

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

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

Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ О. В. Папина  
подпись  
«\_\_\_\_\_» \_\_\_\_\_ 2023 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.03 Прикладная информатика

Разработка веб-приложения «Дополнительное профессиональное образование»  
для ХТИ – филиала СФУ

Руководитель \_\_\_\_\_ доцент, канд. физ.-мат. наук М. А. Бурева  
подпись, дата

Выпускник \_\_\_\_\_ Д. А. Попов  
подпись, дата

Консультанты  
по разделам:

Экономический \_\_\_\_\_ М. А. Бурева  
подпись, дата

Нормоконтролер \_\_\_\_\_ А. Н. Кадычегова  
подпись, дата

Абакан 2023

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

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

Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ О. В. Папина  
подпись  
« \_\_\_\_ » \_\_\_\_\_ 2023 г.

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

Абакан 2023

Студенту Попову Данилу Андреевичу

Группа ХБ 19-02

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка веб-приложения «Дополнительное профессиональное образование» для ХТИ – филиала СФУ

Утверждена приказом по институту № 283 от 11.05.2023 г.

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

Исходные данные для ВКР: заказ ХТИ – филиала СФУ.

Перечень разделов ВКР:

1. Анализ предметной области. Выбор средств проектных решений.
2. Разработка веб-приложения «Дополнительное профессиональное образование».
3. Технико-экономическое обоснование проектного решения.

Перечень графического материала: нет

Руководитель ВКР

\_\_\_\_\_

подпись

М. А. Буреева

Задание принял к исполнению

\_\_\_\_\_

подпись

Д. А. Попов

«11» мая 2023г.

## РЕФЕРАТ

Выпускная квалификационная работа на тему «Разработка веб-приложения «Дополнительное профессиональное образование» для ХТИ – филиала СФУ» содержит 69 страниц текстового документа, 46 рисунков, 4 таблицы, 7 формул, 1 приложение, 19 использованных источников.

ВЕБ-ПРИЛОЖЕНИЕ, VSCODE, JAVASCRIPT, REACT, NODE.JS, TAILWINDCSS, РАСЧЕТ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ, ЭСКИЗЫ СТРАНИЦ, DFD, IDEF3, MYSQLWORKBENCH.

Объект исследования – деятельность отдела НМСиДО по привлечению слушателей на программы дополнительного профессионального образования (ДПО).

Предмет исследования – процесс освещения программ дополнительного профессионального образования в сети Интернет и оформление предварительной заявки на курс.

Цель работы: разработать веб-приложение «Дополнительное профессиональное образование» для увеличения количества слушателей по программам дополнительного профессионального образования ХТИ – филиал СФУ.

В первом разделе ВКР описывается основная деятельность ХТИ – филиала СФУ в направлении ДПО. На основе анализа аналогичных продуктов разработана концепция IT-проекта.

Во втором разделе описан процесс выбора и установки необходимого программного обеспечения, поэтапно описана разработка веб-приложения.

В третьем разделе выполнен расчет затрат реализации проекта по методике ТСО, проанализированы риски, которые могут возникнуть на всех этапах жизненного цикла системы.

В результате выполнения ВКР разработано веб-приложение «Дополнительное профессиональное образование» для ХТИ – филиала СФУ.

## SUMMARY

The theme of the graduation thesis is «Web application “Additional Professional Education” for KHTI, branch of SibFU». It contains 69 pages, 46 figures, 4 tables, 7 formulae, 1 application, 19 references.

WEB APPLICATIONS, VS CODE, JAVASCRIPT, REACT, NODE.JS, TAILWINDCSS, ECONOMIC EFFICIENCY CALCULATION, PAGE THUMBNAILS, DFD, IDEF3, MYSQL WORKBENCH.

The object of the study is the activity of “Science, International Relations and Additional Education Office” to attract students to additional professional education (APE) programs.

The subject of the study is the process of covering additional professional education programs on the Internet and making a preliminary application for the course.

The purpose of the work: to develop a web application “Additional Professional Education” to increase the number of students in the programs of additional professional education in the KHTI - branch of SibFU.

The first section describes the activity of KHTI - branch of SibFU in case of the APE. Considering the analysis of similar products, the concept of an IT project has been developed.

The second section describes the process of selecting and installing the necessary software; represents the steps of the web application development.

The third section provides the project implementation costs calculated according to the TCO methodology; the risks arising have been analyzed.

Results: the web application “Additional Professional Education” has been developed for the KHTI, branch of SibFU.

English language supervisor

\_\_\_\_\_

N.V. Chezybaeva

## СОДЕРЖАНИЕ

Введение .....	6
1 Анализ предметной области .....	8
1.1 Характеристика основной деятельности организации .....	8
1.2 Концепция IT-проекта .....	10
1.3 Анализ аналогичных программных продуктов и обоснование необходимости собственной разработки .....	12
1.4 Структурное моделирование бизнес-процесса онлайн-записи на курсы дополнительного образования .....	16
1.4.1 Диаграмма потоков данных DFD .....	16
1.4.2 Модель действий пользователя в веб-приложении в нотации IDEF3 ..	19
1.5 Выбор средств разработки .....	24
Выводы по разделу «Анализ предметной области» .....	29
2 Разработка веб-приложения «Дополнительное профессиональное образование» .....	29
2.1 Эскизы страниц .....	29
2.2 Описание процесса разработки главной страницы .....	34
2.3 Описание процесса разработки страницы отдельного курса .....	40
2.4 Создание личного кабинета пользователя и администратора .....	45
2.5 Предварительная запись слушателя на курс и заполнение шаблона договора .....	50
Выводы по разделу «Разработка веб-приложения "Дополнительное профессиональное образование"» .....	54
3 Оценка стоимости владения проектом .....	54
3.1 Расчет проектных затрат .....	54
3.2 Расчет капитальных затрат .....	59
3.3 Расчет эксплуатационных затрат .....	61
3.4 Расчет совокупной стоимости владения системой .....	64
Выводы по разделу «Оценка стоимости владения проектом» .....	66

ЗАКЛЮЧЕНИЕ .....	67
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	68
ПРИЛОЖЕНИЕ А .....	70

## ВВЕДЕНИЕ

Хорошо проработанный информационный ресурс – это лицо любой современной компании. Именно в интернете потенциальный клиент будет искать информацию о продукте в первую очередь. Актуальность работы состоит в том, что ни один другой ресурс не дает столько преимуществ, сколько собственный сайт, будь это визитка, Интернет-магазин или любой другой веб-ресурс.

Объект исследования – деятельность отдела НМСиДО по привлечению слушателей на программы дополнительного профессионального образования.

Предмет исследования – процесс освещения программ дополнительного профессионального образования в сети Интернет и оформление предварительной заявки на курс.

Цель работы: разработать веб-приложение «Дополнительное профессиональное образование» для увеличения количества слушателей по программам дополнительного профессионального образования ХТИ – филиал СФУ.

Для достижения поставленной цели будут решены следующие задачи:

- 1) ознакомиться с деятельностью ХТИ – филиал СФУ;
- 2) проанализировать предметную область;
- 3) проанализировать аналогичные решения;
- 4) определить архитектуру и средства разработки веб-приложения;
- 5) обосновать экономическую эффективность создания веб-приложения
- 6) разработать и протестировать веб-приложение.

В первом разделе ВКР описывается основная деятельность ХТИ – филиала СФУ. Проанализированы средства для создания веб-приложения «Дополнительное профессиональное образование» для ХТИ – филиала СФУ. Выбраны средства разработки веб-приложения «Дополнительное профессиональное образование» для ХТИ – филиала СФУ.

Во втором разделе описан процесс установки необходимого



программного обеспечения. Поэтапно описан процесс разработки веб-приложения, а также протестирован итоговый вариант.

В третьем разделе выполнен расчет затрат реализации проекта по методу ТСО, проанализированы риски, которые могут возникнуть в процессе разработки, представлены преимущества собственной разработки.

# 1 Анализ предметной области

## 1.1 Характеристика основной деятельности организации

Хакасский технический институт – филиал федерального государственного автономного образовательного учреждения высшего образования «Сибирский федеральный университет».

Миссией 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[1].

С организационной структурой ХТИ – филиала СФУ можно ознакомиться на рисунке 1.

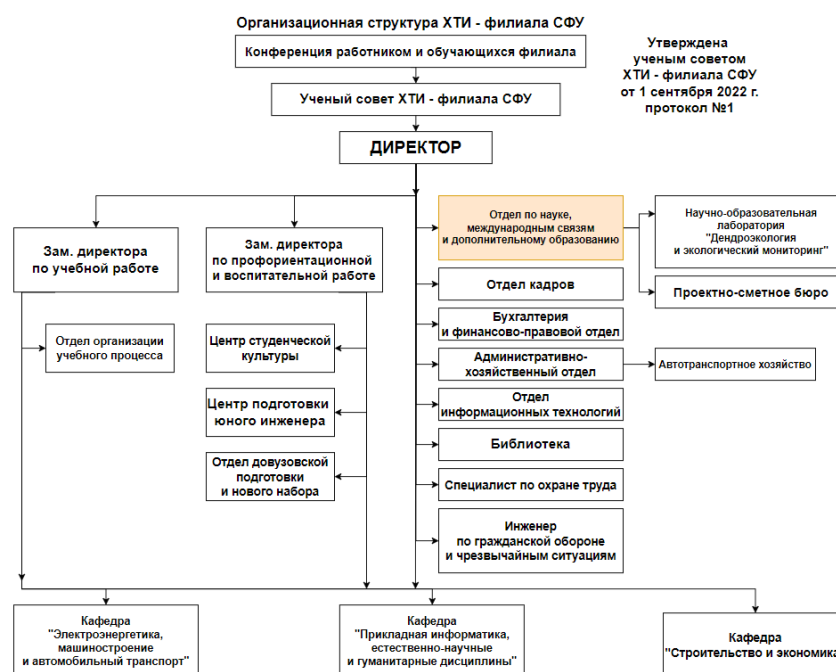


Рисунок 1 – Структура ХТИ – филиал СФУ

ХТИ – филиал СФУ имеет лицензию на осуществление образовательной деятельности серии 90Л01 № 0009304, выданная Федеральной службой по надзору в сфере образования и науки 4 июля 2016 г. (бессрочно), рег. № 2251, Свидетельство о государственной аккредитации серии 90А01 № 0003102, выданное Федеральной службой по надзору в сфере образования и науки на срок до 19 декабря 2024 г., рег. № 2957.

Институт имеет сертификат соответствия системы менеджмента качества требованиям ГОСТ ISO 9001-2015 (ISO 9001:2015) применительно к деятельности в области образования.

Институт осуществляет подготовку по очной, очно-заочной и заочной формам по 1 направлению специалитета и 6 направлениям бакалавриата, 1 направлению магистратуры.

Ведётся подготовка специалистов и бакалавров в следующих областях:

- электроэнергетика;
- машиностроение и материалобработка;
- строительство;
- экономика;
- транспорт;
- информатика.

Отдел по науке, международным связям и дополнительному образованию разрабатывает и реализует программы повышения квалификации и профессиональной переподготовки по самым востребованным направлениям: строительство, энергетика, автомобили и автомобильное хозяйство, информатика и информационные технологии, экология, экономика и управление, лингвистика.

Хакасский технический институт – филиал СФУ осуществляет образовательную деятельность по дополнительным профессиональным программам профессиональная переподготовка и повышение квалификации

для лиц, имеющих среднее профессиональное и (или) высшее образование и лиц, получающих среднее профессиональное и (или) высшее образование).

Штатное расписание отдела утверждается директором института. Внесение изменений в структуру и штатное расписание отдела осуществляется в установленном порядке по представлению начальника отдела.

Более подробно деятельность отдела описана в положении об отделе НМСиДО[2].

Одним из основных видов деятельности отдела являются координация и контроль за реализацией программ дополнительного профессионального образования.

В рамках этого вида деятельности сотрудники отдела решают следующие задачи:

- совершенствование нормативной документации, регламентирующей работу программ ДПО;
- взаимодействие с предприятиями и организациями по вопросам обучения по программам ДПО;
- формирование заявок от юридических и физических лиц для повышения квалификации и переподготовки;
- организация взаимодействия с другими подразделениями института по вопросам ДПО.

Для решения этих задач сотрудникам отдела необходимо размещать информацию для потенциальных слушателей в сети Интернет, а также на информационных стендах. Одним из средств достижения этой цели и является веб-приложение.

## **1.2 Концепция IT-проекта**

Обучение (получение дополнительного образования) персонала для большинства российских организаций в настоящее время приобретает особую актуальность. Это связано с тем, что работа в условиях рынка

предъявляет высокие требования к уровню квалификации персонала, знаниям и навыкам работников: знания, навыки, установки, которые помогали персоналу успешно работать еще вчера, сегодня теряют свою действенность. Очень быстро изменяются как внешние условия (экономическая политика государства, законодательство и система налогообложения, появляются новые конкуренты и т.п.), так и внутренние условия функционирования организации (реструктуризация предприятий, технологические изменения, появление новых рабочих мест и др.), что ставит большинство российских организаций перед необходимостью подготовки персонала к сегодняшним и к завтрашним изменениям [3].

На данный момент запись на программы повышения квалификации и профессиональной переподготовки осуществляется в очной форме. Будущему слушателю необходимо прийти лично в институт для полного оформления договора обучения.

Ключевыми моментами разработки являются:

- перенос предварительной записи в онлайн-формат;
- регистрация и создание личного кабинета слушателя;
- автоматическое формирование договора;
- автоматическое сохранение договора на компьютере слушателя;
- отправка уведомления специалисту о новом слушателе;
- добавление специалистом отдела по науке, международным связям и дополнительному образованию нового курса на сайт.

В результате реализации проекта разработки веб-приложения онлайн-записи на курсы дополнительного образования ожидается увеличение количества слушателей программ дополнительного образования.

### **1.3 Анализ аналогичных программных продуктов и обоснование необходимости собственной разработки**

На данный момент существует большое количество веб-приложений для записи на курсы повышения квалификации. Это связано с тем, что квалифицированные сотрудники, их навыки имеют спрос на рынке труда и знания, полученные на курсах, применяются на многих предприятиях и организациях, для того чтобы повысить производительность предприятия и качество конечного товара или услуги.

Разработка любого программного продукта должна проходить с учетом существующих продуктов, предлагаемых на рынке услуг, в связи с чем рассмотрим веб-приложения ВУЗов в сфере дополнительного образования.

Большинство людей оценивают компанию, опираясь на визуальные составляющие ее сайта, на котором предлагается услуга или продукт. Данному факту есть логическое объяснение. Если сайт удобен в использовании, информативен и визуально приятен, значит компания заботится о комфорте своих клиентов. Опираясь на эти рассуждения, можно сделать вывод, что дизайн – это критерий успешности бизнеса. Если визуальный контент не продумать, то немалая часть целевой аудитории (ЦА) будет утеряна [4].

Рассмотрим два веб-приложения, которые являются прямыми конкурентами в сфере курсов повышения квалификации.

Первое – веб-приложение, предлагающее услуги дополнительного образования и повышение квалификации в ТУСУРе [5].

При некорректном оформлении сайта (некачественные или стоковые изображения) у посетителя возникают сомнения в профессионализме самой компании. Поэтому, важно отслеживать резкость изображений, избегать ошибок в тексте, размещенном на картинках и придерживаться грамотного форматирования. Все элементы дизайна сайта должны быть продуманы.

Шрифты в дизайне сайта играют важную роль. При их неправильном подборе у читателя может сложиться неприятное впечатление от текста.

С первого взгляда понятно, что на сайте представлены услуги дополнительного образования на главной странице в шапке (рисунок 2). Однако, данная страница не вызывает ощущения качества и актуальности предоставляемых услуг. Подобная оценка называется первым впечатлением о сайте, благодаря которому, выделяется ЦА (целевая аудитория) бизнеса.

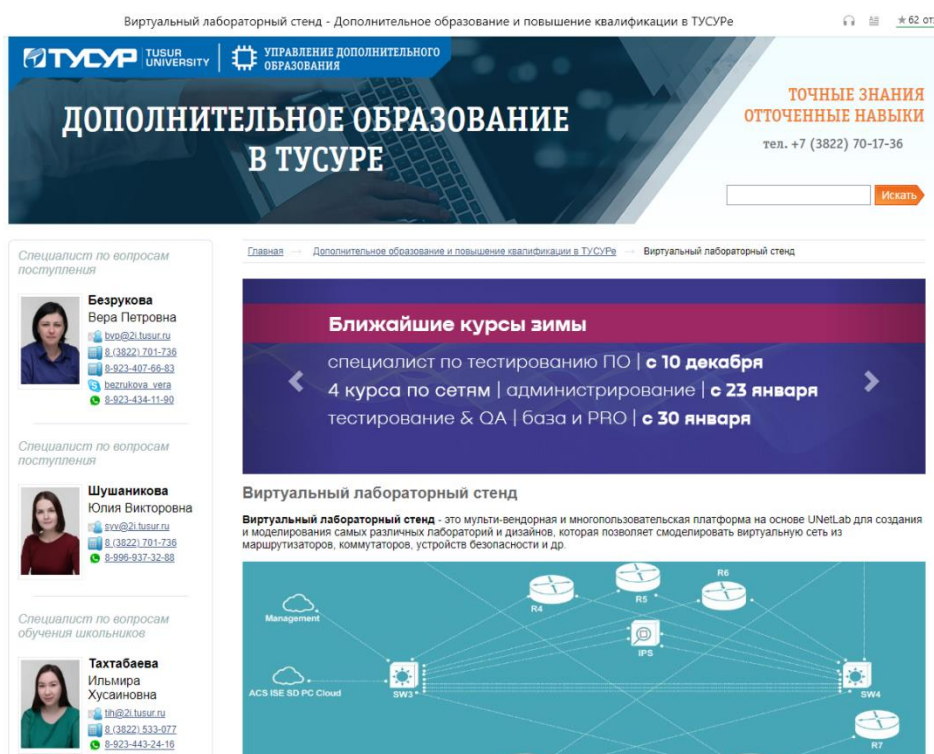


Рисунок 2 – Скриншот главной страницы ТУСУРа

Человек ежедневно посещает десятки сайтов. Далеко не каждый готов тратить время на неинтересный материал. Поэтому необходимо максимально тщательно продумывать дизайн сайта.

Множество дополнительных элементов, не несущих никакой смысловой нагрузки, скорее всего, будут только перегружать внимание и мешать концентрации на главном.

Тем не менее, представленное веб-приложение обладает всем необходимым функционалом, а именно:

- контакты специалистов;
- поиск по сайту;

- направления профессиональной переподготовки;
- направления повышения квалификации;
- элементы навигации по сайту.

Второе веб-приложение – Института непрерывного образования СФУ [6]. Скриншот главной страницы сайта представлен на рисунке 3.

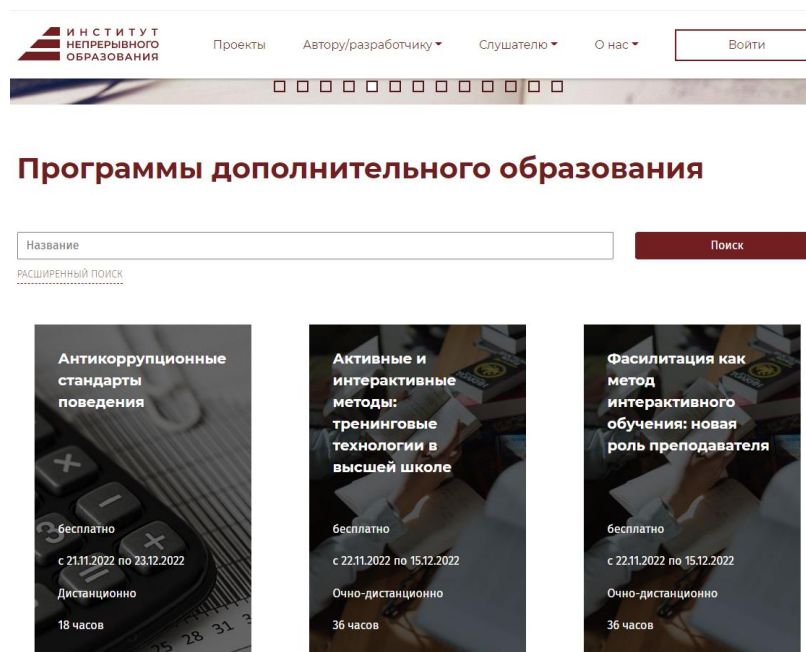


Рисунок 3 – Главная страница Института непрерывного образования СФУ

Правильная работа с пространством сайта очень важна. Главный принцип, которого следует придерживаться – это размещение связанных между собой предметов вблизи друг от друга, а несвязанных – вдали. Важно: для того, чтобы сфокусироваться на определенном элементе сайта, каждому объекту внимания необходимо давать достаточное количество свободного пространства. Это могут быть белые поля, рамки, межстрочные интервалы и так далее. В данном веб-приложении все это прекрасно реализовано.

Восприятие окружающих нас объектов во многом зависит от цвета. Он может создать определенное настроение, влиять на аппетит и подтолкнуть к принятию решения. Не стоит недооценивать этот инструмент. Важным моментом является подбор цветов. Для их гармоничного сочетания существуют



специальные приложения и сайты. Стоит помнить, что использование множества контрастных цветов затрудняет восприятие информации. На изображении выше цвета выполняют не только функцию создания нужного настроения, но и организуют пространство, логично расставляя акценты.

Человек на подсознательном уровне всегда склоняется к симметрии. Поэтому сбалансированный дизайн относительно горизонтальной и вертикальной осей смотрится наиболее выигрышно. Симметрия также помогает в расстановке приоритетов в выделенном пространстве.

Дизайн должен быть органичным, сбалансированным и привлекательным. Но не стоит забывать о его главной задаче – выгодном представлении товара. Для компании, которая за счет сайта хочет увеличить прибыль, нет смысла тратить деньги на дизайн, который не будет работать. Сайт Института непрерывного образования демонстрирует отличный пример, когда вся визуальная составляющая сайта направлена на то, чтобы выигрышно представить товар.

Веб-приложение обладает всем необходимым функционалом:

- расширенный поиск по сайту;
- удобная навигация в «шапке» страницы;
- доступ к контактам для связи со специалистом в три клика;
- адрес и положение на карте;
- онлайн запись на курс;
- личный кабинет пользователя.

Институт предоставляет возможность обучения по следующим направлениям:

- Иностранные языки.
- Образование.
- Менеджмент. Экономика. Право.
- Промышленность.
- Связь. Информационные технологии.

- Спорт, медицина и здоровье.
- Сфера услуг.
- Социальная работа, административно-управленческая деятельность, волонтеры.
- Строительство и благоустройство территорий. Энергетика.
- Электронное обучение и дистанционные образовательные технологии.

Средняя стоимость образовательных услуг составляет 43 120 рублей.

Стоит отметить, что имеются курсы с бесплатным обучением по актуальным направлениями повышения квалификации.

## **1.4 Структурное моделирование бизнес-процесса онлайн-записи на курсы дополнительного образования**

### **1.4.1 Диаграмма потоков данных DFD**

DFD (акроним DataFlowDiagrams) – методология графического структурного анализа, описывающая внешние по отношению к системе, источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ [7].

При проектировании информационных систем, диаграмма DFD наглядно отображает течение информации в пределах процесса или системы: входящие и выходящие потоки данных, точки хранения информации и пути ее передвижения между источниками и пунктами доставки.

Контекстная диаграмма потоков данных разрабатываемого веб-приложения представлена на рисунке 4.

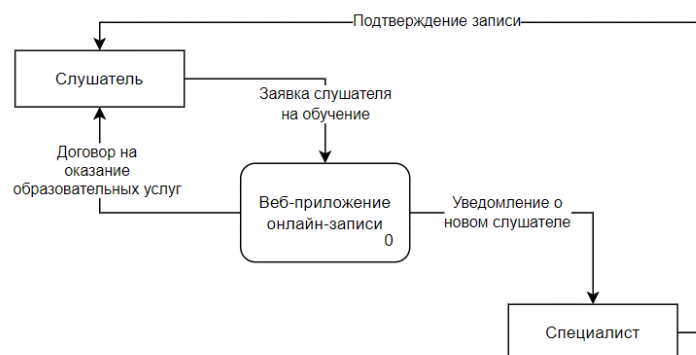


Рисунок 4 – Контекстная диаграмма потоков данных

Внешними сущностями веб-приложения являются «Слушатель» и «Специалист».

Слушатель – любой пользователь сети Интернет, который может перейти к веб-приложению. Является потенциальным получателем образовательных услуг по курсам дополнительного образования.

Специалист – это сотрудник отдела по науке, международным связям и дополнительному образованию. Получает уведомления по новым заявкам на обучение, связывается с потенциальным слушателем для уточнения информации и заключения договора.

Декомпозиция блока «Веб-приложение онлайн-записи» представлена на рисунке 5.

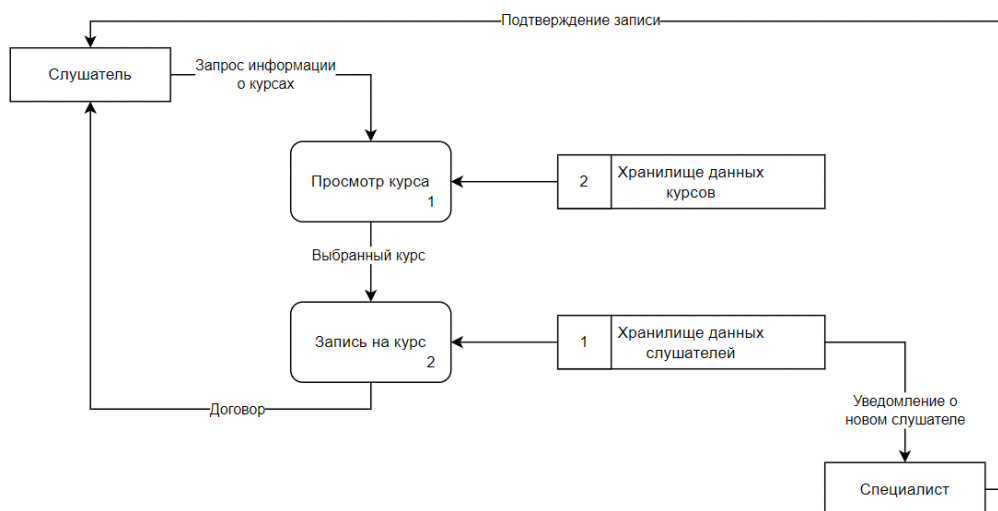


Рисунок 5 – Декомпозиция блока «Веб-приложение онлайн-записи»

При просмотре курса слушатель получает информацию из хранилища данные о курсах и их руководителях.

На странице курса пользователь может записаться на обучение. Декомпозиция блока «Запись на курс» представлена на рисунке 6.

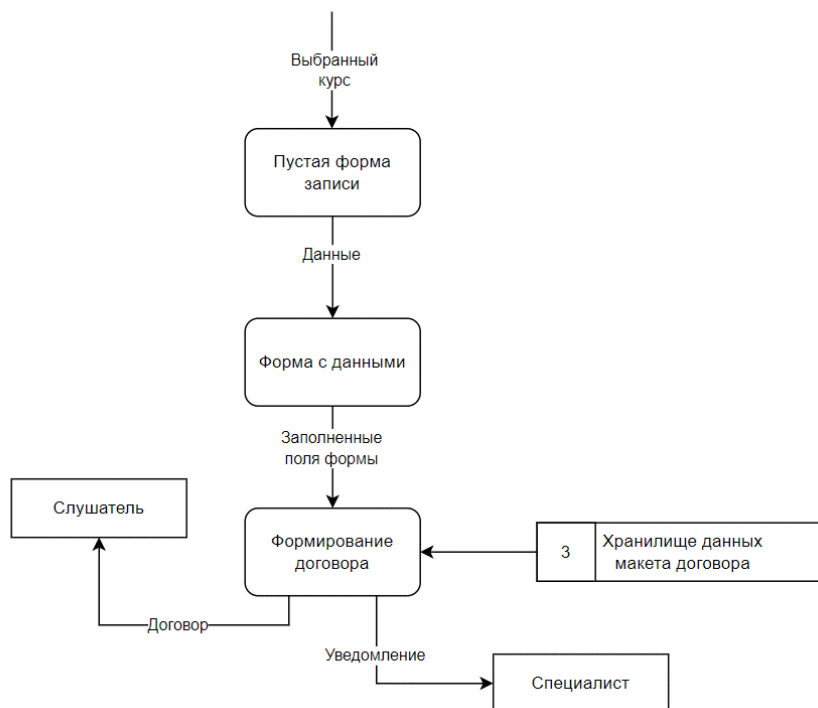


Рисунок 6 – Декомпозиция блока «Запись на курс»

Хранилище данных слушателей содержит файл макета договора об оказании образовательных услуг, файл учебного плана по программе дополнительного образования

В результате моделирования потоков данных определено две сущности и три хранилища. Подробнее со структурой хранилищ можно ознакомиться на информационно-логической модели базы данных, представленной на рисунке 7.

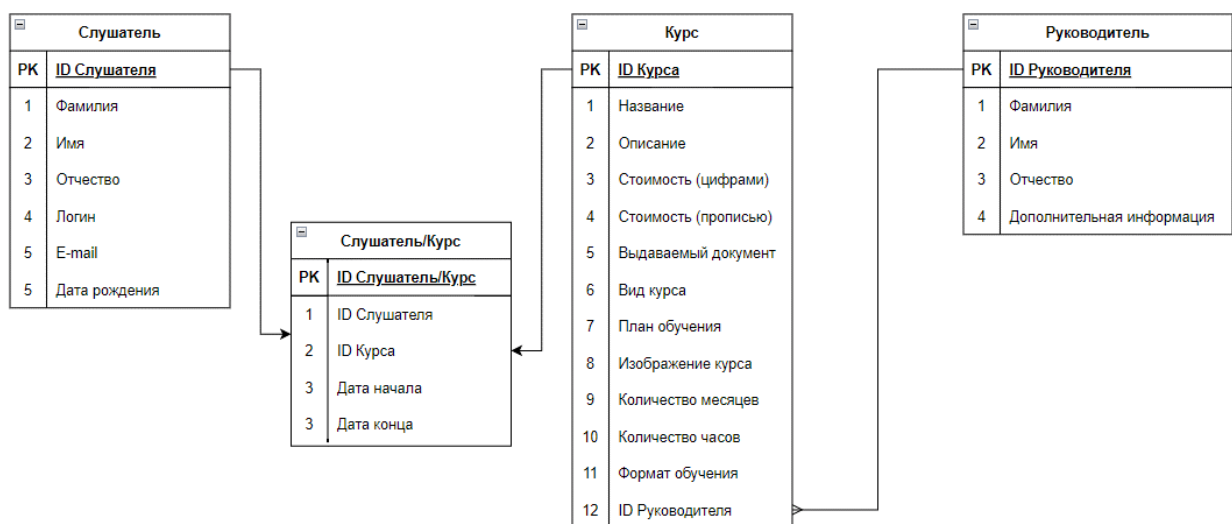


Рисунок 7 – Информационно-логическая модель базы данных

Информационно-логическая модель отображает данные предметной области в виде совокупности информационных объектов и связей между ними. Эта модель представляет данные, подлежащие хранению в базе данных.

#### 1.4.2 Модель действий пользователя в веб-приложении в нотации IDEF3

IDEF – (акроним Integrated Computer-Aided Manufacturing) - комплексное автоматизированное производство или интегрированное компьютерное производство.

IDEF3 – методология моделирования и стандарт документирования процессов (в т. ч. технологических процессов), происходящих в системе, а также механизм сбора информации о процессах. IDEF3 показывает причинно-следственные связи между ситуациями и событиями, используя структурный метод выражения знаний о том, как функционирует система, процесс или предприятие [8].

Цель IDEF3 – описать ситуацию, когда процессы выполняются в определенной последовательности, и объекты, участвующие совместно в одном процессе.

Процессная модель IDEF3 должна обеспечить:

1. Отображение последовательности процессов.
2. Демонстрацию логики взаимодействия элементов системы.

Контекстная модель действий пользователя разрабатываемого веб-приложения представлена на рисунке 8.

Для пользователя существует четыре основных страницы, на которые он может перейти, а именно: главная страница веб приложения, страница с контактной информацией, страница личного кабинета и страница конкретного курса.

На каждой странице пользователь может совершить характерные для нее действия, описанные в представленной на рисунке 7 модели.

К общим действиям относятся: записаться на курс, отправить электронное письмо, позвонить по номеру, выбрать курс, переход между страницами, изменение личных данных.

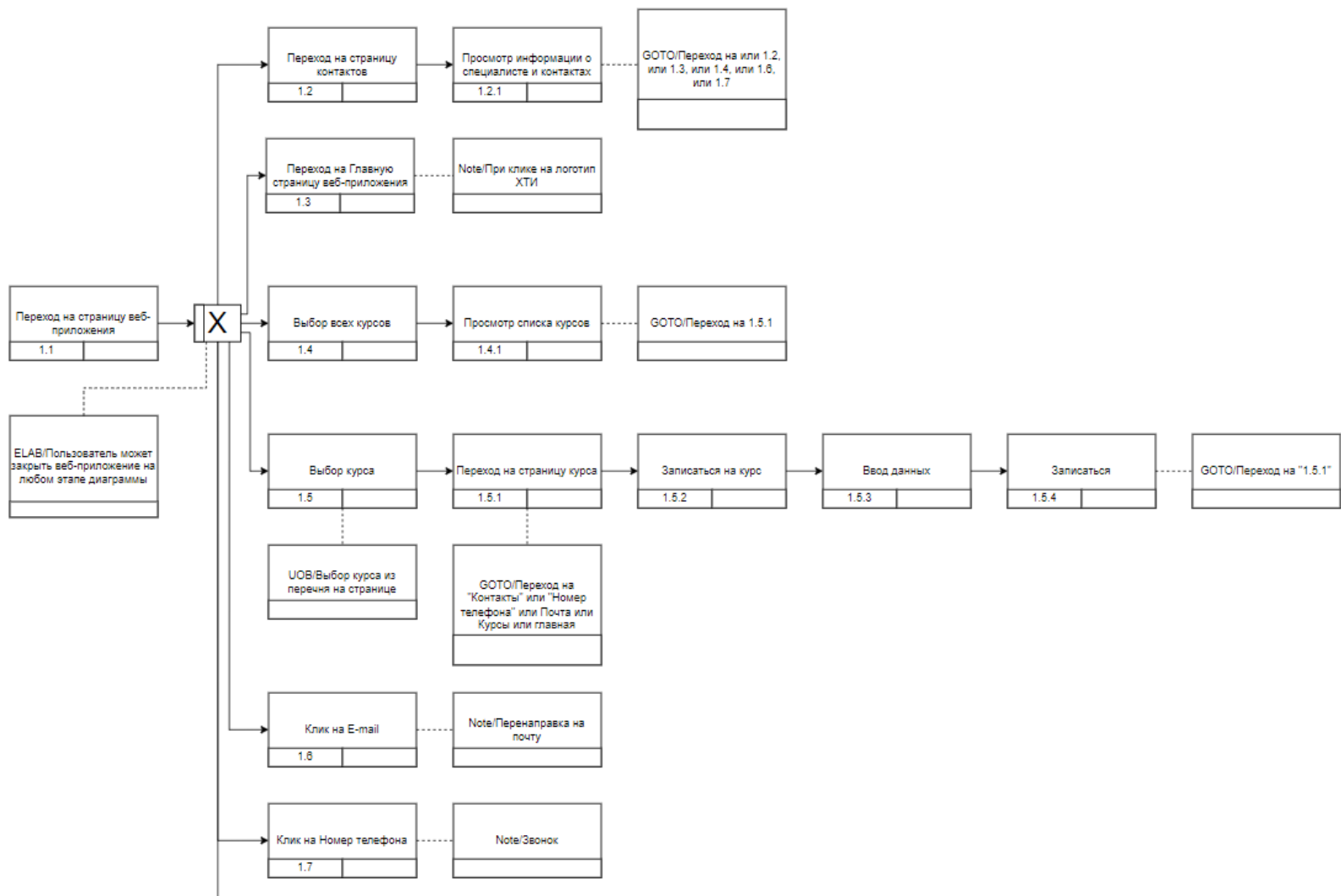


Рисунок 8 – Контекстная модель IDEF3, лист 1

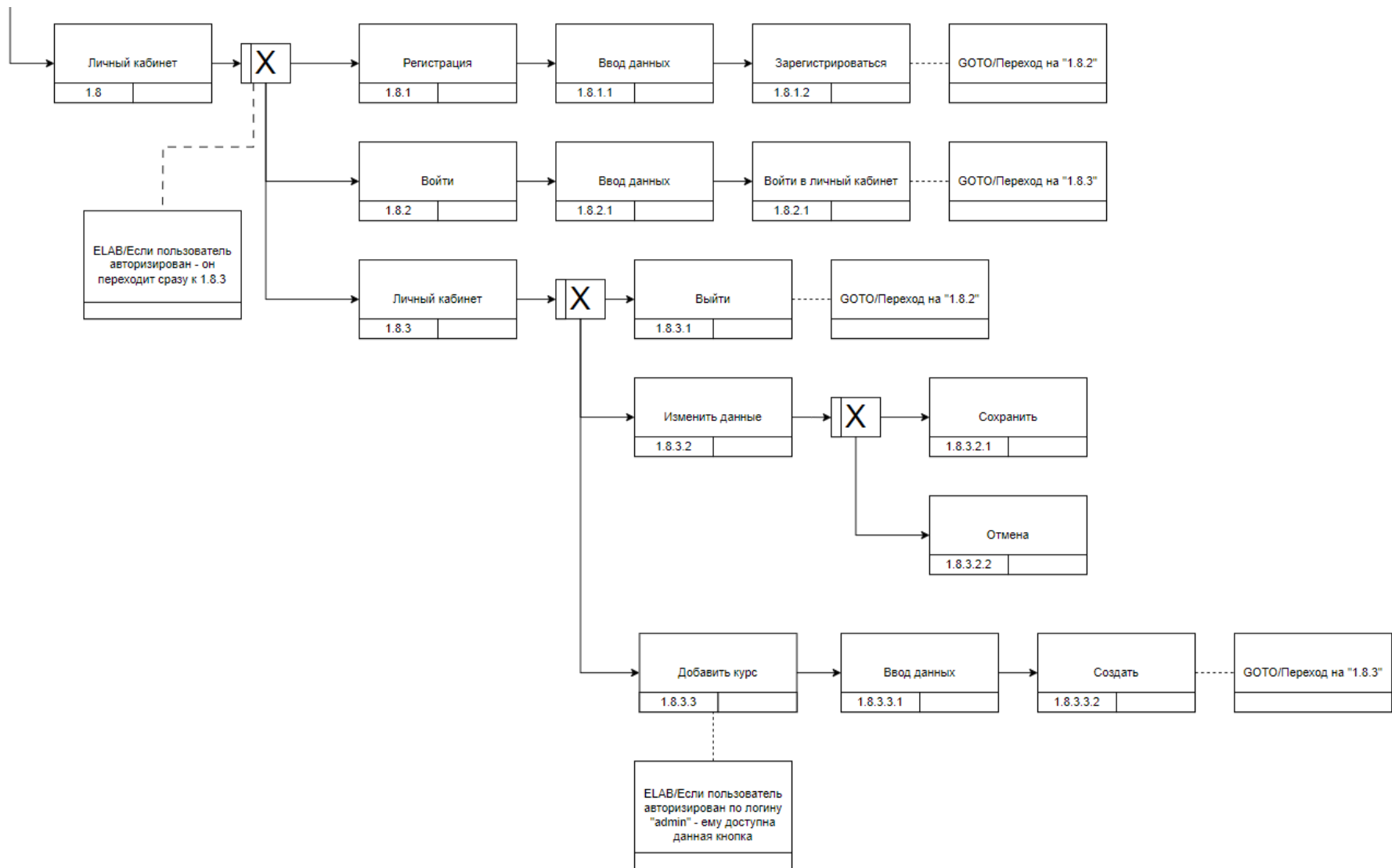


Рисунок 8, лист 2



Существуют различные типы перекрестков. В контекстной модели IDEF3 используется эксклюзивное (исключающее) «ИЛИ». Смысл в случае слияния стрелок: только один предшествующий процесс должен быть завершен; а смысл в случае разветвления стрелок: только один следующий процесс запускается.

В декомпозиции блока 1.5.3 «Ввод данных», представленном на рисунке 9, используется синхронное «И». Смысл в случае слияния стрелок: все предшествующие процессы должны быть завершены одновременно; а смысл в случае разветвления стрелок: все последующие процессы запускаются одновременно.

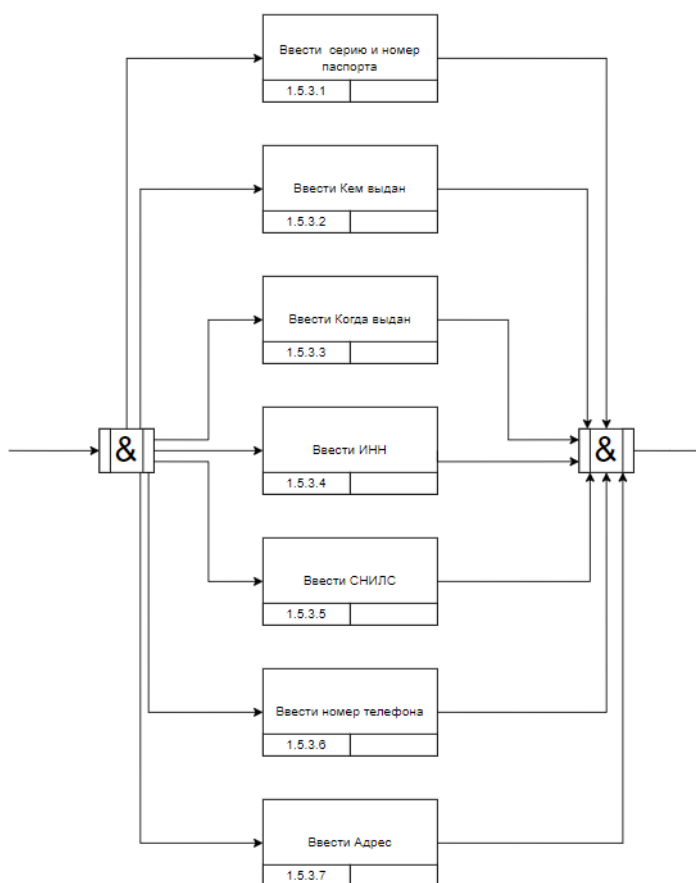


Рисунок 9 – Декомпозиции блока 1.5.3 «Ввод данных»

## 1.5 Выбор средств разработки

Для разработки веб-приложения был выбран редактор кода VSCode. VisualStudioCode – один из самых популярных редакторов исходного кода, используемых программистами.

Это бесплатный, с открытым исходным кодом и кроссплатформенный редактор, который работает на Windows, Linux и macOS, так что можно работать независимо от платформы, на которой основано устройство.

Как и любой другой популярный редактор, VisualStudioCode также обеспечивает экстремальную настройку благодаря своей гибкой настройке предпочтений и множеству расширений. VSC дает возможность изменить тему, изменить сочетания клавиш, настроить настройки, создать фрагменты кода и многое другое [9].

Чтобы установить VisualStudioCode необходимо перейти на официальный сайт редактора и нажать кнопку «Download for Windows» (рисунок 10).

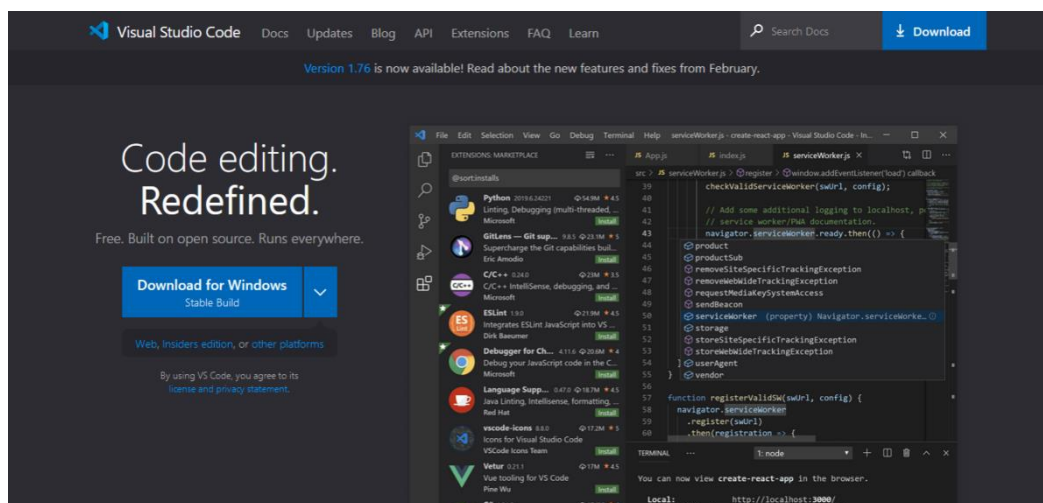


Рисунок 10 – Официальный сайт VisualStudioCode

Для разработки frontend части были выбраны следующие программные решения: React, TypeScript, ReactRouter DOM, ReduxToolkit, Tailwind CSS.

React (иногда React.js или ReactJS) – JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов. React может

использоваться для разработки одностраничных и мобильных приложений. Его цель – предоставить высокую скорость разработки, простоту и масштабируемость. В качестве библиотеки для разработки пользовательских интерфейсов React часто используется с другими библиотеками, такими как MobX, Redux и GraphQL [10].

TypeScript – язык программирования, представленный Microsoft в 2012 году и позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript. TypeScript является обратно совместимым с JavaScript и компилируется в последний. Фактически, после компиляции программу на TypeScript можно выполнять в любом современном браузере или использовать совместно с серверной платформой Node.js. Код экспериментального компилятора, транслирующего TypeScript в JavaScript, распространяется под лицензией Apache [11].

ReactRouter DOM – это модуль узла, который предназначен для маршрутизации в веб-приложениях. Он позволяет инженерам создавать маршруты для одностраничного приложения React. Декларативная маршрутизация – это стиль кодирования, используемый в React и ReactRouter. Декларативные маршруты React являются компонентами и используют ту же инфраструктуру, что и любое приложение React. Эти маршруты связывают веб-адреса с определенными страницами и другими компонентами, используя мощный механизм рендеринга React и условную логику для программного включения и выключения маршрутов [12].

ReduxToolkit – это сторонняя библиотека, такая как Redux, созданная и поддерживаемая командой Redux. Пакет ReduxToolkit предназначен для стандартного способа написания логики Redux. Весь смысл использования ReduxToolkit состоит в том, чтобы сделать использование Redux проще и эффективнее [13].

Tailwind CSS – CSS фреймворк открытым исходным кодом. Главная особенность этой библиотеки заключается в том, что, в отличие от других CSS-фреймворков, таких как Bootstrap, она не предоставляет серию

предопределенных классов для элементов, таких как кнопки или таблицы. Вместо этого он создает список "служебных" классов CSS, которые можно использовать для оформления каждого элемента путем смешивания и сопоставления. Концепция, основанная на полезности, относится к основной отличительной черте Tailwind. Вместо создания классов вокруг компонентов (кнопка, панель, меню, текстовое поле и т.д.), классы создаются вокруг определенного элемента стиля (желтый цвет, жирный шрифт, очень крупный текст, центральный элемент и т.д.). Каждый из этих классов называется служебными классами [14].

JavaScript – язык, который в своей основе содержит скрипт. Работать со скриптами сможет каждый браузер, что делает язык кроссплатформенным. JavaScript – это мультипарадигменный язык. Он поддерживает несколько стилей одновременно: функциональный, объектно-ориентированный, императивный. На практике используется в виде встраиваемого для обеспечения программного доступа к объектам софта.

К преимуществам JavaScript можно отнести следующее:

- Наличие полной интеграции с версткой страниц и серверной частью.
- Скорость работы. JS может сделать итоговое приложение более быстрым и удобным.
- Комфортность использования пользовательских интерфейсов.
- Производительность.

Для работы со сторонними JavaScript библиотеками и сборки проекта необходимо установить пакетный менеджер NPM.

NPM (NodePackageManager) – дефолтный пакетный менеджер для JavaScript, работающий на Node.js. Менеджер npm состоит из двух частей:

- CLI (интерфейс командной строки) – средство для размещения и скачивания пакетов;
- онлайн-репозитории, содержащие JS пакеты.

Структуру репозитория `npmjs.com` можно представить, как центр исполнения заказов, который получает товары (`npm`-пакеты) от продавцов (авторы пакетов) и распространяет эти товары среди покупателей (пользователи пакетов).

В центре исполнения заказов (`npmjs.com`) в качестве персональных менеджеров для каждого покупателя работает армия вомбатов (`npm CLI`) [15].

Он устанавливается автоматически вместе со средой выполнения `Node.js`. Поэтому, необходимо скачать `Node.js` на компьютер. Для этого нужно перейти на официальный сайт `Node.js` и нажать одну из кнопок под заголовком “Download for Windows (x64)” (рисунок 11). Здесь есть две версии – `stable` и `latest`. `Stable` версия содержит не так много новых функций как `latest`, но зато она уже протестирована и менее подвержена ошибкам. Поэтому необходимо нажать кнопку скачивания с версией `18.15.0 LTS` и запустить установщик, после чего среда `Node.js` и пакетный менеджер `NPM` установятся на компьютер.

`Node.js` — это программная среда на JavaScript-движке (`V8`). Используется для компиляции JS-кода в машинный язык. Другими словами, эта программная платформа транслирует исходный JS непосредственно в машинный код, который и будет исполняться уже на серверной стороне.

Важно уточнить: `Node.js` не является самостоятельным языком программирования, это только программная платформа для использования JavaScript на серверной стороне. При этом она использует подходы событийно-ориентированного и так называемого реактивного программирования. Чаще всего, `Node.js` задействуется в качестве веб-сервера, также эта платформа может применяться и для создания десктопных программ и других целей (например, для программирования микроконтроллеров). Платформа `Node.js` появилась в 2009 году и сегодня используется очень широко, в самых разнообразных отраслях разработки. По типу программной платформы она относится к средам выполнения.

`Node.js` — это платформа, которая умеет компилировать код JavaScript напрямую в машинный код. Таким образом достигается трансформация

JavaScript из специализированного языка для браузера в язык программирования общего назначения [16].

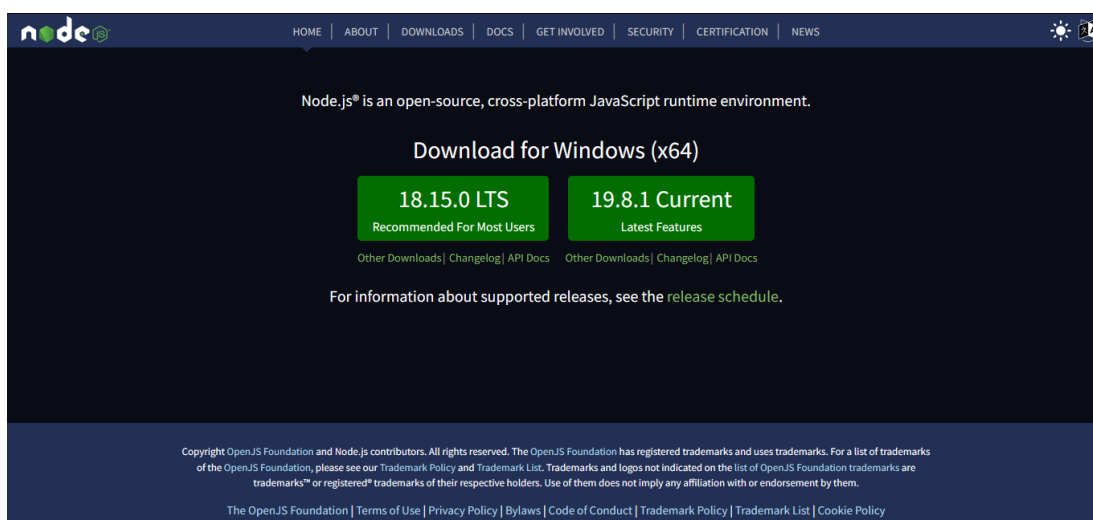


Рисунок 11 – Официальный сайт Node.js

Чтобы проверить, что NPM действительно был установлен, можно набрать в терминале следующую команду: `npm --version`. Если установка прошла успешно, то в терминале, в качестве ответа, можно будет увидеть текущую версию NPM (рисунок 12).

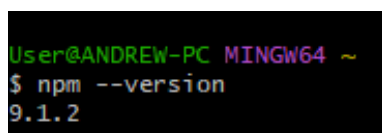


Рисунок 12 – Версия NPM

В данном случае, текущая версия NPM 9.1.2, что значит, что NPM был успешно установлен на компьютер.

Для хранения данных веб-приложения будет создана база данных. Исходя из требований заказчика и личных предпочтений разработчика в работе будет использоваться СУБД MySQL.

MySQL — свободная реляционная система управления базами данных (СУБД). Под словом «свободная» подразумевается ее бесплатность, под

«реляционная» – работа с базами данных, основанных на двумерных таблицах. MySQL универсальна, и применяется при разработке веб-сайтов, веб-приложений и корпоративных баз данных начального уровня [17].

Подобранные программные средства предоставляют все возможности для выполнения разработки веб-приложения. Они бесплатны, мощны и многофункциональны.

### **Выводы по разделу «Анализ предметной области»**

Таким образом, в результате анализа предметной области была составлена характеристика основной деятельности ХТИ – филиала СФУ по организации дополнительного профессионального образования, проанализированы программные продукты по данной тематике и составлена концепция IT-проекта.

В разделе представлена диаграмма потоков данных DFD и создана модель действий пользователя в веб-приложении в нотации IDEF3.

Проанализированы и выбраны средства разработки для создания веб-приложения.

## **2 Разработка веб-приложения «Дополнительное профессиональное образование»**

### **2.1 Эскизы страниц**

Эскиз главной страницы веб-приложения представлен на рисунке 13.

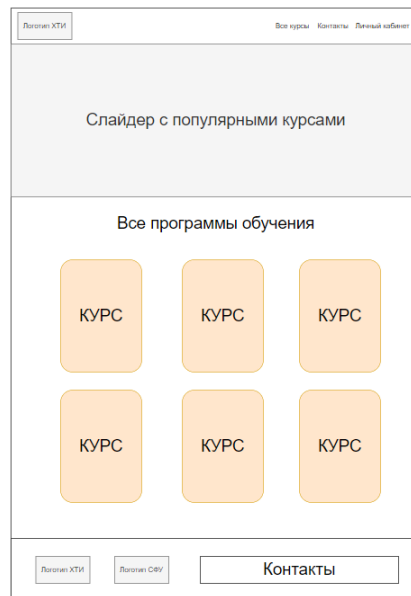


Рисунок 13 – Главная страница веб-приложения

Эскиз страницы отдельного курса представлен на рисунке 14.



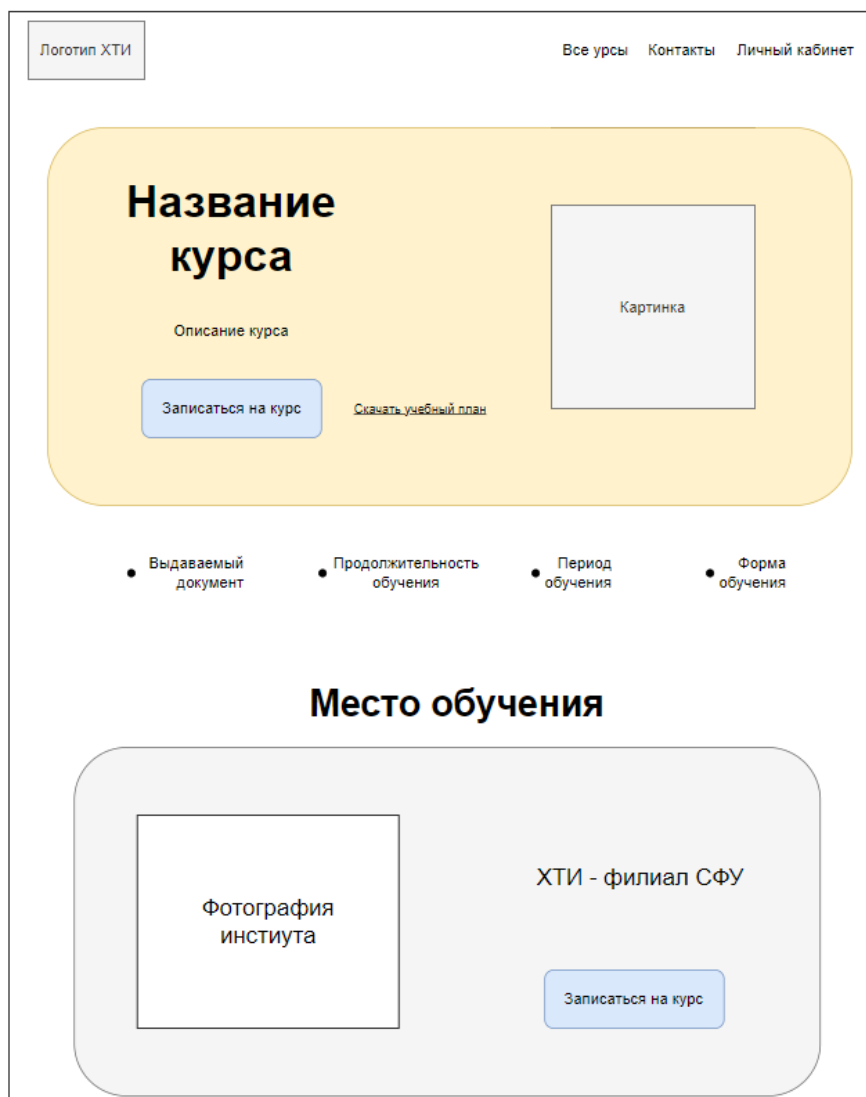


Рисунок 14– Страница курса, лист 1

<b>Руководитель курса</b>	
Фотография Руководителя	Фамилия Имя Отчество Дополнительная информация
<b>Есть вопрос?</b>	
Фотография Специалиста	Фамилия Имя Отчество Дополнительная информация Телефон E-mail
<b>Стоимость обучения</b>	
	<a href="#">Записаться на курс</a>
Логотип ХТИ	Логотип СФУ
<b>Контакты</b>	

Рисунок 14, лист 2

Эскиз личного кабинета пользователя представлен на рисунке 15.

Рисунок 15 – Личный кабинет пользователя

Эскиз формы записи на курс и формы регистрации представлены на рисунках 16 и 17 соответственно.

Рисунок 16 – Форма записи на курс

The image shows a registration form titled "Регистрация" (Registration) enclosed in a rounded rectangle. It contains the following elements from top to bottom: a text input field for "Фамилия" (Surname), a text input field for "Имя" (Name), a text input field for "Отчество" (Patronymic), a text input field for "Дата рождения" (Date of Birth), a text input field for "E-mail", a text input field for "Логин" (Login), a text input field for "Пароль" (Password), a dark blue button with the text "Зарегистрироваться" (Register), and a link "Уже есть аккаунт? Войти" (Already have an account? Log in) in blue text.

Рисунок 17 – Форма регистрации личного кабинета нового слушателя

Разработанные эскизы помогут: понять дальнейший объем работ, сформировать задание для верстки, избежать ошибок при дальнейшей разработке.

## 2.2 Описание процесса разработки главной страницы

Создание веб-приложения начинается с верстки интерфейса главной страницы. Файл MainPages.tsx содержит необходимый код (рисунок 18), в результате выполнения которого на странице отображается графический интерфейс главной страницы.

Файл TSX - это файл TypeScript (.TS), написанный с использованием синтаксиса JSX. Он содержит код, который является частью веб-приложения. Файлы TSX можно открыть в любом текстовом редакторе, но они предназначены для открытия в редакторах исходного кода.

JSX, также известный как JavaScript XML, является расширением языка программирования JavaScript. Разработчики используют синтаксис JSX при

разработке пользовательских интерфейсов и других компонентов приложения с использованием React, библиотеки JavaScript с открытым исходным кодом, разработанной Facebook и поддерживаемой им [18].

```
frontend > src > pages > MainPage.tsx > MainPage
59
60 ..... <h2 className="text-center font-extrabold text-4xl mb-5">
61 ..... Все программы обучения
62 ..... </h2>
63
64 ..... <div className="w-full flex justify-center flex-wrap gap-3">
65 ..... {
66 .....   courses.map(course => {
67 .....     return (
68 .....       <CourseCard
69 .....         key={course.id}
70 .....         id={course.id}
71 .....         title={course.title}
72 .....         months={course.months}
73 .....         image={course.image}
74 .....       />
75 .....     );
76 .....   });
77 ..... }
78 ..... </div>
79 ..... </section>
80 ..... );
81 ..... };
82
83 export default MainPage;
```

Рисунок 18 – Создание главной страницы

В этом же файле присутствует фрагмент кода (рисунок 19), который создает слайдер на главной странице.

```
MainPage.tsx X
frontend > src > pages > MainPage.tsx > MainPage
11
12 const dispatch = useAppDispatch();
13 const courses = useAppSelector(state => state.courses.courses);
14
15 useEffect(() => {
16   | dispatch(fetchCourses());
17   }, []);
18
19 return (
20   <section className="w-full">
21     <Header />
22
23     <div className="w-full flex justify-center">
24       <Carousel
25         showThumbs={false}
26         showStatus={false}
27         autoplay={true}
28         className="mb-10 w-full max-w-[85%]"
29       />
30     </div>
31   </section>
32 );
```

Рисунок 19 – Создание слайдера на главной странице

Для формирования верстки требуется информация о курсах, которая хранится в базе данных. На рисунках 20-23 представлен терминальный интерфейс с созданием таблиц БД с помощью MySQLWorkbench.

MySQL Workbench — это кроссплатформенный инструмент проектирования реляционных баз данных с открытым исходным кодом, который добавляет функциональность и упрощает разработку MySQL и SQL. Он объединяет проектирование, разработку, создание, администрирование и обслуживание SQL, а также предлагает графический интерфейс для структурированной работы с базами данных [19].

```
contacts

CREATE TABLE `contacts` (
  `id` int NOT NULL AUTO_INCREMENT,
  `fullname` varchar(200) NOT NULL,
  `job_title` varchar(500) NOT NULL,
  `address` varchar(400) NOT NULL,
  `email` varchar(100) NOT NULL,
  `phone` varchar(100) NOT NULL,
  `picture` varchar(2000) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE InnoDB,
  CHARSET utf8mb4,
  COLLATE utf8mb4_0900_ai_ci;
```

Рисунок 20 – Таблица контактов

```
courses

CREATE TABLE `courses` (
  `id` int NOT NULL AUTO_INCREMENT,
  `title` varchar(100) NOT NULL,
  `description` varchar(1000) NOT NULL,
  `price_number` int NOT NULL,
  `price_letter` varchar(2000) NOT NULL,
  `reward` varchar(200) NOT NULL,
  `kind` varchar(300) NOT NULL,
  `image` varchar(2000) NOT NULL,
  `course_plan` varchar(2000) NOT NULL,
  `curator` int NOT NULL,
  `begin` varchar(50) NOT NULL,
  `end` varchar(50) NOT NULL,
  `format` varchar(100) NOT NULL,
  `hours` int NOT NULL,
  `months` int NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE InnoDB,
  CHARSET utf8mb4,
  COLLATE utf8mb4_0900_ai_ci;
```

Рисунок 21 – Таблица курсов

```
curators

CREATE TABLE `curators` (
  `id` int NOT NULL AUTO_INCREMENT,
  `fullname` varchar(200) NOT NULL,
  `degree` varchar(1000) NOT NULL,
  `picture` varchar(2000) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE InnoDB,
  CHARSET utf8mb4,
  COLLATE utf8mb4_0900_ai_ci;
```

Рисунок 22 – Таблица кураторов

```
users

CREATE TABLE `users` (
  `id` int NOT NULL AUTO_INCREMENT,
  `lastname` varchar(100) NOT NULL,
  `firstname` varchar(100) NOT NULL,
  `surname` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `birthday` varchar(100) NOT NULL,
  `login` varchar(100) NOT NULL,
  `password` varchar(300) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE InnoDB,
  CHARSET utf8mb4,
  COLLATE utf8mb4_0900_ai_ci;
```

Рисунок 23 – Таблица пользователей

Команда `CREATETABLE` создает таблицу с указанным именем в текущей БД.

Команда `PRIMARYKEY` создает первичный ключ из указанного поля.

Команда в поле `Engine` показывает, что таблица хранится в подсистеме хранения данных `InnoDB`.

С результатом работы кода в файле `MainPages.tsx` можно ознакомиться на рисунке 24.

## Прикладная информатика

Программа направлена на формирование компетенций, необходимых для ведения профессиональной деятельности в областях, связанных с использованием вычислительной техники и информационных технологий: связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации информационных систем, управления их жизненным циклом).



### Все программы обучения

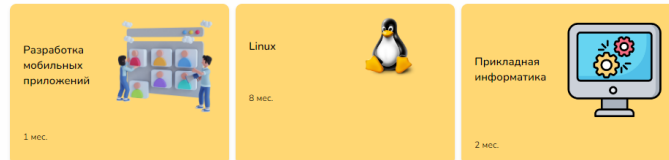


Рисунок 24 – Графическое отображение главной страницы

Далее рассмотрим создание страницы контактов. Необходимый фрагмент кода содержится в файле ContactsPage.tsx (рисунок 25).

```
frontend > src > pages > ContactsPage.tsx > ContactsPage
49
50
51   {
52     contacts.map((contact) => {
53       return (
54         <ContactCard
55           image={contact.picture}
56           name={contact.fullname}
57           job={contact.job_title}
58           phone={contact.phone}
59           email={contact.email}
60           address={contact.address}
61         />
62       );
63     });
64   }
65 </div>
66 </section>
67 };
68
69 export default ContactsPage;
```

Рисунок 25 – Создание страницы контактов



В этом же файле содержится код (рисунок 26), благодаря которому мы можем получить необходимую информацию о преподавателе из базы данных и сформировать карточку отдельного контакта.

```
--
12  useEffect(() => {
13    (async () => {
14      const response = await axios.get(`${BASE_API_URL}/api/contacts/all`, {
15        withCredentials: true
16      });
17
18      return response.data;
19
20    })().then((value) => {
21      setContacts(value)
22    })
23  }, []);
--
```

Рисунок 26 – Получение данных о преподавателе и БД

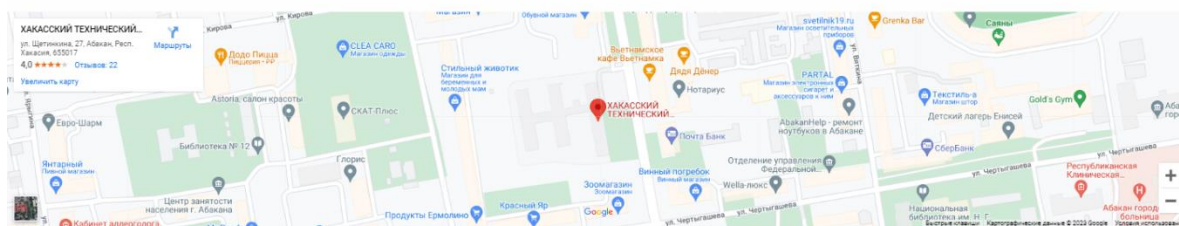
Так же на страницу контактов интегрирована карта из сервиса GoogleMaps. С фрагментом кода, который выполняет данную интеграцию можно ознакомиться на рисунке 27.

```
35  <iframe
36    src="https://www.google.com/maps/embed?pb=!1m3m3!1m2!1s0x5d2c3d5525e15a49%3A0xe8193e8a3!5m2!1sru!2sru"
37    height="350"
38    allowFullScreen={true}
39    loading="lazy"
40    referrerPolicy="no-referrer-when-downgrade"
41    className="outline-none w-full mb-10">
42  </iframe>
```

Рисунок 27 – Интеграция карты

С результатом выполнения кода в файле ContactsPage.tsx можно ознакомиться на рисунке 28.

## О нас



## Наши сотрудники



**Пулкова Татьяна Сергеевна**  
Специалист  
ул. Щетинкина, д.27, ауд. 116  
Телефон: 8 (3902) 22-53-55  
Email: nims-kht@yandex.ru



**Плотникова Татьяна Николаевна**  
Кандидат технических наук, доцент  
г. Абакан, ул. Щетинкина, 27, ауд. 213  
Телефон: 8 (3902) 22-53-55  
Email: nims-kht@yandex.ru

Рисунок 28– Страница контактов

## 2.3 Описание процесса разработки страницы отдельного курса

Следующим этапом разработки было создание шаблона страницы отдельного курса. С фрагментами кода файла `CourseDetails.tsx`, создающего графическое отображение всей информации о курсе можно ознакомиться на рисунке 29.

```

frontend > src > pages > CourseDetails.tsx > CourseDetails
1  import { useNavigate, useParams } from "react-router-dom";
2  import { Contact, Course, Curator, Student } from "../types/data";
3  import { useEffect, useState } from "react";
4  import Header from "../components/Header";
5  import axios from "axios";
6  import CuratorCard from "../components/CuratorCard";
7  import PriceCard from "../components/PriceCard";
8  import Modal from "../components/Modal";
9  import CloseIcon from "../assets/icons/close-icon.svg";
10 import GetCourseForm from "../components/GetCourseForm";
11 import { BASE_API_URL } from "../utils/constants";
12 import ContactCard from "../components/ContactCard";
13 import useAuth from "../hooks/useAuth";
14 import KhtiImage from "../assets/images/khti-image.jpg";
15
16 const CourseDetails: React.FC = () => {
17
18     const [course, setCourse] = useState<Course | null>(null);
19     const { id } = useParams();
20     const { isAuth } = useAuth();
21
22     const [isModalVisible, setIsModalVisible] = useState(false);
23
24     const [student, setStudent] = useState<Student>({
25         seriesAndNumberOfPassport: "",
26         inn: "",
27         snils: "",
28         course_id: id ? +id : 0,
29         whenGetPassport: "",
30         whereGetPassport: "",
31         address: "",
32         phone: ""
33     });
34
35     const [contact, setContact] = useState<Contact>();
36

```

Рисунок 29 – Фрагмент кода, создающего страницу курса

В этом же файле, с помощью кода мы получаем информацию с БД (базы данных) о деталях курса и о кураторе курса (по аналогии со страницей контактов). Фрагмент кода представлен на рисунке 30.

```

38
39 useEffect(() => {
40   (
41     async () => {
42       const response = await axios.get(`${BASE_API_URL}/api/courses/${id}`, {
43         withCredentials: true
44       });
45       const data = response.data;
46
47       const curatorResponse = await axios.get(`${BASE_API_URL}/api/curators/${data[0]["curator"]}`, {
48         withCredentials: true
49       });
50       const curator = curatorResponse.data;
51
52       console.log(curator);
53
54       return {
55         ...data[0],
56         curator: curator[0]
57       };
58     }
59   )
60   ().then(data => {
61     setCourse(data);
62   })
63   , []];
64

```

Рисунок 30 – Фрагмент кода, запрашивающий информацию из БД

В строке 42 мы получаем информацию о самом курсе. Строка 47 отвечает за получение данных о кураторе курса. Строка 61 – сохранение данных.

Так же в данном файле реализована функция ограничения записи на курсы. С кодом, которое формирует условие можно ознакомиться на рисунке 31.

```

CourseDetails.tsx ×
frontend > src > pages > CourseDetails.tsx > CourseDetails
//
78 const getCourse = (e: React.MouseEvent<HTMLButtonElement, MouseEvent>) => {
79   e.preventDefault();
80
81   if (isAuth) {
82     setIsModalVisible(true);
83
84   } else {
85     navigate("/sign-in")
86   }
87 };
88

```

Рисунок 31– Условие допуска к записи на курс

Логика вышепредставленного кода проста: если пользователь авторизован – показывается форма записи на курс, а в противном случае (не авторизован) происходит редирект на страницу входа в личный кабинет.


Это делается с целью создания пользователем аккаунта в системе и хранения о нем ограниченного количества данных, собранных в процессе регистрации, которые в дальнейшем будут использоваться для уведомления специалиста отдела по науке, международным связям и дополнительному образованию о новом слушателе.

Графический результат выполнения кода представлен на рисунке 32.

The screenshot shows a course page for "Разработка мобильных приложений" (Development of mobile applications). The page features a blue header with navigation links: "Все курсы", "Контакты", and "Личный кабинет". The main content area has a light blue background and includes the course title, a brief description, and two buttons: "Записаться на курс" (Enroll in the course) and "Скачать учебный план" (Download the syllabus). To the right is an illustration of two people interacting with a large screen displaying user avatars. Below the main content, there are four key details: "По окончании курса" (Upon course completion) leading to "Удостоверение" (Certificate), "Длительность" (Duration) of "1 мес." (1 month), "Период" (Period) from "с 01.09.2023 по 01.10.2023", and "Формат" (Format) as "Онлайн, очно-заочник, заочный" (Online, part-time, distance). Below this is the heading "Место обучения" (Place of study), followed by a photograph of the XTI building and the text "ХТИ - Филиал СФУ" (XTI - Branch of SFEDU) with a "Записаться на курс" button.


Рисунок 32 – Страница курса, лист 1

**Куратор курса**





**Янченко Инна Валериевна**  
Кандидат педагогических наук

**Есть вопрос?**



**Пулыова Татьяна Сергеевна**  
Специалист  
ул. Щетинкина, д.27, пуд. 116  
Телефон: 8 (3902) 22-53-55  
Email: tims\_khtb@yandex.ru

**Стоимость обучения** **10000 руб.** Записаться на курс



СИБИРСКИЙ  
ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ | SIBIRIAN  
FEDERAL  
UNIVERSITY

г. Абакан, ул. Щетинкина, 27, каб. 116 тел.: 8-983-191-42-39, 8 (3902) 22-53-55 (доб. 162).  
E-mail: cdo@khtb.ru

Рисунок 32, лист 2

Описанная ранее форма записи на курс представлена на рисунке 33.

**Записаться на курс** ✕

**Записаться**

\* мы не собираем и не используем эти данные

Рисунок 33 – Форма записи на курс

## 2.4 Создание личного кабинета пользователя и администратора

Веб-приложение «Дополнительное профессиональное образование» для ХТИ – филиала СФУ, как упомянуто ранее, должно обладать личным кабинетом отдельно взятого пользователя.

Создание личного кабинета пользователя и стало следующим этапом разработки. С фрагментом кода файла ProfilePage.tsx можно ознакомиться на рисунке 34.

```
frontend > src > pages > ProfilePage.tsx > ProfilePage
4 import { User } from "../types/data";
5 import Header from "../components/Header";
6 import axios from "axios";
7 import ProfileIcon from "../assets/icons/profile-icon.svg";
8 import { useNotification } from "../hooks/useNotification";
9
10 const ProfilePage: React.FC = () => {
11
12     const [user, setUser] = useState<User>({
13         fullname: "",
14         login: "",
15         birthday: "",
16         email: ""
17     });
18
19     const { successNotification, errorNotification } = useNotification();
20
21     const navigate = useNavigate();
22
23     const [edit, setEdit] = useState(false);
24
25     useEffect(() => {
26         (async () => {
27             const response = await axios.get(`${BASE_API_URL}/api/users/me`, {
28                 withCredentials: true
29             });
30             return response.data[0];
31         })().then((data) => {
32             console.log(data);
33             setUser({
34                 fullname: `${data.lastname} ${data.firstname} ${data.surname}`,
35                 email: data.email,
36                 birthday: data.birthday,
37                 login: data.login
38             });
39         });
40     })
```

Рисунок 34 – Фрагмент файла ProfilePage.tsx, лист 1

```

43  ✓   const logout = async () => {
44  ✓       try {
45  ✓           await axios.get(`${BASE_API_URL}/api/auth/logout`, {
46  ✓               |   withCredentials: true
47  ✓           });
48  ✓           navigate("/sign-in");
49
50  ✓       } catch (error) {
51  ✓           console.log(error);
52  ✓       }
53  ✓   };
54
55  ✓   const updateUserInfo = async () => {
56  ✓       await axios.put(`${BASE_API_URL}/api/users/me`, user, {
57  ✓           |   withCredentials: true
58  ✓       })
59  ✓       .then(() => {
60  ✓           |   successNotification("Данные успешно обновлены!");
61  ✓       })
62  ✓       .catch((reason) => {
63  ✓           |   errorNotification(reason.response.data);
64  ✓       })
65
66  ✓       setEdit(false);
67  ✓   };

```

Рисунок 34, лист 2

```

87  ✓   <div className="w-full flex gap-2">
88  ✓       <button
89  ✓           className="bg-stone-300 border-stone-400 border-2 p-2 rounded-2xl"
90  ✓           onClick={(e) => {
91  ✓               if (edit) {
92  ✓                   setEdit(prev => !prev);
93  ✓               }
94  ✓               (async () => {
95  ✓                   const response = await axios.get(`${BASE_API_URL}/api/users/me`, {
96  ✓                       |   withCredentials: true
97  ✓                   });
98  ✓                   return response.data[0];
99  ✓               })().then((data) => {
100  ✓                   console.log(data);
101  ✓                   setUser({
102  ✓                       |   fullname: `${data.lastname} ${data.firstname} ${data.surname}`,
103  ✓                       |   email: data.email,
104  ✓                       |   birthday: data.birthday,
105  ✓                       |   login: data.login
106  ✓                   });
107  ✓               })
108  ✓           } else {
109  ✓               setEdit(prev => !prev)
110  ✓           }
111  ✓       })
112  ✓   >
113  ✓   </div>
114  ✓   {!edit ? "Изменить данные" : "Отмена"}
115  ✓   </button>

```

Рисунок 34, лист 3



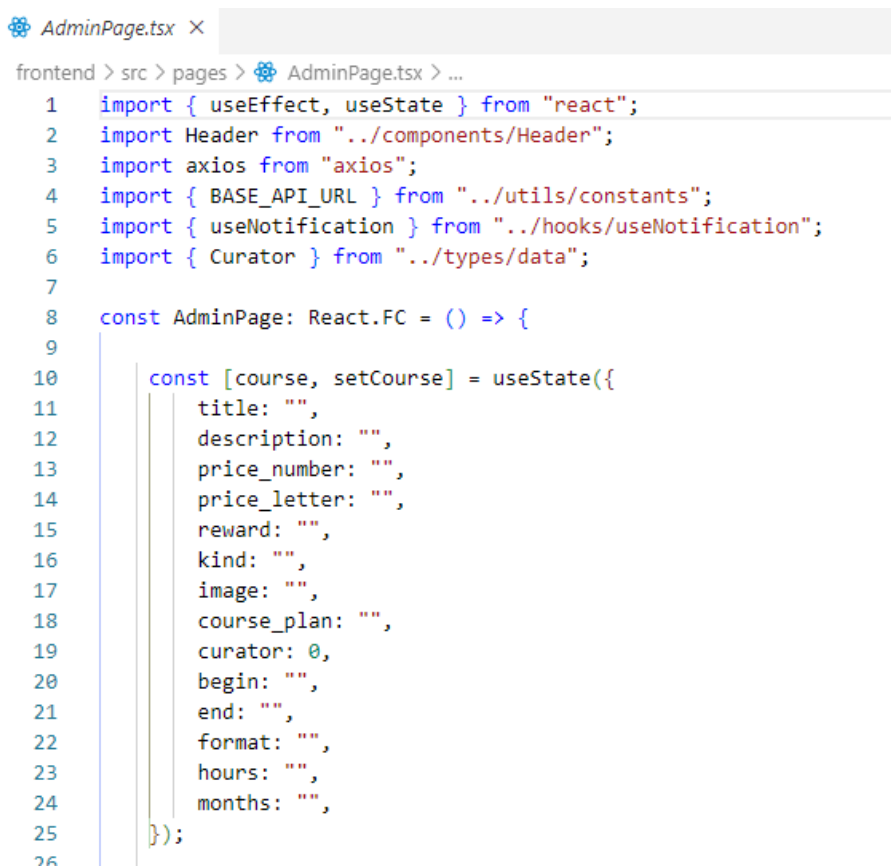
В личном кабинете реализована роль администратора систем. У администратора есть привилегированное право – добавлять новый курс. Для этого создано условие, с которым можно ознакомиться на рисунке 35.

```
<Link
  | to={"/admin"}
  | className={`bg-stone-300 border-stone-400 border-2 p-2 rounded-2xl ${user.login === "admin" ? "block" : "hidden"} `}
  | >
  | Добавить курс
  | </Link>
```

Рисунок 35 – фрагмент файла ProfilePage.tsx

Логика следующая: если логин пользователя –admin, то мы добавляем на странице личного кабинета кнопку «Добавить курс», при нажатии которой происходит редирект на форму добавления нового курса на сайт.

С фрагментом кода, с помощью которого создавалась страница формы добавления нового курса, можно ознакомиться на рисунке 36.



```
AdminPage.tsx ×
frontend > src > pages > AdminPage.tsx > ...
1 import { useEffect, useState } from "react";
2 import Header from "../components/Header";
3 import axios from "axios";
4 import { BASE_API_URL } from "../utils/constants";
5 import { useNotification } from "../hooks/useNotification";
6 import { Curator } from "../types/data";
7
8 const AdminPage: React.FC = () => {
9
10   const [course, setCourse] = useState({
11     title: "",
12     description: "",
13     price_number: "",
14     price_letter: "",
15     reward: "",
16     kind: "",
17     image: "",
18     course_plan: "",
19     curator: 0,
20     begin: "",
21     end: "",
22     format: "",
23     hours: "",
24     months: "",
25   });
26
```

Рисунок 36 – Создание формы нового курса

С результатом работы вышеописанных фрагментов кода можно ознакомиться на рисунках 37 и 38.

Центр дополнительного образования

## Информация о профиле

Изменить данные

Логин: КиняевТест1

Имя: Киняев Федор Иванович

Email: КунТ111@mail.ru

День рождения: 01.01.2001

Выйти

Рисунок 37 – Личный кабинет пользователя

## Информация о профиле

Изменить данные

Добавить курс

Логин: admin

Имя: Админов Админ Админович

Email: admin@gmail.com

День рождения: 01.01.2001

Выйти

Рисунок 38 – Личный кабинет администратора

Форма добавления нового курса на сайт представлена на рисунке 39.

## Добавить новый курс

Награда

Удостоверение о повышении квалификац ▼

Вид курса

Повышение квалификации ▼

Куратор

Буреева Мария Александровна ▼

Создать

Рисунок 39 – Форма добавления нового курса на сайт

## 2.5 Предварительная запись слушателя на курс и заполнение шаблона договора

Одной из главных функций разрабатываемого веб-приложения должна являться функция автозаполнения шаблона договора записи слушателя на курс. Это стало следующим этапом разработки веб-приложения.

Для этого в веб-приложение были добавлены шаблоны договора с динамическими переменными (рисунок 40).

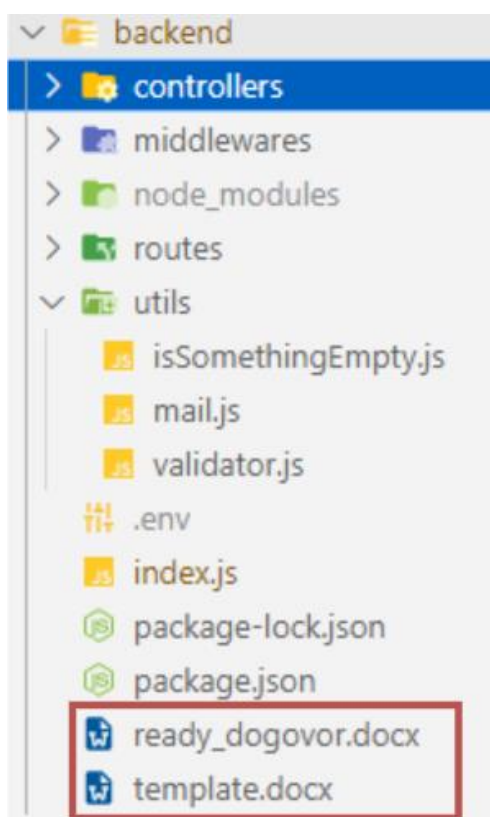


Рисунок 40 – Добавление шаблона договора в веб-приложение

Фрагмент кода, с помощью которого формировался договор представлен на рисунке 41.

```

9  export const getStudent = (req, res) => {
10
11     const form = req.body;
12
13     db.query(`SELECT lastname, firstname, surname, email, birthday FROM users WHERE id=${req.user.id}`, (err, data) => {
14         const student = data[0];
15
16         if (!student) {
17             return res.status(500).json("Errorrrrrn");
18         }
19
20         db.query(`SELECT title, price_number, price_letter, kind, hours, reward, end FROM courses WHERE id=${form.course_id}`, (err, data) => {
21             const course = data[0];
22
23             if (!course) {
24                 return res.status(500).json("No course");
25             }

```

Рисунок 41 – Формирование договора

Строка 11 – получение данных с формы записи на курс (рисунок 33).

Строка 13 – получение недостающих данных пользователе из БД.

Строка 20 – получение данных о курсе, который выбрал пользователь при записи.

На рисунке 42 указаны динамические значения, которые изменятся в договоре.

```

39     doc.render({
40         fullname: `${student.lastname} ${student.firstname} ${student.surname}`,
41         passport_seria: `${form.seriesAndNumberOfPassport.slice(0, 4)}`,
42         passport_number: `${form.seriesAndNumberOfPassport.slice(4)}`,
43         snils: form.snils,
44         inn: form.inn,
45         email: student.email,
46         birthday: student.birthday,
47         price_number: course.price_number,
48         price_letter: course.price_letter,
49         reward: course.reward,
50         hours: course.hours,
51         course_title: course.title,
52         kind_of_course: course.kind,
53         when_get_passport: form.whenGetPassport,
54         where_get_passport: form.whereGetPassport,
55         phone: form.phone,
56         student_address: form.address,
57         end: course.end
58     });

```

Рисунок 42 – Динамические значения в шаблоне договора

Сами динамические значения в шаблоне выглядят следующим образом (рисунок 43):

Гражданин РФ {fullname}, именуемый(ая) в дальнейшем «Заказчик», действующий на основании паспорта серия {passport seria} № {passport number}, выдан {where get passport} {when get passport}, и {fullname}, именуемый(ая) в дальнейшем «Обучающийся», с другой стороны, заключили настоящий договор о нижеследующем:

#### Рисунок 43 – Фрагмент шаблона договора до заполнения

Динамические значения заменятся на личные данные слушателя, но в базе данных не сохраняются. Потенциальный слушатель сможет ознакомиться со всеми условиями соглашения не рискуя распространить свои персональные данные и защитить от их возможной утечки при потенциальной атаке на БД.

Фрагмент заполненного договора выглядит следующим образом (рисунок 44):

Гражданин РФ Киняев Федор Иванович, именуемый(ая) в дальнейшем «Заказчик», действующий на основании паспорта серия 1111 № 222222, выдан МВД по Республике Хакасия 01.01.2021, и Киняев Федор Иванович, именуемый(ая) в дальнейшем «Обучающийся», с другой стороны, заключили настоящий договор о нижеследующем:

#### Рисунок 44 – Фрагмент договора после заполнения

Заполненный договор представлен в приложении А.

Сразу после формирования и сохранения договора на персональный компьютер пользователя, необходимо оповестить специалиста отдела по науке, международным связям и дополнительному образованию о новом потенциальном слушателе.

Это стало задачей следующего этапа разработки. На рисунке 45 приведен фрагмент кода, с помощью которого реализуется данная функция.

```

backend > utils > mail.js > ...
1  import nodemailer from "nodemailer";
2
3  export const sendMail = (text) => {
4    const transporter = nodemailer.createTransport({
5      service: "gmail",
6      auth: {
7        user: " ", "@gmail.com",
8        pass: " ",
9      }
10   });
11
12   const mailOptions = {
13     from: " ", "@gmail.com",
14     to: " ", "@mail.ru",
15     subject: "НОВЫЙ ПОЛЬЗОВАТЕЛЬ ЗАПИСАЛСЯ НА КУРС",
16     text
17   };
18
19   transporter.sendMail(mailOptions, (err, info) => {
20     if (err) {
21       console.log(err);
22     } else {
23       console.log(info)
24     }
25   });
26
27 };

```

Рисунок 45 – Функция отправки уведомления на эл. почту

Строка 4 – функция авторизация в аккаунте почты, с которой будет отправка письма (уведомления).

Строка 12 – Формирование письма (уведомления).

Строка 19 – Отправка письма (уведомления) специалисту.

С результатом функция можно ознакомиться на рисунке 46.

## НОВЫЙ ПОЛЬЗОВАТЕЛЬ ЗАПИСАЛСЯ НА КУРС



@gmail.com Сегодня, 13:25

Кому: вам

Название курса: Разработка мобильных приложений  
 Телефон для связи: 8(983)111-11-11

Рисунок 46 – Уведомление на почту

Таким образом, было разработано и протестировано веб-приложение «Дополнительное профессиональное образование» для ХТИ – филиала СФУ.

### **Выводы по разделу «Разработка веб-приложения "Дополнительное профессиональное образование"»**

В разделе представлены созданные эскизы страниц веб-приложения, описан процесс разработки основных страниц и форм веб-приложения, а также представлен результат выполнения программного кода.

Протестирован окончательный вариант. Веб-приложение обладает всем необходимым функционалом и выполняет поставленные перед ним задачи.

## **3 Оценка стоимости владения проектом**

Для определения экономической ликвидности разработки проводится технико-экономическое обоснование проекта.

Цель технико-экономического обоснования (цель ТЭО) – доказать и обосновать эффективность инвестиционного проекта для кредиторов или руководителей организации, в случае если финансовые результаты подготовки технико-экономического обоснования и ТЭР (подготовки ТЭО и ТЭР) положительны.

### **3.1 Расчет проектных затрат**

Формула (1) применяется для вычисления затрат на проектирование веб-приложения

$$K_{\text{пр}} = K_{\text{зп}} + K_{\text{ипс}} + K_{\text{свт}} + K_{\text{проч}}, \quad (1)$$

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



Кипс– затраты на инструментальные программные средства для проектирования;

Ксвт– затраты на средства вычислительной техники для проектирования;

Кпроч– прочие затраты на проектирование.

Заработная плата разработчиков вычисляется с учетом районного и северного коэффициентов. В Хакасии действует районный коэффициент и северная надбавка в размере 1,3 (или 30%), введенный постановлением Правительства РФ от 3 декабря 1992 года N 933 «О районном коэффициенте к заработной плате на территории Республики Хакасия».

В 2022 году в России действует обновленный перечень северных территорий, утвержденный постановлением Правительства от 16.11.2021 № 1946. Республика Хакасия не оказалась ни в списке районов крайнего севера, ни в списке приравненных местностей. Процентная надбавка, в отличие от районного коэффициента, начисляется в зависимости от длительности трудового стажа:

– За первый год работы к зарплате добавляется 10 процентов.

– За каждые следующие два года – по 10% до получения суммарных 30 процентов (или коэффициент 0,3).

Расчет заработной платы разработчика происходит по следующим соображениям:

–  $20\,390 \text{ руб. (Оклад)}/184 \text{ ч. (максимальное кол-во рабочих часов за месяц)} = 110,82 \text{ руб./час.}$

–  $110,82 \text{ руб./час} * 43 \text{ (отработанных часов за месяц)} = 4765,26 \text{ руб.}$

Доходы от коэффициентов суммируются с заработной платой. После всех вычислений получают общие затраты на заработную плату разработчикам. От итоговой зарплаты разработчика отчисляется НДФЛ в размере 0,13 (или 13%).

Доходы и расходы представлены в таблице 1.

Таблица 1 – Затраты на заработную плату разработчика

Доходы, руб.		Расходы, руб.	
Заработная плата	4765,26	НДФЛ	13% (-991,17)
Районный коэффициент	30% (+1430)		
Северный коэффициент	30% (стаж > 5 лет)		
Сумма	7624,42		
Всего на руки	6633,25		

В фонд оплаты труда входят обязательные отчисления на пенсионное, медицинское и социальное обеспечение (единый социальный налог).

- Пенсионное страхование – 22%.
- Медицинское страхование – 5,1%.
- Социальное страхование на случай временной нетрудоспособности и материнства – 2,9%.
- Социальное страхование от несчастных случаев и профессиональных заболеваний – 0,2%.

Итого ФОТ = затраты на зарплату разработчика + 30,2% от зарплаты.

$$K_{зп} = 7624,42 * 1,302 = 9927 \text{ руб.}$$

Период работы 10.04.23 – 20.05.23. Заработная плата за весь период составит 19854 руб. (9927 руб./мес.\*2 мес.).

Для разработки проекта необходимо специальное оборудование. Требуется посчитать стоимость на все элементы оборудования, чтобы узнать общие затраты.

В таблице 2 представлены технические характеристики компьютера, использовавшегося при разработке веб-приложения. Стоимость

комплектующих взята с сайта магазина цифровой и бытовой техники DNS (дата обращения 16.12.2022).

Таблица 2 – Характеристики компьютера

Комплектующие	Наименование	Цена, руб.
Системный блок ПК DEXP Atlas H323	Intel Core i5-10400, 6x2.9 ГГц, 8 ГБ DDR4, SSD 240Гб	37 500
Клавиатура проводная	DEXP K-5003BU	1 200.
Мышь беспроводная	Jet.A R300G черный	1 100.
23.8" Монитор MSI Pro MP241X черный	1920x1080@75 Гц, VA, 3000 : 1, 250 Кд/м <sup>2</sup> , 178°/178°, HDMI, VGA (D-Sub)	9 000
<b>ИТОГО</b>		48 800

При применении линейного метода амортизации сумма начисленной за один месяц амортизации в отношении объекта амортизируемого имущества определяется как произведение его первоначальной (восстановительной) стоимости и нормы амортизации, определенной для данного объекта.

При применении линейного метода норма амортизации по каждому объекту амортизируемого имущества определяется по формуле

$$K = [1/n] * 100\% , \quad (2)$$

где K – норма амортизации в процентах к первоначальной (восстановительной) стоимости объекта амортизируемого имущества;

n – срок полезного использования данного объекта амортизируемого имущества, выраженный в месяцах.

Таким образом, по формуле (2), амортизация оборудования на 5 лет составит:

$$K = [1/60] * 100\% = 1,67\% ,$$

что в денежном выражении будет означать:

$$48\,800 * 0,0167 = 814,96 \text{ руб./мес.}$$

$$\text{Агод} = 814,96 * 12 = 9780 \text{ руб.}$$

Проектная амортизация вычисляется по формуле

$$\text{Апр} = (\text{Агод} / \text{Крдг}) * \text{Кдэ}, \quad (3)$$

где Апр – проектная амортизация;

Крдг – количество рабочих дней в 2023 году;

Кдэ – количество дней эксплуатации.

В 2023 году 247 рабочих дней. С 10.04.23 по 20.05.23 при разработке веб-приложения будет задействовано 22 дня. Исходя из этих данных, получаем:

$$\text{Апр} = (9\,780 / 247) * 22 = 871 \text{ руб.}$$

$$\text{Ксвт} = 871 \text{ руб.}$$

Расчет стоимости программного обеспечения (ПО) представлен в таблице 3.

Таблица 3 – Используемое ПО

Наименование	Стоимость, руб.	Срок использования, лет
Windows 10 Professional	15 000	5
VS Code	Бесплатная лицензия	Бессрочно
Figma	Бесплатная лицензия	Бессрочно
MySQL	Бесплатная лицензия	Бессрочно

Microsoft Windows 10 Professional (x64), стоимостью 15000 рублей, обновляется раз в 5 лет. Исходя из этого, месячная стоимость использования составляет:  $15000 / 60 = 250$  руб./мес.

$$K_{\text{ипс}} = 250 * 2 = 500 \text{ руб.}$$

Затраты на инструментальные программные средства для проектирования составляют 500 рублей.

Прочие затраты равны 3% от всей суммы затрат.

$$K_{\text{проч}} = (19854 + 871 + 500) * 0,03 = 636,75 \text{ руб.}$$

Итого,

$$K_{\text{пр}} = 19\ 854 + 871 + 500 + 636,75 = 21\ 861,75 \text{ руб.}$$

### **3.2 Расчет капитальных затрат**

Капитальные затраты – это средства, используемые для приобретения или модернизации основных средств компании, такие как расходы на основные средства (PP&E). В бухгалтерском учете капитальные затраты добавляются к счету активов, тем самым увеличивая базу актива (стоимость или стоимость актива, скорректированную для целей налогообложения).

Вычисление капитальных затрат производится по формуле

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{пс}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оэ}}, \quad (4)$$

где  $K_{\text{пр}}$  – затраты на проектирование веб-приложения;

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

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

$K_{\text{пс}}$  – 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.

Рассчитаем затраты на технические средства управления ( $K_{ТС}$ ).

Разработанная система будет размещена на локальном сервере. Цена хостинга не будет входить в стоимость, поскольку она уже учитывается в Кпр.

Администрирование системы будет реализовано на компьютере, характеристики которого представлены в таблице 4.

Таблица 4 – ПК администратора

Комплектующие	Наименование	Цена, руб
Системный блок ПК DEXP Atlas H323	Intel Core i5-10400, 6x2.9 ГГц, 8 Гб DDR4, SSD 240Гб	37 500
Клавиатура проводная	DEXP K-5003BU	1 200
Мышь беспроводная	Jet.A R300G черный	1 100
23.8" Монитор MSI Pro MP241X черный	1920x1080@75 Гц, VA, 3000 : 1, 250 Кд/м <sup>2</sup> , 178°/178°, HDMI, VGA (D-Sub)	9 000
<b>ИТОГО</b>		<b>48 800</b>

Период внедрения составляет 5 дней. Предполагается покупка ПК на срок 5 лет.

Амортизация за время внедрения составит:

$$20\% / 12 / 30 * 5 * 48\ 800 = 135,56 \text{ руб.}$$

Затраты на создание линий связи локальных сетей ( $K_{ЛС}$ ) не учитываются, потому что компьютеры, используемые для разработки системы работают в автономном режиме.

Рассчитаем затраты на программные средства ( $K_{ПС}$ ).

ПК для администрирования уже имеется у заказчика. ОС и антивирусное ПО уже установлены и настроены, поэтому их амортизация не учитывается.

Для внедрения веб-приложения используется ПО, указанное ранее в затратах на проектирование. Поэтому стоимость ПО составляет 0 руб.

Рассчитаем затраты на формирование информационной базы ( $K_{ио}$ ).

В данной ИС используется бесплатная база данных MySQL. Функционал сайта предусматривает регистрацию и хранение данных пользователей.

Рассчитаем затраты на обучение персонала ( $K_{об}$ ).

Обучение администрированию будет проводиться проектировщиком в течение одного рабочего дня работы над проектом (4 часа). Стоимость обучения входит в заработную плату и составляет заработную плату разработчика за указанное время:

$$(443,26 + (443,26 * 0,3) + (443,26 * 0,3)) * 1,302 = 923,44 \text{ руб.}$$

Рассчитаем затраты на опытную эксплуатацию ( $K_{оэ}$ ).

Сопровождение ИС проектировщиком будет происходить в течение всего времени существования ИС. Данные затраты будут посчитаны далее в разделе эксплуатационных затрат. Проектировщиком является штатный сотрудник, который и будет поддерживать работы ИС.

Итоговые капитальные затраты, по формуле (3), составляют:

$$K = 21861,75 + 135,56 + 0 + 0 + 0 + 923,44 + 0 = 22\,921 \text{ руб.}$$

### 3.3 Расчет эксплуатационных затрат

Расчет эксплуатационных затрат проводится по формуле

$$C = C_{зп} + C_{ао} + C_{то} + C_{гс} + C_{ни} + C_{проч}, \quad (5)$$

где  $C_{зп}$  – зарплата персонала, работающего с веб-приложением;

$C_{ао}$  – амортизационные отчисления;

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

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

Сни – затраты на носитель информации;

Спроч – прочие затраты.

Проект не требует постоянной капитальной технической поддержки. Будет обслуживаться одним разработчиком (далее технический специалист) с минимальной тратой времени.

Обязанности технического специалиста:

– редактирование перечня базы данных новыми данными по программам обучения;

– поддержка корректной работы сайты;

– выгрузка нового контента на сайт;

– поддержка корректной работы системы обратной связи с клиентом.

Из  $K_{зп} = 9927$  руб./мес. следует, что при работе над поддержкой 4 часа в неделю,

$S_{зп} = 923,44 * 1$  (день в неделю) \* 4 (недель в месяце) \* 12 = 44325,12 руб./год.

При применении линейного метода амортизации сумма начисленной за один месяц амортизации в отношении объекта амортизируемого имущества определяется как произведение его первоначальной (восстановительной) стоимости и нормы амортизации, определенной для данного объекта.

При применении линейного метода норма амортизации по каждому объекту амортизируемого имущества определяется по формуле (2).

Таким образом, амортизация оборудования на 5 лет составит:

$$K = [1/60] * 100\% = 1,6\%,$$

что в денежном выражении будет означать:

$$A_{мес.} = 48\ 800 * 0,0167 = 814,96 \text{ руб./мес.}$$

$$C_{ао} = 814,96 * 12 = 9780 \text{ руб.}$$



Проект располагает уже имеющимися у предприятия серьезными техническими средствами такими, как сервера. Затраты по этой категории равны 0.

Запуск антивирусного программного обеспечения в компьютерных системах производится на базе MicrosoftWindows 10 Professional (x64) (встроенные антивирусные программные решения).

MicrosoftWindows 10 Professional (x64), стоимостью 15000 рублей, обновляется раз в 5 лет. Исходя из этого, месячная стоимость использования составляет:  $15000/60 = 250$  руб./мес.

$$C_{то} = 250 * 12 = 3000 \text{ руб.}$$

Расчет затрат на использование глобальных сетей (Слс).

$$C_{лс} = 0 \text{ руб.}$$

Вся информация находится на хранении у собственного локального сервера.

$$C_{ни} = 0 \text{ руб.}$$

Расчет прочих затрат (Спроч).

Прочие затраты составляют 3% от всей ранее полученной суммы, а именно:

$$C_{проч} = (44325,12 + 9780 + 3000 + 0 + 0) * 0,03 = 1713,15 \text{ руб.}$$

Итого, эксплуатационные затраты составляют:

$$C = 44325,12 + 9780 + 3000 + 0 + 0 + 1713,15 = 58\,819 \text{ руб.}$$

### 3.4 Расчет совокупной стоимости владения системой

Расчет TCO проводится по формуле

$$TCO = DE + IC_1 + IC_2, \quad (6)$$

где DE – прямые расходы;

IC<sub>1,2</sub> – косвенные расходы первой и второй групп соответственно.

IC<sub>1,2</sub> – затраты на проведение обучения и заполнение информационной базы внесены в капитальные, других косвенных затрат нет.

$$IC_{1,2} = 0$$

Расчеты прямых затрат проводятся по формуле

$$DE = DE_1 + DE_2 + DE_3 + DE_4 + DE_5 + DE_6 + DE_7 + DE_8, \quad (7)$$

где DE<sub>1</sub> – капитальные затраты;

DE<sub>2</sub> – расходы на управление ИТ;

DE<sub>3</sub> – расходы на техническую поддержку АО и ПО;

DE<sub>4</sub> – расходы на разработку прикладного ПО внутренними силами;

DE<sub>5</sub> – расходы на аутсорсинг;

DE<sub>6</sub> – командировочные расходы;

DE<sub>7</sub> – расходы на услуги связи;

DE<sub>8</sub> – другие группы расходов.

$$DE_1 = 22\,921 \text{ руб.}$$

$$DE_2 = 44\,325 \text{ руб.}$$

DE<sub>3</sub> = 0, так как техническая поддержка включена в дальнейшую заработную плату разработчика и не учитывается в рамках проекта.

$DE_4 = 0$ , так как проект не подразумевает расходы на разработку прикладного ПО внутренними силами.

$DE_5 = 0$ , так как передачи проекта аутсорсинговой компании не подразумевается. Проект поддерживается техническим специалистом.

$DE_6 = 0$ , так как нет необходимости отправлять сотрудников в командировку.

$DE_7 = 0$ , так как стоимость услуг связи уже рассчитаны в другой группе расходов и находятся в  $DE_3$  ( $Slc = 0$  руб.).

$DE_8 = 0$ , так как прочие расходы рассчитывались отдельно для каждой группы расходов.

$DE = 67\ 246$  руб.

$TCO = 67\ 246$  руб.

Таким образом, собственная разработка обходится, с экономической точки зрения, выгодней. Возможность привлечь к выполнению проекта высококвалифицированного человека за более меньший бюджет, в отличие от того, если бы он работал в статусе фрилансера. В сторонней разработке у фрилансера были бы и другие недостатки.

В первую очередь, сложно контролировать рабочий процесс: дистанционный сотрудник – это человек, которого не видишь, не понимаешь его психологического состояния и того, как быстро он выдаст результат, который нужен. Хотя современный мир позволяет достаточно просто коммуницировать – видео, аудио, письменно – но всё равно делать это дистанционно достаточно тяжело. Есть проблема защиты конфиденциальности и риск утечки некой закрытой информации. Сложно привлечь фрилансера к долгосрочному многоэтапному проекту в силу ряда причин и одна из самых простых – фрилансер может просто исчезнуть в какой-то момент времени. Бывает и такое. И в зависимости от применения систем налогообложения у предприятия могут быть проблемы с оплатой услуг фрилансера, если последний не желает декларировать свои доходы и не является налоговым резидентом.

## **Выводы по разделу «Оценка стоимости владения проектом»**

В данном разделе выполнен расчет всех капитальных (единовременных) и эксплуатационных затрат.

Определена полная совокупная стоимость программного продукта.

Также приведены преимущества разработки собственными силами, что приведет к уменьшению рисков, денежных и временных затрат.

Все полученные данные представлены в расчетах и таблицах.

## ЗАКЛЮЧЕНИЕ

В результате анализа деятельности ХТИ – филиала СФУ выявлена основная проблема: ХТИ – филиал СФУ не имеет собственного веб-приложения для продвижения курсов дополнительного профессионального образования в сети Интернет, а так же частичной автоматизации работы специалиста отдела по науке, международным связям и дополнительному образованию.

Принято решение о разработке собственного веб-приложения для ХТИ – филиала СФУ. После анализа средств разработки решено использовать для создания веб приложения следующие средства: VSCode, React, TypeScript, ReactRouter DOM, ReduxToolkit, Tailwind CSS, MySQLWorkbench.

В веб-приложении «Дополнительное профессиональное образование» предусмотрено три роли: слушатель (регистрация личного кабинета пользователя, отправка предварительной заявки для записи на выбранный курс, формирование договора), специалист (добавление нового курса на сайт), администратор (технический специалист, осуществляющий поддержку веб-приложения).

В разделе «Разработка веб-приложения "Дополнительное профессиональное образование"» продемонстрировано подробное и поэтапное создание веб-приложения выбранными средствами. Разработанное веб-приложение обладает всем заявленным функционалом.

Для данного проекта выполнено технико-экономическое обоснование разработки и перечислены преимущества создания веб-приложения собственными силами.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ХТИ – филиал / [khti.ru](https://khti.ru/): [сайт]. – URL: <https://khti.ru/> (дата обращения: 15.06.2023).
2. ХТИ – филиал / [khti.ru](https://khti.ru/documents/Институт/Положение%20об%20отделе%20НМСиДО.pdf): [сайт]. – URL: <https://khti.ru/documents/Институт/Положение%20об%20отделе%20НМСиДО.pdf> (дата обращения: 15.06.2023).
3. Студенческая библиотека онлайн / [studbooks.net](https://studbooks.net): [сайт]. – URL: [https://studbooks.net/1395595/menedzhment/aktualnost\\_povysheniya\\_kvalifikatsii\\_personala](https://studbooks.net/1395595/menedzhment/aktualnost_povysheniya_kvalifikatsii_personala) (дата обращения: 15.06.2023).
4. Crystal – 15 критериев для оценки сайта / [crystal-digital.ru](https://crystal-digital.ru): [сайт]. – URL: <https://crystal-digital.ru/blog/15-kriteriev-chtoby-oczenit-dizajn-sajta/>
5. ТУСУР / [do.tusur.ru](https://do.tusur.ru) : [сайт]. - URL: [https://do.tusur.ru/courses/virtual\\_lab](https://do.tusur.ru/courses/virtual_lab) (дата обращения: 15.06.2023).
6. Институт непрерывного образования / [ino.sfu-kras.ru](https://ino.sfu-kras.ru): [сайт]. – URL: <https://ino.sfu-kras.ru/> (дата обращения: 15.06.2023).
7. Моделирование бизнес-процессов : учебно-методическое пособие / Т. С. Карасева, Р. И. Кузьмич, И. А. Панфилов; Сибирский федеральный университет. – Красноярск : СФУ, 2021. – 51 с.
8. IDEF – Википедия / [ru.wikipedia.org](https://ru.wikipedia.org): [сайт]. – URL: <https://ru.wikipedia.org/wiki/IDEF> (дата обращения: 15.06.2023).
9. Почему VSCode популярен / [open.zeba.academy](https://open.zeba.academy) : [сайт]. – URL: <https://open.zeba.academy/pochemu-vscode-populyaren/>
10. React – Википедия / [ru.wikipedia.org](https://ru.wikipedia.org): [сайт]. – URL: <https://ru.wikipedia.org/wiki/React>(дата обращения: 15.06.2023).
11. TypeScript – Википедия / [ru.wikipedia.org](https://ru.wikipedia.org): [сайт]. – URL: <https://ru.wikipedia.org/wiki/TypeScript>(дата обращения: 15.06.2023).
12. Маршрутизация в ReactRouter / [ru.hexlet.io](https://ru.hexlet.io): [сайт]. – URL: <https://ru.hexlet.io/blog/posts/react-router-v6>(дата обращения: 15.06.2023).

13. Руководство по Redux / webformyself.com: [сайт]. – URL: [https://webformyself.com/kak-ispolzovat-redux-dlya-upravleniya-sostoyaniem/#:~:text=Redux%20Toolkit%20—](https://webformyself.com/kak-ispolzovat-redux-dlya-upravleniya-sostoyaniem/#:~:text=Redux%20Toolkit%20—%20это%20сторонняя,стандартного%20способа%20написания%20логики%20Redux(дата%20обращения:15.06.2023).)

[%20это%20сторонняя,стандартного%20способа%20написания%20логики%20Redux\(дата обращения: 15.06.2023\).](https://webformyself.com/kak-ispolzovat-redux-dlya-upravleniya-sostoyaniem/#:~:text=Redux%20Toolkit%20—%20это%20сторонняя,стандартного%20способа%20написания%20логики%20Redux(дата%20обращения:15.06.2023).)

14. TailwindCSS – Википедия / en.wikipedia.org: [сайт]. – URL: [https://en.wikipedia.org/wiki/Tailwind\\_CSS\(дата обращения: 15.06.2023\).](https://en.wikipedia.org/wiki/Tailwind_CSS(дата%20обращения:15.06.2023).)

15. Proglib – Что такое npm? / proglib.io: [сайт]. – URL: [https://proglib.io/p/chto-takoe-npm-gayd-po-node-package-manager-dlya-nachinayushchih-2020-07-21\(дата обращения: 15.06.2023\).](https://proglib.io/p/chto-takoe-npm-gayd-po-node-package-manager-dlya-nachinayushchih-2020-07-21(дата%20обращения:15.06.2023).)

16. Что такое Node.js? / Loftschool.com: [сайт]. – URL: [https://loftschool.com/blog/posts/node-java-script#node1\(дата обращения: 15.06.2023\).](https://loftschool.com/blog/posts/node-java-script#node1(дата%20обращения:15.06.2023).)

17. Что такое MySQL? / nic.ru: [сайт]. – URL: [https://www.nic.ru/help/chto-takoe-mysql\\_8510.html\(дата обращения: 15.06.2023\).](https://www.nic.ru/help/chto-takoe-mysql_8510.html(дата%20обращения:15.06.2023).)

18. Fileinfo – TSXfile / file.info.com: [сайт]. – URL: [https://fileinfo.com/extension/tsx#:~:text=Файл%20TSX-это%20файл%20TypeScript%20\(.TS\)%2C,частью%20одностраничного%20или%20мобильного%20приложения\(дата обращения: 15.06.2023\).](https://fileinfo.com/extension/tsx#:~:text=Файл%20TSX-это%20файл%20TypeScript%20(.TS)%2C,частью%20одностраничного%20или%20мобильного%20приложения(дата%20обращения:15.06.2023).)

19. Bestprogrammer – MySQLWorkbench, учебное пособие / bestprogremmer.ru: [сайт] – URL: [https://bestprogrammer.ru/baza-dannyh/uchebnoe-posobie-po-mysql-workbench-polnoe-rukovodstvo-po-instrumentu-subd#:~:text=MySQL%20Workbench%20—%20это%20кроссплатформенный,работы%20с%20вашиими%20базами%20данных\(дата обращения: 15.06.2023\).](https://bestprogrammer.ru/baza-dannyh/uchebnoe-posobie-po-mysql-workbench-polnoe-rukovodstvo-po-instrumentu-subd#:~:text=MySQL%20Workbench%20—%20это%20кроссплатформенный,работы%20с%20вашиими%20базами%20данных(дата%20обращения:15.06.2023).)

## ПРИЛОЖЕНИЕ А

### Пример заполненного договора на обучение по программе дополнительного профессионального образования

#### ДОГОВОР № ПП \_\_\_\_/23 о переподготовке кадров, повышении квалификации

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г. г. Абакан

Федеральное государственное автономное образовательное учреждение высшего образования «Сибирский федеральный университет», именуемое в дальнейшем «Исполнитель», в лице директора Хакасского технического института - филиала ФГАОУ ВО «Сибирский федеральный университет» (ХТИ – филиал СФУ) Бабушкиной Елены Анатольевны, действующего на основании Положения о Хакасском техническом институте - филиале ФГАОУ ВО «Сибирский федеральный университет», доверенности № 385 от 05.12.2022 г., Лицензии регистрационный номер № Л035-00115-24/00096964, выданной Федеральной службой по надзору в сфере образования и науки 04.07.2016 г. (бессрочно), с одной стороны, и

Гражданин РФ Киняев Федор Иванович, именуемый(ая) в дальнейшем «Заказчик», действующий на основании паспорта серия 1234 № 123456, выдан МВД по Республике Хакасия 01.01.2021, и Киняев Федор Иванович, именуемый(ая) в дальнейшем «Обучающийся», с другой стороны, заключили настоящий договор о нижеследующем:

#### 1. Предмет договора

1.1 «Исполнитель» принимает на себя обязательства по организации и проведению дополнительного профессионального образования «Обучающихся» в виде повышения квалификации по программе: **Разработка мобильных приложений**. Срок освоения программы (продолжительность обучения) 72 ак.ч.

Форма обучения: очно-заочная с применением дистанционных технологий.

Обучение осуществляется по мере формирования группы.

Период формирования групп не ограничен.

1.2 После успешного окончания обучения по программе профессиональной переподготовки на основании договора «Исполнитель» выдает «Обучающемуся» Удостоверение о повышении квалификации установленного образца при условии фактического поступления оплаты за обучение на счет «Исполнителя».

#### 2. Права и обязанности «Исполнителя»

2.1. «Исполнитель» обязан организовать и обеспечить учебный процесс в соответствии с учебным планом.

2.2. «Исполнитель» вправе самостоятельно осуществлять образовательный процесс, выбирать системы оценок, формы, порядок и периодичность промежуточной аттестации «Обучающегося», применять к нему меры поощрения и меры дисциплинарного взыскания в пределах, предусмотренных Уставом СФУ, Положением о ХТИ – филиале СФУ и другими локальными актами «Исполнителя».



2.3. Из фондов библиотеки «Исполнителя» «Обучающемуся» предоставляется имеющаяся учебная, научная, методическая и иная литература, необходимая для обучения в соответствии с программой с обязательным возвратом ее в установленные сроки.

2.4. «Исполнитель» имеет право отчислить «Обучающегося» по основаниям, предусмотренным Уставом СФУ, Положением о ХТИ – филиале СФУ, в том числе в случае нарушения установленных в настоящем договоре сроков оплаты обучения.

## **1. Права и обязанности «Заказчика»**

3.1. «Заказчик» имеет право требовать подготовки «Обучающегося» в соответствии с учебным планом.

3.2. «Заказчик» вправе требовать от «Исполнителя» предоставления информации по вопросам организации и обеспечения образовательного процесса, об успеваемости «Обучающегося».

3.3. «Заказчик» обязан своевременно и в полном объеме производить оплату обучения «Обучающегося» в соответствии с условиями настоящего договора.

3.4. «Заказчик» обязан обеспечить посещение «Обучающимся» учебных занятий в соответствии с учебным планом и расписанием учебных занятий, а также посещение «Обучающимся» консультаций, зачетов, мероприятий промежуточной и итоговой аттестации.

## **4. Права и обязанности «Обучающегося»**

4.1. «Обучающийся» имеет право получить полную и достоверную информацию об оценке своих знаний, умений и навыков, а также о критериях этой оценки.

4.2. «Обучающийся» вправе пользоваться имуществом «Исполнителя», необходимым для осуществления образовательного процесса, во время занятий, предусмотренных расписанием.

4.3. «Обучающийся» имеет право обращаться к работникам «Исполнителя» по вопросам, касающимся обучения.

4.4. «Обучающийся» обязан посещать занятия, указанные в учебном расписании.

4.5. «Обучающийся» обязан соблюдать требования Устава СФУ, Положения о ХТИ – филиале СФУ, Правил внутреннего распорядка, соблюдать учебную дисциплину и общепринятые нормы поведения, бережно относиться к имуществу «Исполнителя».

## **5. Ответственность Сторон**

5.1. За неисполнение либо ненадлежащее исполнение обязательств по договору «Исполнитель» и «Заказчик» несут ответственность, предусмотренную договором и законодательством Российской Федерации.

5.2. При обнаружении недостатка платных образовательных услуг, в том числе оказания их не в полном объеме, предусмотренном образовательными программами (частью образовательной программы), «Заказчик» вправе по своему выбору потребовать:

- а) безвозмездного оказания образовательных услуг;
- б) соразмерного уменьшения стоимости оказанных платных образовательных услуг;
- в) возмещения понесенных им расходов по устранению недостатков оказанных платных образовательных услуг своими силами или третьими лицами.

5.3. «Заказчик» вправе отказаться от исполнения договора и потребовать полного возмещения убытков, если в установленный договором срок недостатки платных образовательных услуг не устранены «Исполнителем». «Заказчик» также вправе отказаться от исполнения договора, если им обнаружен существенный недостаток оказанных платных образовательных услуг или иные существенные отступления от условий договора.

5.4. Если «Исполнитель» нарушил сроки оказания платных образовательных услуг (сроки начала и (или) окончания оказания платных образовательных услуг и (или) промежуточные сроки оказания платной образовательной услуги) либо если во время оказания платных образовательных услуг стало очевидным, что они не будут осуществлены в срок, «Заказчик» вправе по своему выбору:

а) назначить «Исполнителю» новый срок, в течение которого «Исполнитель» должен приступить к оказанию платных образовательных услуг и (или) закончить оказание платных образовательных услуг;

б) поручить оказать платные образовательные услуги третьим лицам за разумную цену и потребовать от «Исполнителя» возмещения понесенных расходов;

в) потребовать уменьшения стоимости платных образовательных услуг;

г) расторгнуть договор.

5.5. «Заказчик» вправе потребовать полного возмещения убытков, причиненных ему в связи с нарушением сроков начала и (или) окончания оказания платных образовательных услуг, а также в связи с недостатками платных образовательных услуг.

5.6. По инициативе «Исполнителя» договор может быть расторгнут в одностороннем порядке в следующем случае:

а) применение к «Обучающемуся», достигшему возраста 15 лет, отчисления как меры дисциплинарного взыскания;

б) невыполнение «Обучающимся» по программе дополнительного профессионального образования (части программы) обязанностей по добросовестному освоению такой программы (части программы) и выполнению учебного плана;

в) установление нарушения порядка приема в ХТИ – филиал СФУ, повлекшего по вине «Обучающегося» его незаконное зачисление в ХТИ – филиал СФУ;

г) просрочка оплаты стоимости платных образовательных услуг;

д) если надлежащее исполнение обязательств по оказанию платных образовательных услуг вследствие действий (бездействий) «Обучающегося».

5.7. В случае просрочки оплаты или неоплаты стоимости образовательных услуг «Исполнитель» вправе потребовать от «Заказчика» уплаты пени в размере 1/300 ключевой ставки Центрального Банка РФ за каждый день просрочки от суммы неисполненных денежных обязательств.

## **6. Условия и порядок оплаты**

6.1. «Заказчик» производит оплату в рублях согласно калькуляции расходов на обучение.

6.2. Стоимость обучения «Обучающегося» составляет 10000 (Десять тысяч) рублей. Полная стоимость обучения по настоящему договору составляет 10000 (Десять тысяч) рублей. Услуги по обучению НДС не облагаются. Услуги считаются оказанными в полном объеме и надлежащего качества после подписания сторонами Акта оказанных услуг.

6.3. «Заказчик» вносит плату, составляющую стоимость обучения (в соответствии с п. 6.2 настоящего договора) в следующем порядке:

10000 (Десять тысяч) рублей до 01.10.2023;

6.4. Оплата обучения производится путем перечисления денежных средств на соответствующий расчетный счет «Исполнителя».

Днем оплаты считается день поступления денежных средств на соответствующий расчетный счет «Исполнителя».

6.5. В случае, если группа обучающихся не будет сформирована, а «Заказчиком» перечислены денежные средства, то «Исполнитель» должен возратить указанные денежные средства в полном объеме (в количестве оплаты стоимости обучения по договору). При этом «Заказчик», не вправе предъявлять какие – либо претензии и требования к «Исполнителю» имущественного характера.

## **7. Конфиденциальность**

7.1. Для целей настоящего договора термин «Конфиденциальная информация» означает любую информацию по настоящему договору, имеющую действительную или потенциальную ценность в силу неизвестности ее третьим лицам, не предназначенную для широкого распространения и/или использования неограниченным кругом лиц, включая фактические или аналитические данные о научной, инновационной, коммерческой и иной деятельности, о технических, финансовых и других возможностях и инфраструктуре Исполнителя, сведения о технических характеристиках, охране и антитеррористической защищенности объектов Исполнителя, пропускном и внутриобъектовом режимах, обеспечении инженерно-технической, физической защиты, пожарной безопасности объектов Исполнителя, персональные данные работников и обучающихся Исполнителя.

7.2. Стороны обязуются в течение 10 лет с момента заключения настоящего договора сохранять Конфиденциальную информацию и принимать все необходимые меры для ее защиты, в том числе в случае реорганизации или ликвидации Сторон. Стороны настоящим соглашаются, что не разгласят и не допустят разглашения Конфиденциальной информации никаким третьим лицам без предварительного письменного согласия другой Стороны, кроме случаев непреднамеренного и/или вынужденного раскрытия Конфиденциальной информации по причине обстоятельств непреодолимой силы или в силу требований действующего законодательства Российской Федерации, вступивших в силу решений суда соответствующей юрисдикции либо законных требований компетентных органов государственной власти и управления, при условии, что Сторона раскроет только ту часть Конфиденциальной информации, раскрытие которой необходимо в силу применения положений действующего законодательства Российской Федерации, вступивших в законную силу решений судов соответствующей юрисдикции либо законных требований компетентных органов государственной власти и управления.

7.3. Соответствующая Сторона настоящего договора несет ответственность за действия (бездействие) своих работников и иных лиц, получивших доступ к Конфиденциальной информации.

7.4. Стороны обязуются принимать надлежащие меры, предусмотренные действующими нормативными правовыми актами РФ, по защите Конфиденциальной информации.

7.5. Для целей настоящего договора «разглашение Конфиденциальной информации» означает несанкционированные соответствующей Стороной действия другой Стороны, в результате которых какие-либо третьи лица получают доступ и возможность ознакомления с Конфиденциальной информацией. Разглашением Конфиденциальной информации признается также бездействие соответствующей Стороны, выразившееся в не обеспечении надлежащего уровня защиты Конфиденциальной информации и повлекшее получение доступа к такой информации со стороны каких-либо третьих лиц.

7.6. Соответствующая Сторона несет ответственность за убытки, которые могут быть причинены другой Стороне в результате разглашения Конфиденциальной информации или несанкционированного использования Конфиденциальной информации в нарушение условий настоящей статьи, за исключением случаев раскрытия Конфиденциальной информации, предусмотренных в настоящем разделе.

## **8. Порядок заключения, изменения и прекращения действия договора**

8.1. Настоящий договор считается заключенным с момента подписания его сторонами.

8.2. Настоящий договор действует до момента исполнения сторонами своих обязательств по настоящему договору. Договор, может быть, расторгнут досрочно по соглашению сторон, письменному заявлению «Заказчика» и иным случаям, предусмотренными ГК РФ.

8.3. Изменение условий настоящего договора оформляется Дополнительным соглашением к нему.

## **9. Заключительные положения**

9.1. Все споры, возникающие по настоящему договору, решаются путем переговоров, либо в порядке, предусмотренном действующим законодательством РФ, в том числе Постановлением Правительства РФ от 15 сентября 2020 №1441 «Об утверждении правил оказания платных образовательных услуг».

9.2 Сведения, полученные сторонами в ходе исполнения настоящего договора, являются конфиденциальной информацией. За разглашение конфиденциальной информации стороны несут ответственность в соответствии с действующим законодательством.

9.3. Договор составлен в двух экземплярах, имеющих равную юридическую силу, один из которых находится у «Заказчика», один – у «Исполнителя».

## 10. Юридические адреса и подписи сторон:

### ИСПОЛНИТЕЛЬ:

ФГАОУ ВО «Сибирский федеральный университет»  
660041, г. Красноярск, пр. Свободный, 79  
ХТИ – филиал СФУ  
655017, РХ, г. Абакан, ул. Щетинкина, 27  
тел/факс 8(3902) 22-66-60,  
**Банковские реквизиты:** УФК по  
Республике Хакасия  
(ХТИ – филиал СФУ л.сч 30806Щ02240)  
ИНН 2463011853 / КПП 190102001  
Банк: ОТДЕЛЕНИЕ-НБ РЕСПУБЛИКА  
ХАКАСИЯ БАНКА РОССИИ//УФК по  
Республике Хакасия г. Абакан  
Счет 03214643000000018000  
Кор.счет 40102810845370000082  
БИК 019514901  
ОКТМО 95701000 КБК 0

Директор ХТИ – филиала СФУ

\_\_\_\_\_  
(подпись) Е. А. Бабушкина  
(ФИО)  
МП

Согласовано:

Начальник отдела по НМС и ДО  
\_\_\_\_\_  
Т. Н. Плотникова

Заведующий ПС  
\_\_\_\_\_  
Т. И. Потылицына

### ЗАКАЗЧИК:

ФИО Киняев Федор Иванович  
Дата рождения: 01.01.2001  
Паспорт 1234 № 123456, выдан МВД по  
Республике Хакасия, дата выдачи 01.01.2021  
Адрес: г.Абакан, ул. Пушкина 123  
СНИЛС 194-123-123 74  
ИНН 1234567890123456  
Тел. 8(983)111-22-33  
E-mail: KynT111@mail.ru

\_\_\_\_\_  
(подпись) (ФИО)

### ОБУЧАЮЩИЙСЯ:

ФИО Киняев Федор Иванович  
Дата рождения: 01.01.2001  
Паспорт 1234 № 123456, выдан МВД по  
Республике Хакасия, дата выдачи 01.01.2021  
Адрес: г.Абакан, ул. Пушкина 123  
СНИЛС 194-123-123 74  
ИНН 1234567890123456  
Тел. 8(983)111-22-33  
E-mail: KynT111@mail.ru

\_\_\_\_\_  
(подпись) (ФИО)

Выпускная квалификационная работа выполнена мной самостоятельно.  
Использованные 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.

«\_\_\_\_\_» \_\_\_\_\_ 2023 г.

\_\_\_\_\_ Попов Данил Андреевич  
подпись

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

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

Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
О. В. Папина  
подпись  
« 19 » июня 2023 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.03 Прикладная информатика

Разработка веб-приложения «Дополнительное профессиональное образование»  
для ХТИ – филиала СФУ

Руководитель М. А. Буреева 19.06.23г. доцент, канд. физ.-мат. наук М. А. Буреева  
подпись, дата

Выпускник Д. А. Попов 19.06.23г. Д. А. Попов  
подпись, дата

Консультанты  
по разделам:

Экономический М. А. Буреева 19.06.23г. М. А. Буреева  
подпись, дата

Нормоконтролер А. Н. Кадычегова 19.06.23г. А. Н. Кадычегова  
подпись, дата

Абакан 2023