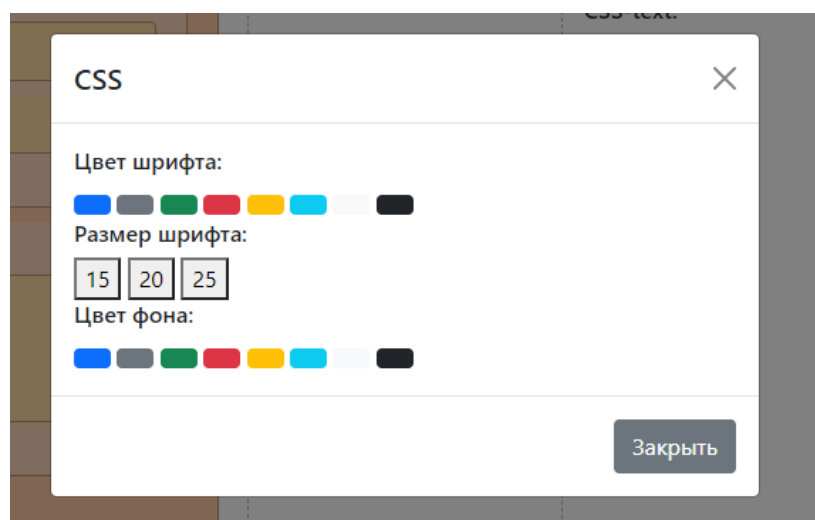


Рисунок 29 – Меню стилизации

При нажатии на каждую кнопку она выводит соответствующий CSS код, который, при создании фрейма будет присвоен своему элементу; результат на рисунке 30.

```
Text-zone:  
HTML-text:  
<!DOCTYPE html> <html> <head> <link> <title> Заголовок </title>  
</head> <body> <div> <h1> Hello World! </h1> </div> </body>  
</html>  
  
CSS-text:  
div{  
background-color: blue;  
}  
  
h1{  
color: red;  
}
```

Рисунок 30 – Код выбранных стилей

Данный код добавляется во фрейм, стилизуя расположенные в нем элементы. Результат выполнения второго задания с этим скриптом показан на рисунке31.

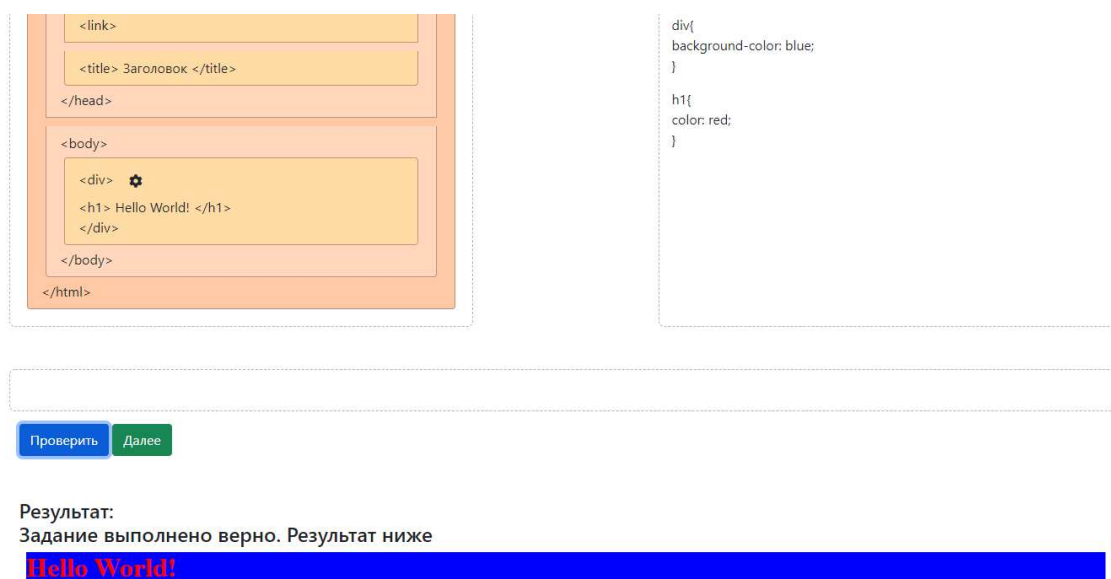


Рисунок 31 – Результат выполнения второго задания

## 2.7 Разработка остальных заданий

Так как сценарий заданий был описан в пункте 2.3, первые два задания с основными функциями разработаны, необходимо разработать следующие уровни, с доработкой существующих скриптов.

Согласно сценарию, в третьем задании пользователь работает с классами и ID, их стилизация описана, но она не работает. Для реализации этого задания был добавлен поле ввода, в которые пользователь должен ввести названия классов и ID (рис.32).

```
<div>
<h1>
Заголовок первого уровня
Введите zag
</h1>
<p>
Абзац 1
Введите a1
</p>
<p>
Абзац 2
Введите a2
</p>
</div>
```

Рисунок 32 – Добавленные поля

После ввода названий текст автоматически передается в текстовую составляющую HTMLкода, для дальнейшего взаимодействия с ним. Результат показан на рисунке 33.

### HTML-text:

```
<!DOCTYPE html> <html> <head> <link> <title> Заголовок </title>
</head> <body> <div> <h1 id="zag"> Заголовок первого уровня
</h1> <p class="a1"> Абзац 1 </p> <p class="a2"> Абзац 2 </p>
</div> </body> </html>
```

Рисунок 33 – Результат прописывания классов и ID

Для реализации четвертого задания, в котором пользователь должен поработать с блочными элементами, а именно расположить их правильно на странице и стилизовать (рис. 34).

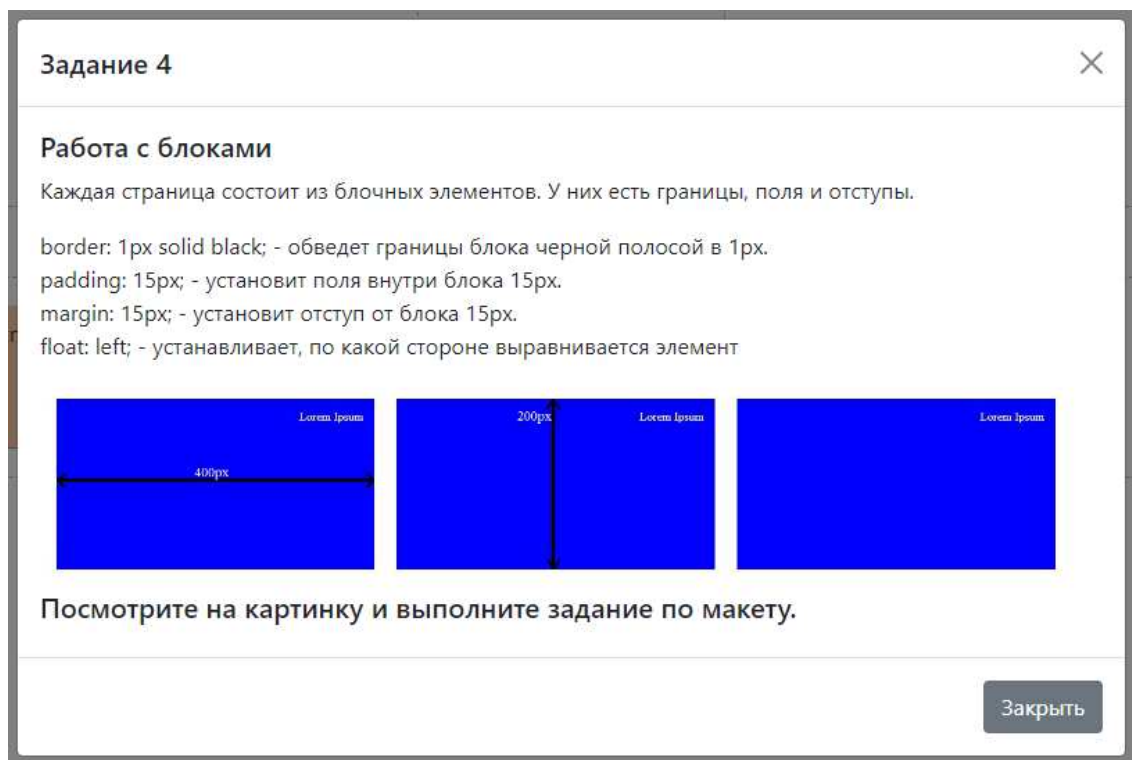


Рисунок 34 – Задание 4

Для выполнения задания пользователь должен расположить блоки в соответствии с разметкой HTML, которая была описана в первом задании. А также, исходя из макета, представленного в задании, стилизовать блоки. Добавить им цвет фона, ширину и высоту блока, внутренние и внешние отступы, а также расположить текст в блоке согласно заданию (рис. 35).



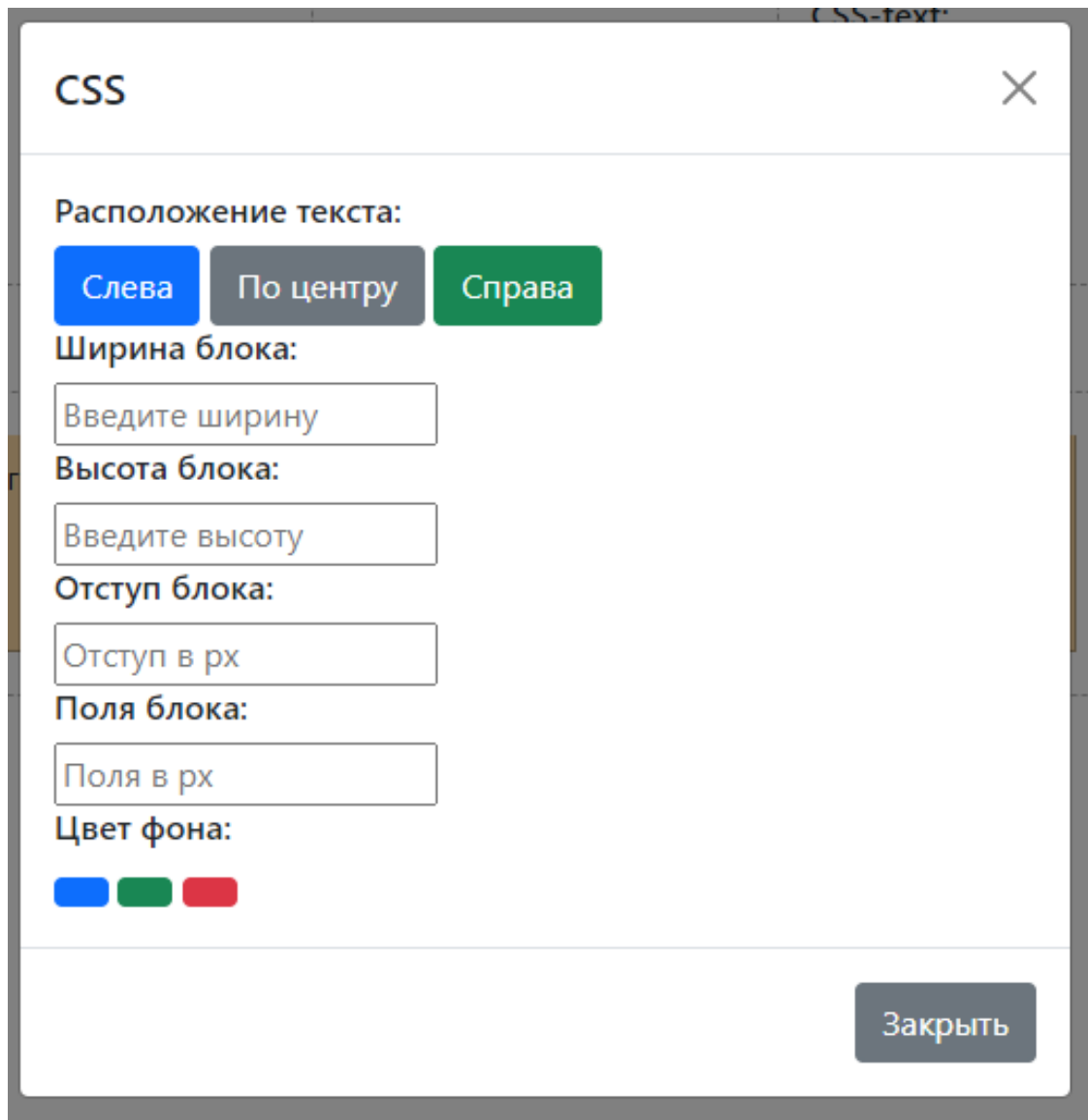


Рисунок 35 – Меню стилизации блока

При вводе с клавиатуры в поля ввода и выборе расположения текста, а также цвета фона, советуемый код прописывается в свое поле. Результат показан на рисунке 36.

```
CSS-text:
div{
color: white;
float: left;
width: 400px;
height: 200px;
margin: 15px;
padding: 15px;
text-align: right;
background-color: blue;
}
```

Рисунок 36 – Результат стилизации

После расположения всех блоков и их стилизации пользователь нажимает на кнопку проверки, которая запускает скрипт проверки полученного кода, результат показан на рисунке 37.

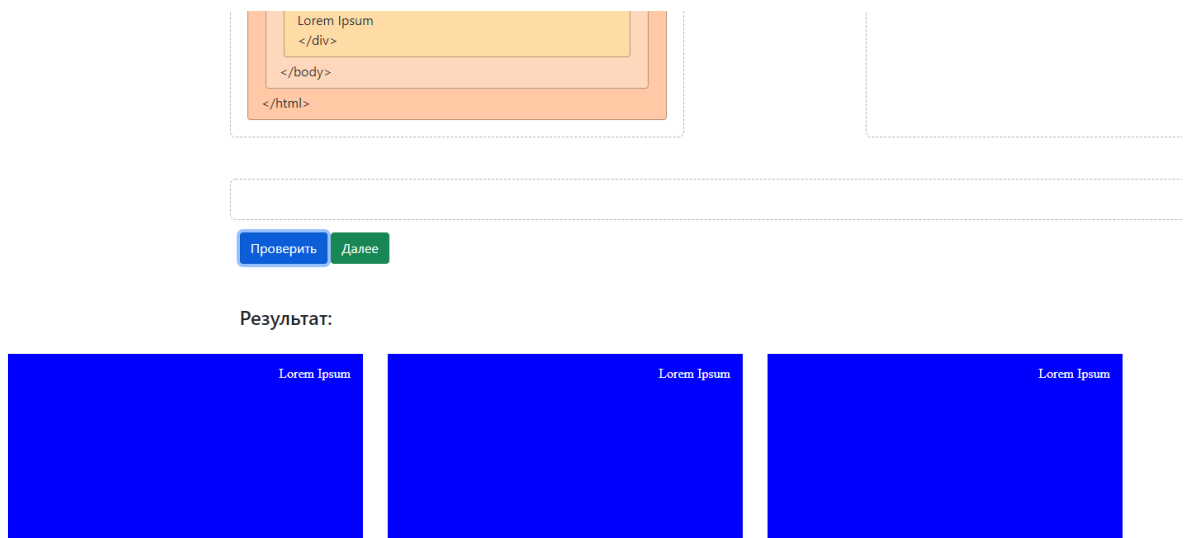


Рисунок 37 – Результат выполнения задания 4

Последнее, пятое задание, состоит из объединения всего того, что пользователь прошел ранее. В этом задании пользователю необходимо разработать маленькую веб-страницу. Для этого ему потребуется правильно расположить элементы, согласно структуре HTML страницы, пример показан на рисунке 24, а также стилизовать необходимые элементы (рис. 38).

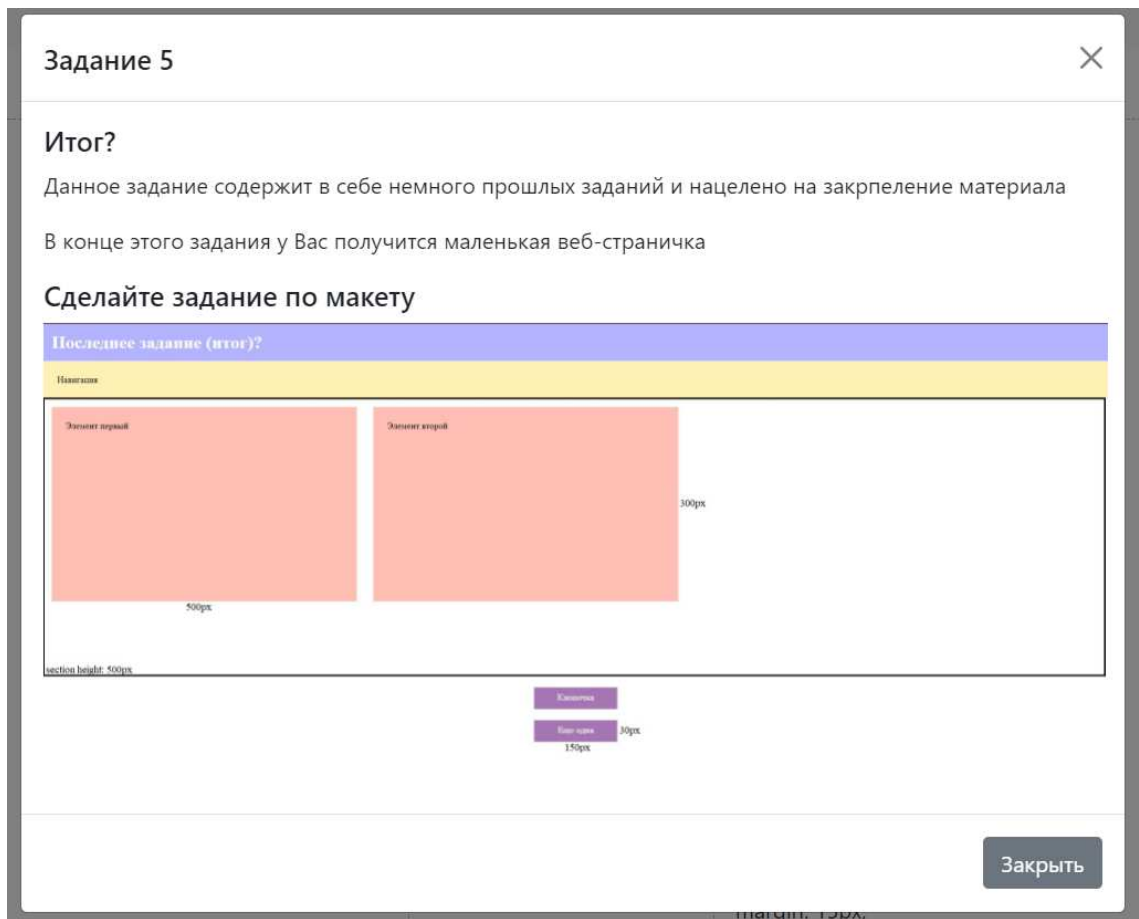


Рисунок 38 – Макет для выполнения задания 5

У пользователя есть все необходимые ему элементы, а также поле для CSS текста прописана некоторая стилизация, которая будет дополняться от действий пользователя. Результат показан на рисунках 39 – 40.



Рисунок 39 – Элементы для 5 задания

```
CSS-text:
body{
padding: 0px;
margin: 0px;
}
header{
background-color: rgb(179, 179, 255);
padding: 15px;
}
h1{
margin: 0px;
}
nav{
background-color: rgb(255, 240, 179);
padding: 25px;
}
section{
}
p{
float: left;
margin: 15px;
background-color: rgb(255, 190, 179);
padding: 25px;
}
.button{
text-decoration: none;
padding-top: 10px;
display: block;
margin: 20px auto;
background-color: #a675b3;
}
```

Рисунок 40 – Готовая стилизация

После расположения блоков пользователь может посмотреть, что получается в промежуточный момент, тем самым понять, чего не хватает для правильной реализации.

Также пользователь должен настроить каждый необходимый блок, они помечены кнопкой с шестеренкой, при нажатии на которую, каждому блоку доступна своя стилизация. Например, для того чтобы узнать необходимые размеры р-элементов, нужно правильно изучить макет перед заданием. На нем можно увидеть заданные показатели ширины и высоты блока. Далее эти

значения необходимо ввести в меню данного блока, для его стилизации. Меню указано на рисунке 41.

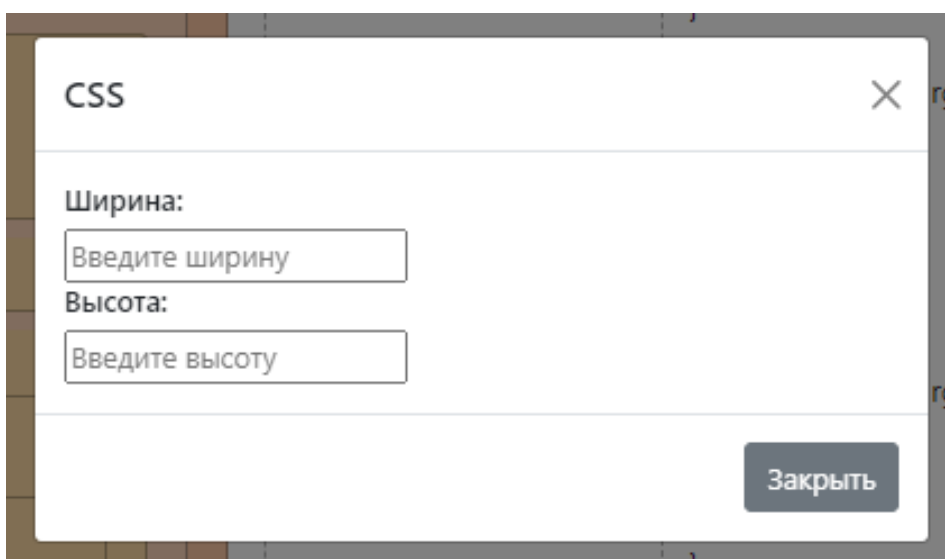
A screenshot of a web application window titled "CSS" with a close button (X) in the top right corner. The window contains two input fields for styling. The first is labeled "Ширина:" (Width) and has a text input field with the placeholder "Введите ширину" (Enter width). The second is labeled "Высота:" (Height) and has a text input field with the placeholder "Введите высоту" (Enter height). In the bottom right corner of the window, there is a dark grey button labeled "Закреть" (Close).

Рисунок 41 – Меню стилизации р-элемента

После выполнения всех действий, при нажатии на кнопку «проверка», пользователь увидит результат, и при пройденной проверке на правильность выполнения появится кнопка для выхода из задания (рис.42).

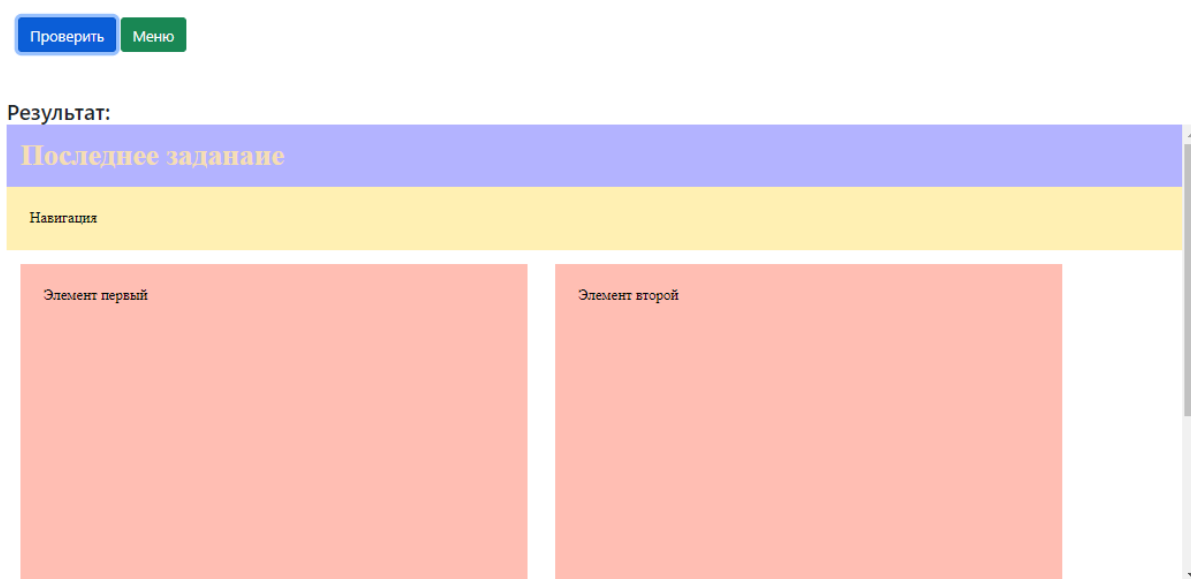


Рисунок 42 – Результат выполнения 5 задания

## 2.8 Выводы по проектному разделу

Так как задания имеют идентичную структуру, они не были расписаны. Было разработано веб-приложение, которое выполняет поставленные задачи и цель. Созданы пять заданий, которые помогут в изучении основ HTML и CSS, разработан функционал, который помогает в использовании веб-приложения, а именно:

- реализована функция перемещения и сортировки элементов страницы, с помощью библиотеки SotrableJS для JavaScript;
- реализована функция проверки выполнения задания;
- реализован фрейм, в котором можно увидеть результат выполненного задания;
- реализована функция стилизации элементов.

## 3 Расчет затрат и оценка эффективности реализации проекта «Разработка Веб-приложения для обучения основам HTML и CSS»

В экономическом разделе рассчитываются затраты на проектирование системы, рассчитываются риски разработки и внедрения системы.

Для расчета затрат выбрана методика TCO.

Затраты реализации можно рассчитать по методике TCO, которая рассчитывается по формуле 1.

$$TCO = DE + IC_1 + IC_2, \quad (1)$$

где DE – прямые затраты;

IC<sub>1</sub> – косвенные затраты первой группы;

IC<sub>2</sub> – косвенные затраты второй группы.

### 3.1 Капитальные затраты

Капитальные затраты вычисляются по следующей формуле 2.

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оэ}}, \quad (2)$$

где  $K_{\text{пр}}$  – затраты на проектирование ИС;

$K_{\text{тс}}$  – затраты на технические средства управления;

$K_{\text{лс}}$  – затраты на создание линий связи локальных сетей;

$K_{\text{по}}$  – затраты на программные средства;

$K_{\text{ио}}$  – затраты на формирование информационной базы,

$K_{\text{об}}$  – затраты на обучение персонала;

$K_{\text{оэ}}$  – затраты на опытную эксплуатацию.

Расчет времени для реализации проекта:

1. анализ предметной области – 2 день.
2. разработка прототипа – 15 дней.
3. тестирование – 5 дня.
4. доработка – 7 дней.
5. введение в эксплуатацию – 1 дня.

Затраты на проектирование ИС рассчитываются по формуле 3.

$$K_{\text{пр}} = K_{\text{зп}} + K_{\text{ипс}} + K_{\text{свт}} + K_{\text{проч}}, \quad (3)$$

где  $K_{\text{зп}}$  – затраты на заработную плату проектировщиков;

$K_{\text{ипс}}$  – затраты на инструментальные программные средства;

$K_{\text{свт}}$  – затраты на средства вычислительной техники;

$K_{\text{проч}}$  – прочие затраты на проектирование.

Средний оклад программиста в регионе равен 20000 рублей в месяц, без учета региональной надбавки, северного коэффициента и НДФЛ.

Расчет зарплаты программиста с учетом региональной надбавки (30%) и северного коэффициента (30%) представлен в таблице 4. Так же следует учитывать перечисления во внебюджетные фонды 30,2%.

Таблица 4 – Расчет зарплаты программиста

Состав затрат	Планируемая сумма, руб.
Оклад	20000
Северный коэффициент	6000
Районный коэффициент	6000
НДФЛ	4160
Отчисления во внебюджетные фонды	964
Итого	41664

$K_{зп}=41664$ рубля.

Затраты на инструментальные программные средства представлены в таблице 5.

Таблица 5 – Необходимое ПО

Наименование ПО	Кол-во	Срок службы	Цена, руб
Windows 10 Home	1	5 лет	14500
Visual Studio Code	1	Бессрочно	Бесплатно
Итого			10 150

Windows 10 Home выпустили в 2015 году, обновляется бесплатно, если сравнить с Windows 7, то срок поддержки составляет 5 лет.

Поэтому амортизация будет составлять в год  $14500/5 = 2900$  рублей. А во время использования  $2900/12 = 241,66$  рублей.

$K_{ипс} = 241,66$  рублей.

Так же программисту для реализации проекта необходим компьютер, его составляющие предоставлены в таблице 6.

Таблица 6 – Компьютер программиста



Наименование	Количество единиц	Срок эксплуатации	Цена, руб
Процессор Ryzen 3 2200G	1	7 лет	9 490
Видеокарта ASUS RX550	1	7 лет	15 100
Оперативная память HyperX DDR4 2x8Gb	1	10 лет	8990
Жесткий диск WD Blue 1ТБ	1	5 лет	2990
Материнская плата MSI B450M-A PRO MAX	1	7 лет	5110
Блок питания FSP PNR 500W	1	10 лет	3120
Монитор HP 22у 22"	1	7 лет	9490
Мышь Logitech G102	1	7 лет	2290
Клавиатура Logitech Keyboard K380	1	7 лет	3490
Итоговая стоимость			60070

Общая стоимость компьютера составляет 60070 рублей, а средний срок эксплуатации составляет 7 лет, далее рассчитаем амортизацию оборудования.

Амортизация рассчитывается по формуле 4.

$$A_{\text{Год}} = C_{\text{б}} * N_{\text{ам}}, \quad (4)$$

где  $A_{\text{Год}}$  – амортизация за год использования;

$C_{\text{б}}$  – балансовая стоимость;

$N_{\text{ам}}$  – норма амортизации.

Проектная амортизация рассчитывается по формуле 5.

$$A_{\text{пр}} = \frac{A_{\text{год}}}{K_{\text{рдг}}}, \quad (5)$$

где  $A_{\text{год}}$  – амортизация за год использования;

$A_{\text{пр}}$  – проектная амортизация;

$K_{\text{рдг}}$  – количество рабочих дней в году;

$K_{\text{дэ}}$  – количество дней эксплуатации.

Норма амортизации рассчитывается по формуле 6.

$$N_{\text{ам}} = \frac{100\%}{T_{\text{эк}}}, \quad (6)$$

где  $N_{ам}$  – норма амортизации;

$T_{эк}$  – срок эксплуатации компьютера.

$$N_{ам} = \frac{100\%}{T_{эк}}, \quad (7)$$

где  $N_{ам}$  – норма амортизации;

$T_{эк}$  – срок эксплуатации компьютера.

$1/7 * 100\% = 14,3\%$  процент амортизации за год.

$60070 * 0,143 = 8590$  рубля амортизация за год.

$8590/12 = 715,83$  рубля амортизация за 30 дней.

$K_{свт} = 715,83$  рубля.

В ходе работы возможен риск, что какое-либо оборудование может выйти из строя, нужно будет обслуживать оборудование. Прочие затраты составляют 3% от расходов на ПО, заработную плату программиста и вычислительную технику.

$K_{проч} = (41664 + 242 + 716) * 0.03 = 1279$  руб.

После всех вычислений составляет  $K_{пр} = 41664 + 242 + 716 + 1279 = 43898$  рубля.

$K_{тс}$  – затраты на технические средства управления, равны 0, так как ХТИ – филиал СФУ имеет все необходимые технические средства, поэтому нет необходимости закупать оборудование; для управления веб-приложением для обучения основам HTML и CSS нужен компьютер, который способен открыть браузер, нагрузка на аппаратную часть будет минимальной;

$K_{лс}$  – затраты на создание линий связи локальных сетей отсутствуют, так как у ХТИ – филиал СФУ уже имеется своя локальная линия связи и выход в сеть интернет;

$K_{по}$  – затраты на программные средства, составляют 0 рублей, так как для внедрения нужен только любой браузер. Браузеры распространяются бесплатно;

$K_{\text{ио}}$  – затраты на формирование информационной базы, равны 0, потому что нет никаких информационных баз.

$K_{\text{об}}$  – затраты на обучение персонала, равны 0, так как веб-приложению не нужны обслуживающие пользователи.

$K_{\text{оз}}$  – затраты на опытную эксплуатацию. Для проверки системы потребуется хостинг, чтобы проверить её в готовом режиме (рис. 43).

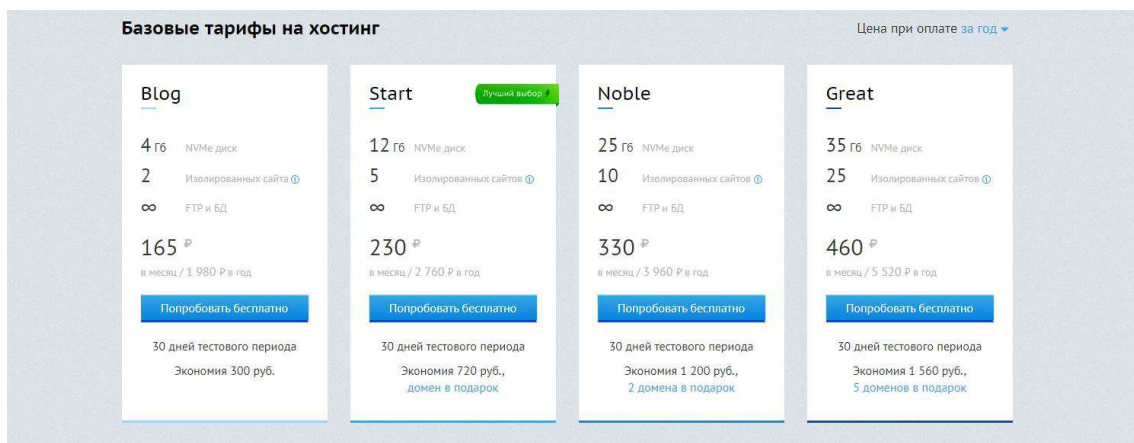


Рисунок 43 – Цены хостинг-сервера

Так как веб-приложению нет необходимости в большом объеме памяти и количестве сайтов, самый подходящий тариф «Blog» составляет 165 рублей.

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оз}}, \quad (8)$$

где  $K_{\text{пр}}$  – затраты на проектирование ИС;

$K_{\text{тс}}$  – затраты на технические средства управления;

$K_{\text{лс}}$  – затраты на создание линий связи локальных сетей;

$K_{\text{по}}$  – затраты на программные средства;

$K_{\text{ио}}$  – затраты на формирование информационной базы,

$K_{\text{об}}$  – затраты на обучение персонала;

$K_{\text{оз}}$  – затраты на опытную эксплуатацию

$$K = 43897,96 + 165 = 44063 \text{ рублей.}$$

Список капитальных затрат указан в таблице 7.

Таблица 7 – Список капитальных затрат

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на разработку ИС	Затраты на заработную плату разработчика	41664
	Затраты на программные средства	242
	Затраты на средства вычислительной техники	716
	Прочие затраты на разработку	1279
Затраты на опытную эксплуатацию		165
Затраты на обучение		0
Итого		44063

Структура капитальных затрат показана на рисунке 44.

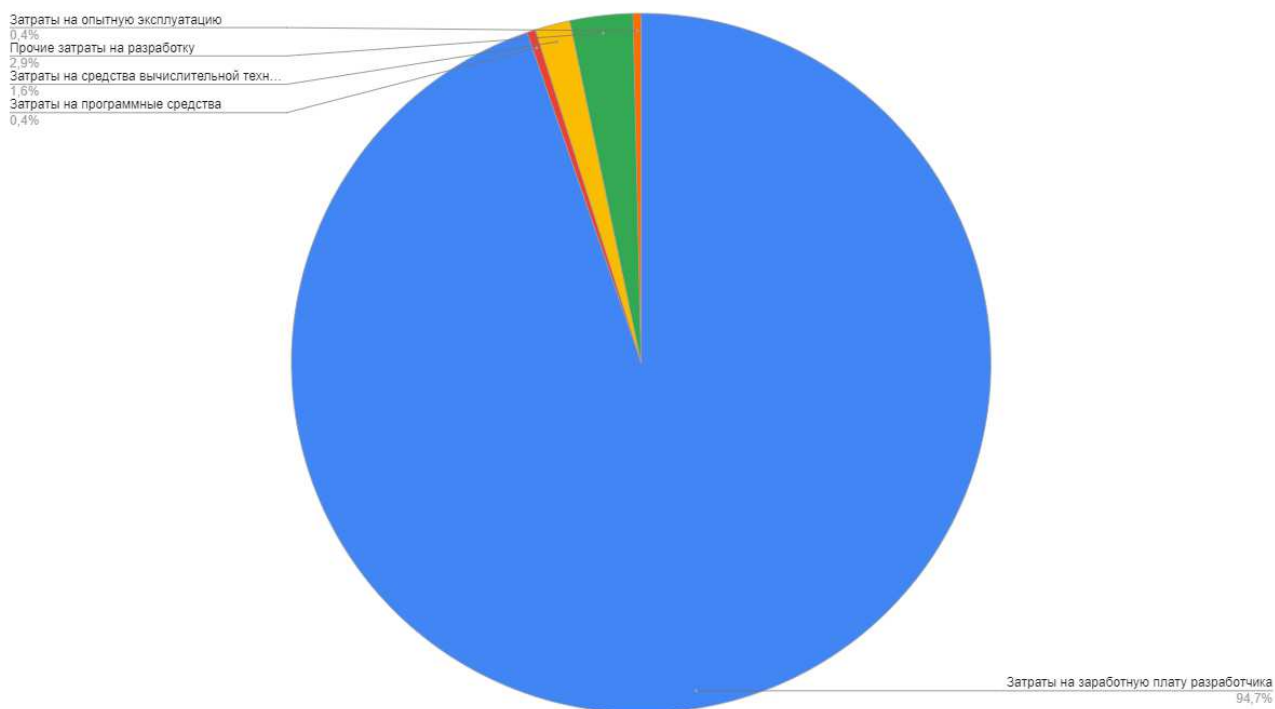


Рисунок 44 – Структура капитальных затрат

### 3.2 Расчет эксплуатационных затрат

Расчет эксплуатационных затрат производится по формуле 9.

$$C = C_{\text{зп}} + C_{\text{ао}} + C_{\text{то}} + C_{\text{лс}} + C_{\text{ни}} + C_{\text{проч}}, \quad (9)$$

где  $C_{\text{зп}}$  – зарплата персонала, работающего с информационной системой:

$C_{\text{ао}}$  – амортизационные отчисления;

$C_{\text{то}}$  – затрата на техническое обслуживание;

$C_{\text{лс}}$  – затраты на использование глобальных сетей;

$C_{\text{ни}}$  – затраты на носители информации;

$C_{\text{проч}}$  – прочие затраты.

#### **Затраты на заработную плату персонала.**

Затраты на заработную плату персонала равны 0, так как персонал не требуется.

#### **Затраты на техническое обслуживание.**

Затраты на техническое обслуживание и ремонт техники непостоянны и составляют 1000 рублей.

$$C_{\text{то}} = 1000 \text{ рублей.}$$

#### **Затраты на использование глобальных сетей.**

Необходимо подсчитать сумму веб-хостинга за год:

$$C_{\text{лс}} = 165 * 12 = 1980 \text{ рублей.}$$

#### **Затраты на носители информации.**

Данные системы будут храниться на хостинге, на котором есть носитель информации, так как хостинг предоставляет данную услугу. Дополнительных носителей информации не требуется.

$$C_{\text{ни}} = 0 \text{ рублей.}$$

#### **Прочие затраты.**

Прочие затраты равны 3% от суммы всех затрат.

$$C_{\text{проч}} = (0 + 0 + 1000 + 1980) * 0,03 = 89,4 \text{ рублей.}$$

$$C = 0 + 0 + 1000 + 1980 + 89,4 = 3069,4 \text{ руб.}$$

Эксплуатационные затраты указаны в таблице 8.

Таблица 8 – Эксплуатационные затраты

Состав затрат	Планируемая сумма, руб.
Затраты на заработную плату персонала	0
Затраты на амортизацию	0
Затраты на техническое обслуживание	1000
Затраты на использование глобальных сетей	1980
Затраты на носители информации	0
Прочие затраты	89,4
Итого	3069

Структура эксплуатационных затрат изображена на рисунке 45.

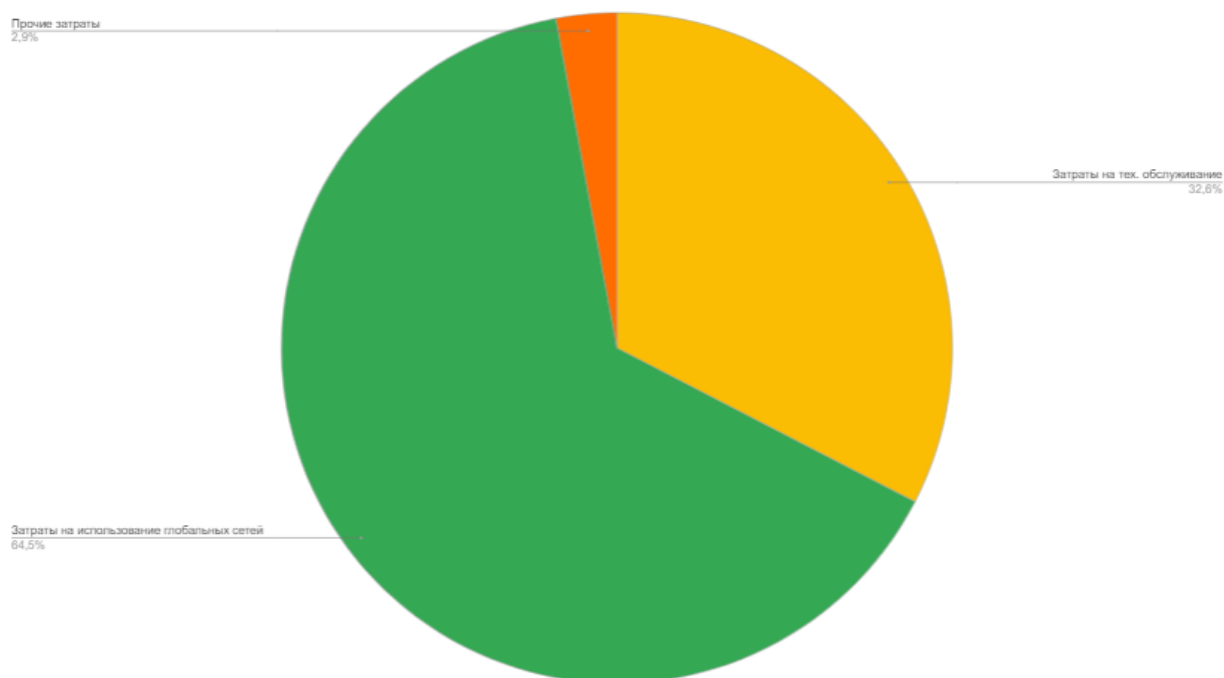


Рисунок 45 – Структура эксплуатационных затрат

### Расчет совокупной стоимости владения системой.

Прямые затраты рассчитываются по формуле 10.

$$DE = DE1 + DE2 + DE3 + DE4 + DE5 + DE6 + DE7 + DE8, \quad (10)$$

где DE1 – капитальные затраты DE1 = 44063 руб.;

DE<sub>2</sub> = 0 рублей – расходы на управление информационными

технологиями;

$DE_3 = C_{ao} + C_{то} = 0 + 1000 = 1000$  рублей –расходы на техническую поддержку автоматизированного обеспечения и программного обеспечения;

$DE_4 = 0$  – расходы на разработку прикладного программного обеспечения внутренними силами, так как нет необходимости для разработки ПО;

$DE_5 = 0$  – расходы на аутсорсинг, так как не было сторонней помощи специалистов;

$DE_6 = 0$  – командировочные расходы, не требуются командировочные расходы;

$DE_7 = 1980$  – расходы на услуги связи;

$DE_8 = C_{проч} + C_{ни} = 89,4$  – другие группы расходов.

Прямые затраты равны:  $DE = 44063 + 1000 + 1980 + 89 = 47132$  руб.

Совокупные затраты владения системой будут равны:

$ТСО = 47132,36 + 0 + 0 = 47132$  руб.

### **3.3 Оценка риска реализации проекта создания ИС**

Процесс управления рисками является актуальным и необходимым для реализации успешных ИТ-проектов.

Возможные риски проекта разработки аттестационного тестирования медицинских работников отмечены следующим списком:

1. увеличение стоимости разработки;
2. срыв сроков проекта;
3. несовместимость с используемым ПО;
4. срыв сроков опытной эксплуатации;
5. ограниченное время разработки.

Нужно определить степень влияния рисков проекта. Поэтому используется качественная оценка рисков, которая регламентируется по трехуровневой шкале влияния рисков на реализацию проекта:

- низкий уровень – малый риск, незначительное влияние на проект;

– средний уровень – риск, имеющий равную вероятность повлиять и не повлиять на результат исполнения проекта;

– высокий уровень – высокий риск, заключающийся в неисполнении проекта или в значительном изменении результата его реализации.

В таблице 9 проведен итог оценки рисков, а также возможность их предотвращения.

Таблица 9 – Перечень рисков проекта

	Риски	Уровень риска	Предотвращение
1	Срыв сроков проекта	Средний уровень	Контроль над ходом разработки
2	Несовместимость с используемым ПО	Низкий уровень	Увеличение сроков тестирования
3	Срыв сроков опытной эксплуатации	Низкий уровень	Составить четкий план
4	Ограниченное время разработки	Высокий уровень	Правильно разграничение времени на выполнение

Риском высокого уровня является ограниченное время разработки. Для его снижения, необходимо правильно составить временной интервал разработки каждого шага.

Риском среднего уровня является срыв сроков проекта. Для его предотвращения необходимо контролировать каждый шаг, во время разработки.

Рисками низкого уровня являются: несовместимость с используемым ПО и срыв опытной эксплуатации. Для их решения необходимо увеличить сроки тестирования, чтобы как можно тщательней провести тестирование, а для решения риска срыва сроков опытной эксплуатации необходимо иметь четкий план.

### **3.4 Определение доходов и экономической эффективности реализации проекта «Разработка веб-приложения для обучения языкам**



## **программирования HTML и CSS»**

### **3.4.1 Анализ рынка продуктов-аналогов**

Рынок веб-приложений для обучения языкам программирования огромен. Каждый день появляется все больше курсов, в основном они направлены на получение дохода от использования, так как после прохождения пробных заданий продолжить обучение можно только на платной основе. Также есть узконаправленные приложения, которые обучают чему-то одному, предлагают свои услуги на бесплатной основе.

Данный проект можно назвать уникальным, так как он, в основном, направлен на привлечение абитуриентов при проведении мероприятий в ХТИ – филиал СФУ, а также при поступлении на направление «Прикладная информатика», такие приложения были описаны в первом разделе.

### **3.4.2 Источники получения доходов**

Внедрение системы не имеет никаких доходов в виде денежных средств в связи с тем, что проект разрабатывается для ХТИ – филиал СФУ и направлен для привлечения абитуриентов, чьи направления обучения направлены на информатику.

Поэтому получение какого-либо дохода напрямую зависит от количества студентов. Стоимость обучения одного студента составляет примерно 70000 рублей за полгода обучения.

## **3.5 Вывод по разделу «Расчет затрат и оценка эффективности реализации проекта «Разработка веб-приложения для обучения основам**

## HTML и CSS»

Можно подвести итоги экономического раздела:

- Капитальные затраты составляют 44063 руб.
- Эксплуатационные затраты составляют 3069 руб.
- Общая стоимость владения, которая была рассчитана по методике ТСО, равна 47132 руб.

Были оценены риски в процессе реализации проекта, а также предложены варианты их предотвращения.

Полученные доходы являются косвенными, так как не направлены на прямое получение дохода, доход выражен от количества привлеченных абитуриентов в ХТИ – филиал СФУ на направление «Прикладная информатика».

## ЗАКЛЮЧЕНИЕ

В первом разделе разработаны принципы функционирования разрабатываемого веб-приложения для обучения основам HTML и CSS. Были выявлены сервисы-конкуренты, выявлены их преимущества и недостатки. Выбраны программные средства разработки.

Для программирования были выбраны такие языки как, HTML, CSS и JS, потому что данные языки не имеют аналогов. Было разработано веб-приложение для обучения основам HTML и CSS.

Во втором разделе были показаны шаги разработки веб-приложения, которое имеет пять заданий, направленных на получение основ HTML и CSS, которые абитуриент может выполнить во время проведения различных мероприятий.

В третьем разделе были рассчитаны капитальные затраты, которые равны 44063 рублей. Также были рассчитаны эксплуатационные затраты, которые равны 3069,4 рублей. Затраты на реализацию проекта выполнялись по методике ТСО, их сумма равна 47132 рублей.

Были проанализированы источники доходов от внедрения системы, благодаря чему стало понятно, что система не имеет никаких прямых доходов в виде денежных средств. Были выявлены преимущества по отношению к сайтам-конкурентов. Выявлены риски и написаны способы предотвращения их.

Также, основываясь на опыте от реализации проектов другого ВУЗа, можно сделать вывод, что при наличии необходимых разрешений, можно добавлять дополнительные баллы при поступлении в ХТИ – филиал СФУ, если абитуриент полностью справился с поставленными задачами.

## **СПИСОК СОКРАЩЕНИЙ**

ХТИ – Хакасский технический институт.

СФУ – Сибирский федеральный университет.

ИТ – информационные технологии.

ПО – программное обеспечение.

VSC– VisualStudioCode.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. CSSDinner. Главная страница [Электронный ресурс]. Режим доступа: <https://flukeout.github.io/>
2. Devacademy [Электронный ресурс]: JavaScript. – Режим доступа: <https://developer.mozilla.org/ru/docs/Web/JavaScript>
3. Devacademy [Электронный ресурс]: Основы HTML. – Режим доступа: [https://developer.mozilla.org/ru/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/HTML_basics)
4. FlexboxFroggy. Главная страница [Электронный ресурс]. Режим доступа: <https://flexboxfroggy.com/#ru>
5. GeekBrains [Электронный ресурс]: Игры, чтобы попрактиковаться в вёрстке. – Режим доступа: [https://gb.ru/posts/games\\_for\\_frontend\\_practice](https://gb.ru/posts/games_for_frontend_practice)
6. GitHub [Электронный ресурс]: SortableJS. – Режим доступа: <https://github.com/SortableJS/Sortable>
7. SeoWiki [Электронный ресурс]: CSS. – Режим доступа: <https://wiki.rookee.ru/css/>
8. Википедия [Электронный ресурс]: Визуальное программирование. – Режим доступа: [https://ru.wikipedia.org/wiki/Визуальное\\_программирование](https://ru.wikipedia.org/wiki/Визуальное_программирование)
9. Википедия [Электронный ресурс]: Скретч (язык программирования). – Режим доступа: [https://ru.wikipedia.org/wiki/Скретч\\_\(язык\\_программирования\)](https://ru.wikipedia.org/wiki/Скретч_(язык_программирования))
10. Выполнение и защита выпускной квалификационной работы по направлению 09.03.03 «Прикладная информатика» [Электронный ресурс]: методические указания / сост. Е. Н. Скуратенко, В. И. Кокова, И. В. Янченко ; Сиб. федер. ун-т, ХТИ – филиал СФУ. – Абакан : ХТИ – филиал СФУ, 2017. – Режим доступа: [https://e.sfu-kras.ru/pluginfile.php/2247348/mod\\_resource/content/1/Методичка%20ВКР.pdf](https://e.sfu-kras.ru/pluginfile.php/2247348/mod_resource/content/1/Методичка%20ВКР.pdf)

11. Интернет-лицей ТПУ [Электронный ресурс]: Интерактивные обучающие игры. – Режим доступа: <https://il.tpu.ru/game>
12. Русская документация по APIjQuery [Электронный ресурс]: Документация APIjQuery.–Режим доступа: <https://jquery-docs.ru/>
13. Сайт Хакасского технического института – филиала СФУ. [Электронный ресурс]: Информация о ХТИ – филиале СФУ. – Режим доступа: <http://khti.sfu-kras.ru/institute/>
14. Сайт Хакасского технического института – филиала СФУ. [Электронный ресурс]: Бакалавриат – Прикладная информатика. – Режим доступа: <http://khti.sfu-kras.ru/obuchenie/bakalavriat-prikladnaya-informatika.php>
15. СТО 4.2 –07– 2014 Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности, Красноярск: ИПК СФУ, 2014. – 60 с. [Электронный ресурс]. – Режим доступа: <http://about.sfu-kras.ru/node/8127>
16. Учебная и научная деятельность Анисимова Владимира Викторовича [Электронный ресурс]: Основы IDEF0. – Режим доступа: [https://www.sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6\\_2](https://www.sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6_2)
17. Учебные материалы для студентов [Электронный ресурс]:Методология IDEF3. – Режим доступа: [https://studme.org/87186/ekonomika/metodologiya\\_idef3](https://studme.org/87186/ekonomika/metodologiya_idef3)
18. Хабр [Электронный ресурс]: Sortable v1.0: Новые возможности. – Режим доступа: <https://habr.com/ru/post/246373/>
19. Яндекс.Практикум[Электронный ресурс]. – Режим доступа: <https://praktikum.yandex.ru/>

Выпускная квалификационная работа выполнена мной самостоятельно. Используемые в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

Отпечатано в одном экземпляре.

Библиография 19 наименований.

Один экземпляр сдан на кафедру.

« \_\_\_\_ » \_\_\_\_\_ 2021 г.

\_\_\_\_\_ Котуленко Василий Сергеевич

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Хакасский технический институт – филиал ФГАОУ ВО  
«Сибирский федеральный университет»

Кафедра прикладной информатики, математики и естественно-научных  
дисциплин

УТВЕРЖДАЮ

Заведующий кафедрой

Е. Н. Скуратенко

подпись

« 21 » 06 2021 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03 Прикладная информатика

Разработка Веб-приложения для обучения основам HTML и CSS

Руководитель

подпись, дата

21.06.21 доцент, канд. физ.-мат. наук А.Н. Таскин

Выпускник

подпись, дата

В. С. Котуленко

Консультанты  
по разделам:

Экономический

подпись, дата

Е. Н. Скуратенко

Нормоконтролер

подпись, дата

В. И. Кокова

Абакан 2021



Студенту Котуленко Василию Сергеевичу

Группа ХБ 17-03

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка Веб-приложения для обучения основам HTML и CSS

Утверждена приказом по институту № 222 от 08.04.2021 г.

Руководитель ВКР: А.Н. Таскин, доцент, канд. физ.-мат. наук, ХТИ – филиал СФУ

Исходные данные для ВКР: заказ ХТИ – филиала СФУ.

Перечень разделов ВКР:

1. Анализ предметной области и выбор средств проектных решений.
2. Описание процесса разработки веб-приложения.
3. Расчет затрат и оценка эффективности реализации проекта «Разработка Веб-приложения для обучения основам HTML и CSS».

Перечень графического материала: нет

Руководитель ВКР

подпись

А.Н. Таскин

Задание принял к исполнению

подпись

В.С. Котуленко

«08» апреля 2021 г.