

Федеральное государственное автономное
образовательное учреждение
высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

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

Кафедра прикладной информатики, математики и естественно-научных
дисциплин

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Е. Н. Скуратенко
подпись

« _____ » _____ 2021 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03 Прикладная информатика

Разработка информационной системы онлайн экскурсий по объектам
культурного наследия Республики Хакасия

Руководитель _____ зав.кафедрой, канд. техн. наук Е.Н. Скуратенко
подпись, дата

Выпускник _____ Н.О. Ленцов
подпись, дата

Консультанты
по разделам:

Экономический _____ Е. Н. Скуратенко
подпись, дата

Нормоконтролер _____ В. И. Кокова
подпись, дата

Абакан 2021

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

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

Кафедра прикладной информатики, математики и естественно-научных
дисциплин

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Е. Н. Скуратенко
подпись

« _____ » _____ 2021 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Ленцову Николаю Олеговичу

Группа ХБ 17-03

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.

Утверждена приказом по институту № 222 от 08.04.2021 г.

Руководитель ВКР: Е.Н. Скуратенко, зав. кафедрой, канд. техн. наук, ХТИ – филиал СФУ

Исходные данные для ВКР: информация о деятельности Министерства культуры Республики Хакасия, информация о локальном сервере.

Перечень разделов ВКР:

1. Анализ предметной области.
2. Разработка веб-приложения для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.
3. Оценка экономической эффективности информационной системы.

Перечень графического материала: нет

Руководитель ВКР

подпись

Е. Н. Скуратенко

Задание принял к исполнению

подпись

Н.О. Ленцов

«08» апреля 2021 г.

РЕФЕРАТ

Выпускная квалификационная работа по теме: «Разработка информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия» содержит 71 страницу текстового документа, 30 рисунков, 8 таблиц, 18 формул, 1 приложение, 20 использованных источников.

ИНФОРМАЦИОННАЯ СИСТЕМА, ВЕБ-ПРИЛОЖЕНИЕ, FIREBASEFLASK, PYTHON, РЕПОЗИТОРИЙ, ВИДЕОСВЯЗЬ, ЭКОНОМИЧЕСКАЯ ЭФФЕКТИВНОСТЬ, КАПИТАЛЬНЫЕ ЗАТРАТЫ, ЭКСПЛУАТАЦИОННЫЕ ЗАТРАТЫ.

Цель работы: Разработать информационную систему онлайн экскурсий по объектам культурного наследия Республики Хакасия.

Задачи:

- изучить предметную область;
- провести анализ деятельности Министерства культуры Республики Хакасия;
- определить цель и задачи проектирования ИС;
- определить информационное обеспечение для разработки ИС;
- спроектировать информационную систему;
- разработать информационную систему;
- произвести расчеты экономической эффективности проекта.

В результате было разработано веб-приложение для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.

SUMMARY

The theme of the graduation thesis is «IT System Development for Online Excursions to Cultural Heritage Sites of the Republic of Khakassia». It contains 71 pages of a text document, 30 figures, 8 tables, 18 formulae, 1 appendix, 20 reference items.

IT SYSTEM, WEB APPLICATION, FIREBASE FLASK, PYTHON, REPOSITORY, VIDEO COMMUNICATION, COST-EFFECTIVENESS, CAPITAL COSTS, OPERATING COSTS.

The purpose of the research: to develop a web application for IT system to provide remote excursions on cultural heritage sites of the Republic of Khakassia.

Objectives:

- to study the subject matter;
- to analyze the workflow of the Ministry of Culture of the Republic of Khakassia;
- to define the purpose and objectives of IT system design;
- to determine information support for the development of IT system;
- to design an IT system;
- to develop an IT system;
- to calculate the economic efficiency of the project.

Results: the web application has been developed for organizing remote excursions on the cultural heritage sites of the Republic of Khakassia.

English language supervisor

Chezybaeva N.V.

signature, date

full name

СОДЕРЖАНИЕ

Введение	8
1 Анализ предметной области для создания информационной системы онлайн экскурсий на объектах культурного наследия Республики Хакасия..	10
1.1 Организационная характеристика предметной области	10
1.2 Построение структурной схемы Министерства культуры Республики Хакасия	11
1.3 Бизнес-процессы предметной области.....	13
1.4 Характеристика IT-инфраструктуры Министерства культуры Республики Хакасия.....	15
1.5 Постановка цели и задач проектирования	16
1.6 Анализ литературы и других источников информации по функционированию систем автоматизации экскурсионных услуг	19
1.7 Обоснование и выбор средств проектных решений.....	22
1.8 Выводы по разделу «Анализ предметной области для создания информационной системы онлайн удаленных экскурсий на объектах культурного наследия Республики Хакасия».....	24
2 Разработка веб-приложения для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.....	25
2.1 Подготовка рабочей среды	25
2.2 Разработка проекта	27
2.3 Описание информационной системы	38
2.4 Выводы по разделу «Разработка веб-приложения для информационной системы онлайн экскурсий на объектах культурного наследия республики Хакасия».....	44
3 Оценка экономической эффективности разработки информационной системы онлайн экскурсий на объектах культурного наследия Республики Хакасия	45

3.1	Расчет затрат реализации проекта информационной системы Республики Хакасия.....	45
3.1.1	Капитальные затраты	45
3.1.2	Эксплуатационные затраты	53
3.2	Расчет по методике ТСО	57
3.3	Оценка рисков при реализации проекта создания ИС	59
3.3.1	Идентификация рисков	59
3.3.2	Анализ рисков	60
3.3.3	Планирование риск-менеджмента и разрешение рисков	61
3.4	Анализ рынка продуктов – аналогов. Установление стоимости программного продукта	62
3.5	Источник получения дохода	64
3.6	Экономическая эффективность реализации проекта	65
3.7	Выводы по экономическому разделу	68
	Заключение	69
	Список использованных источников	70
	ПРИЛОЖЕНИЕ А	72

ВВЕДЕНИЕ

Одной из приоритетных задач современной России является цифровизация процессов производственной и общественной сферы. Перспективное функционирование различных организаций, особенно государственных, связано с использованием существующего и созданием своего программного обеспечения (ПО) для обработки данных.

Министерство культуры Республики Хакасия (далее – Министерство) осуществляет свою деятельность, связанную с контролем осуществления надзорной деятельности в области экскурсионных услуг на объектах культурного наследия Республики Хакасия путем непосредственного контроля подведомственных учреждений, оказывающих услуги на объектах культурного наследия Республики Хакасия.

По заказу Министерства культуры Республики Хакасия была выполнена работа по созданию информационной системы для организации оказания удаленных экскурсий на объектах культурного наследия республики Хакасия

На данный момент у Министерства культуры Республики Хакасия нет информационной системы, которая позволила бы получить доступ к удаленным экскурсиям на объектах культурного наследия и вести мониторинг и отчетность по данной области. Данная информационная система нужна для того, чтобы связать потоки получателей услуг, лиц, оказывающих данные услуги и сотрудников Министерства. Таким образом, актуальность выбранной темы связана с необходимостью разработки информационной системы для организации удаленных экскурсий на объектах культурного наследия, формирование отчетности и совмещение потоков данных пользователей, которые получают и предоставляют услугу.

Целью выпускной квалификационной работы является создание веб-приложения для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.

Работа по созданию информационной системы велась совместно со студентом группы 57-1 Аболенцевым Андреем Евгеньевичем. С его стороны было разработано мобильное приложение для данной системы.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Осуществить поиск информации с целью определения технических характеристик проекта разработки.
2. Сформулировать цель и задачи разработки проекта.
3. Обосновать выбор средств проектных решений.
4. Провести моделирование проекта веб-приложения.
5. Разработать веб-приложение.
6. Оценить экономическую эффективность внедрения разработанного веб приложения для информационной системы.

1 Анализ предметной области для создания информационной системы для организации онлайн экскурсий на объектах культурного наследия Республики Хакасия

1.1 Организационная характеристика предметной области

Экскурсионные услуги в Министерстве культуры оказывают учреждения, занимающиеся музейной деятельностью.

В непосредственном подчинении у Министерства культуры находятся следующие музеи с объектами культурного наследия:

1. Хакасский национальный музей-заповедник “Казановка” – один из самых удивительных и без сомнения самый крупный музей под открытым небом в Республике Хакасия расположен в Аскизском районе, в 5-и километрах от деревни Казановка, в предгорьях Кузнецкого Алатау. Площадь музея-заповедника составляет более 18 тыс.га и содержит огромное количество уникальных образцов историко-культурного наследия хакасского народа.

2. Аскизский краеведческий музей им.Н.Ф. Катанова – основным направлением деятельности музея является краеведение. В музее функционируют следующие экспозиции: «Этнография хакасского народа», «Ученый Николай Фёдорович Катанов», «Аскизский район», «Этих дней не смолкнет слава...», «Пётр Иванович Кузнецов в селе Аскиз». Эти экспозиции расположены на втором этаже здания музея. На первом этаже в выставочном зале организуют выставки разной тематики.

3. Хакасский национальный краеведческий музей им.Л.Р. Кызласова – основан на общественных началах в 1929 г. при окружном отделе народного образования, статус государственного получил на основании Постановления №16 от 29 июля 1931 г. Президиума Хакасского областного исполнительного комитета «Об организации областного музея».

4. Музей «Древние курганы Салбыкской степи» – на территории Усть-Абаканского района Республики Хакасия расположен музей «Древние курганы Салбыкской степи», созданный в 2007 году. Музейный комплекс под открытым небом включает в себя уникальные памятники историко-культурного наследия, самые известные из которых Большой Салбыкский курган, курган «Барсучий лог» и менгир «Ворота в Салбыкскую долину».

Данные учреждения занимаются организацией и проведением информационных экскурсий по объектам культурного наследия Республики Хакасия. Для проведения экскурсий приходится привлекать специалистов в этой области и осуществлять непосредственный выезд на территории, что влечет затраты времени и денежных средств на транспорт. К сожалению, в виду того, что данные гиды занимаются и непосредственной работой в своих учреждениях, иногда выезд на территорию проведения экскурсии в назначенное время туристом может не состояться по независящим от гида причинам. Возникает потребность в автоматизации процесса проведения экскурсий и перевода их в дистанционную форму для увеличения возможностей у гидов проведения экскурсии без выезда на территорию и без отрыва от места основной работы, и появления возможности у «стихийных туристов» воспользоваться экскурсионными услугами на территории Республики Хакасия.

1.2 Построение структурной схемы Министерства культуры Республики Хакасия

На рисунке 1 представлена схема Министерства культуры Республики Хакасия.

Структура Министерства культуры Республики Хакасия



Рисунок 1 – Структурная схема Министерства культуры РХ

В Министерстве функционируют 4 отдела:

1. Координационно-аналитический отдел занимается мониторингом, контролем строительства и капитальных ремонтов объектов культуры, координацией деятельностью Министерства культуры по организации независимой оценки качества оказания услуг учреждениями культуры Республики Хакасия, размещением конкурсной документации государственных заказов Министерства культуры, отслеживание и контроль за ведением конкурсных процедур, информационной деятельностью и медиапланированием, взаимодействием со средствами массовой информации, обеспечением документооборота, материально-техническим и программным обеспечением деятельности Министерства культуры.

2. Отдел экономики и финансового обеспечения отвечает за экономическое обеспечение деятельности Министерства и за все связанные с этим процессами.

3. Отдел современного искусства занимается проведением от имени и через Министерство культуры Республики Хакасия государственной политики в сфере клубного дела, профессионального и самодеятельного искусства, в области библиотечного дела, системы художественного образования сферы культуры и искусства на территории Республики Хакасия.

4. Отдел культурного наследия и архивов в пределах своей компетенции осуществляет курирование и координацию деятельности

республиканских государственных учреждений культуры, оказывающих государственные услуги в области библиотечного, архивного, музейного дела, сохранения нематериального культурного наследия и литературного процесса.

Таким образом, если внедрить автоматизированную систему для проведения экскурсий по памятникам культурного наследия, то возможно увеличение потока туристов, количества оказанных экскурсионных услуг и упрощение процесса сбора статистики с подведомственных учреждений.

1.3 Бизнес-процессы предметной области

Министерство культуры Республики Хакасия является исполнительным органом государственной власти Республики Хакасия, осуществляющим функции в области культуры и искусства, архивного дела, кино [1].

Главной целью Министерства является создание условий для сохранения, развития и совершенствования профессионального и самодеятельного искусства, кино, народного творчества, эстетического воспитания и художественного образования, прежде всего, посредством поддержки и развития организаций, осуществляющих образовательную деятельность по образовательным программам в области культуры и искусства, а также сохранения бесплатности для населения основных услуг общедоступных библиотек. Руководящий аппарат Министерства представлен Министром культуры, помощником Министра, тремя заместителями и ведущим советником.

Министерство напрямую подчиняется Правительству Республики Хакасия и непосредственно Главе Республики Хакасия – Председателю Правительства Республики Хакасия.

При разработке автоматизированной системы учитывалось то, что данный продукт является новым для области культуры, и полных аналогов ее нет, на данный момент имеются только похожие по функционалу продукты, но они не имеют в своем арсенале все функции, которые необходимо решать в процессах оказания экскурсионных услуг на объектах культурного наследия Республики Хакасия. В своем классическом варианте процесс оказания услуги по организации и проведению экскурсионного тура представляет собой процесс сбора заявок, обработки заявки и оказания непосредственно самой услуги. Представленная модель на рисунке 2 отражает процесс оказания услуги до автоматизации (модель черного ящика).

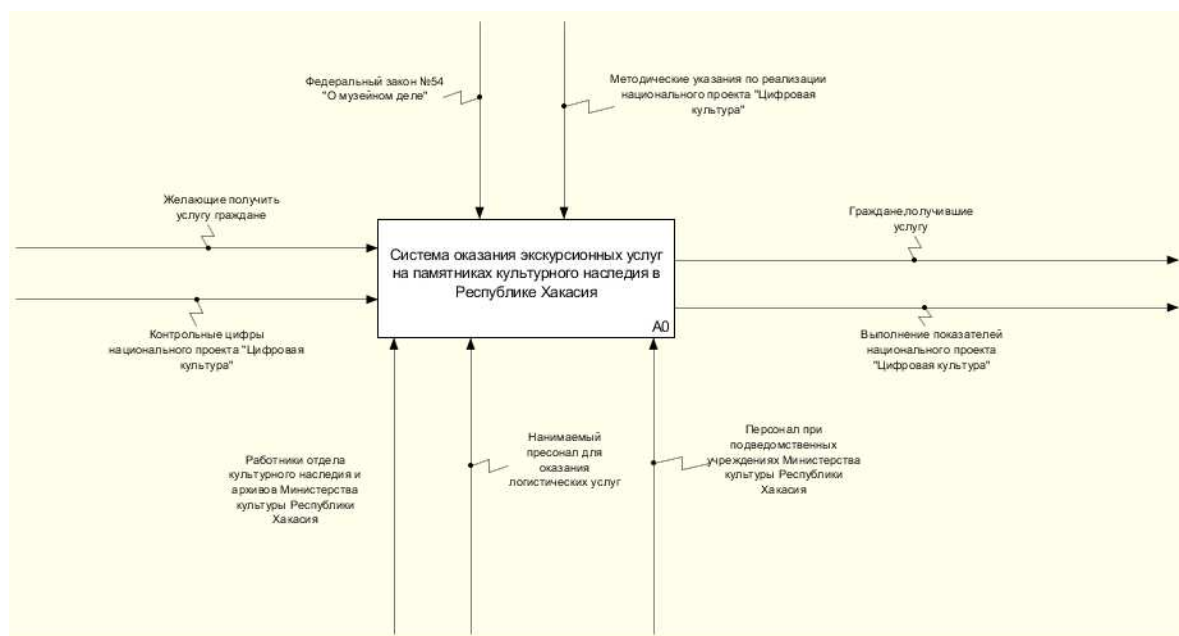


Рисунок 2 – Модель AS-IS (как было) процесса оказания услуг экскурсионного направления

Для более подробного изучения существующего процесса была спроектирована диаграмма IDEF0, отображающая внутренние связи между процессами в системе оказания услуги (рисунок 3).

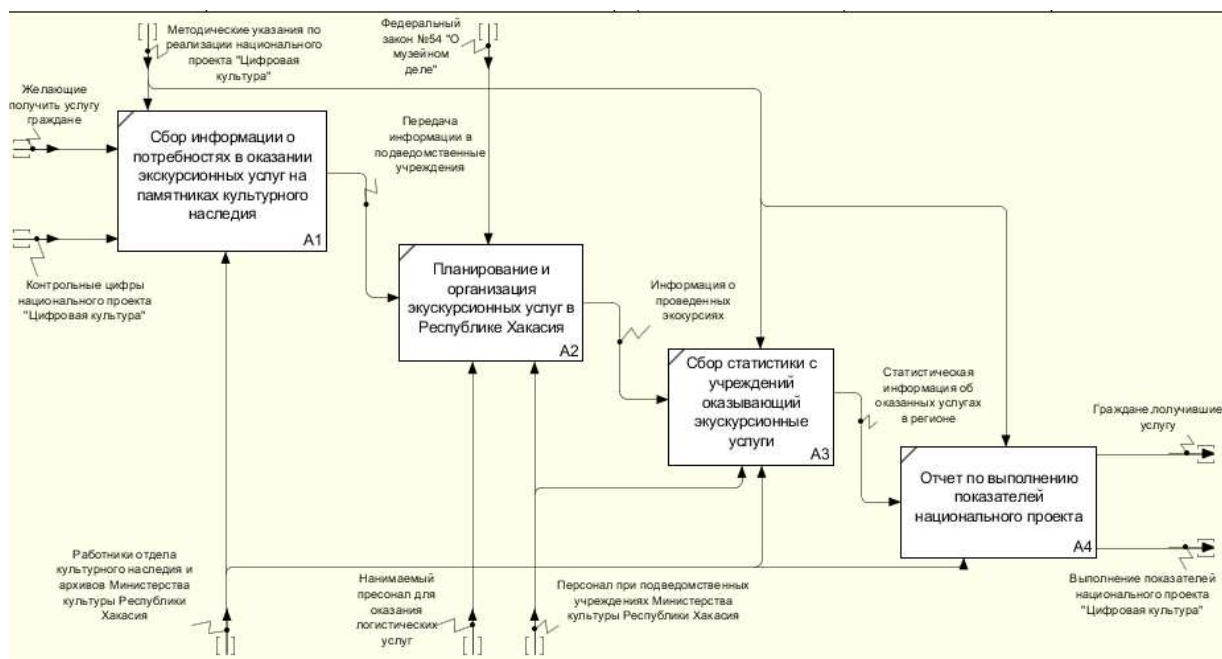


Рисунок 3 – Декомпозиция модели AS-IS до процесса автоматизации оказания услуг

Как видно из данной модели, до процесса автоматизации необходимым механизмом для оказания услуг является нанимаемый персонал для оказания логистических услуг и отсутствует доступ к информации в режиме онлайн для получения данных мониторинга.

По итогам было выполнено проектирование моделей AS-IS до внедрения процесса автоматизации и проанализированы основные моменты, которые будут улучшены, благодаря проведению автоматизации.

1.4 Характеристика IT-инфраструктуры Министерства культуры Республики Хакасия

В Министерстве установлены компьютеры, предназначенные, в основном, для работы с документами, отчетами. Количество рабочих станций составляет 25 единиц. Подменный фонд, который готов немедленно приступить к работе – это 4 единицы полностью укомплектованных станций.

1 компьютер с защищенной сетью с помощью ПО SecretNet, для работы с документами с грифом «Секретно».

В Министерстве используются в основном программы, необходимые для ведения документооборота, а именно:

- MicrosoftOffice.
- Система электронного документооборота «Дело» (версии «толстого» и «тонкого» клиента).
- Программа «ПУ-6» – программа обмена данными с ПФ РФ.
- Программный комплекс «Госслужба».
- ФГИС «ПК ФК», «ПК АФ».
- Системы Барс «Web-свод».
- «Бюджет Смарт».

Для защиты информации в Министерстве применяются системы «SecretNet» и «VipNet», служащие сетевыми экранами и позволяющие подключаться к сетям Правительства Республики Хакасия. Для работы с электронными подписями установлен пакет «КриптоПРО». К другим системам, которые необходимы в работе, имеется web-доступ.

По итогам анализа компьютерного парка предприятия было выяснено, что в Министерстве культуры отсутствует компьютер, который может стать сервером для разрабатываемой информационной системы, ввиду низких технических характеристик.

1.5 Постановка цели и задач проектирования

На данный момент у Министерства культуры Республики Хакасия нет информационной системы, которая позволила бы организовать доступ к удаленным экскурсиям на объектах культурного наследия и вести мониторинг и отчетность по данной области. Данная информационная система нужна для того, чтобы связать потоки получателей услуг, лиц, оказывающих данные услуги и сотрудников Министерства. Таким образом,

актуальность выбранной темы связана с необходимостью разработки информационной системы для организации оказания удаленных экскурсий на объектах культурного наследия, формирование отчетности и совмещение потоков данных пользователей, которые получают и предоставляют услугу.

Целью работы является разработка информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.

Для постановки задач при разработке веб-приложения как части информационной системы, первоначальным этапом является сбор и систематизация требований к программному продукту. Для составления требований к веб-приложению и информационной системе в целом применяются такие методы, как интервьюирование сотрудников и будущих пользователей системы, получение требований от непосредственного заказчика системы и составление требований, исходя из специфики разрабатываемого веб приложения.

Данные требования были собраны, проанализированы и систематизированы, полученный результат представлен в таблице 1.

Таблица 1 – Требования к информационной системе

№ п/п	Требование	Группа требований по ГОСТ	Приоритет требований
1	Предоставление канала связи с пользователем	требования к системе в целом	Необходимо
2	Мобильное приложение	требования к системе в целом	Необходимо
3	Десктопное приложение	требования к системе в целом	Необходимо
4	Возможность администрирования контента	требования к функциям (задачам), выполняемым системой	Необходимо
5	Фирменный стиль Министерства культуры РХ	требования к видам обеспечения.	Дополнительно
6	Наличие медиаконтента	требования к видам обеспечения.	Желательно
7	Наличие авторизации	требования к системе в целом	Необходимо
8	Разграничение прав доступа	требования к системе в целом	Необходимо

Окончание таблицы 1

9	Подсоединение через общую сеть к серверу Министерства культуры РХ	требования к системе в целом	Необходимо
10	Возможность подключения к системе через мобильное приложение	требования к функциям (задачам), выполняемым системой	Необходимо
11	Возможность подключения к системе через десктопное приложение	требования к функциям (задачам), выполняемым системой	Необходимо
12	Возможность у администратора добавления, удаления и изменения контента системы	требования к функциям (задачам), выполняемым системой	Необходимо
13	Применение в мобильном приложении фирменных шрифтов и оформления МК РХ	требования к видам обеспечения.	Дополнительно
14	Предоставление информации в приложении в форматах аудио, видео и текст	требования к видам обеспечения.	Желательно
15	Наличие регистрации и авторизации пользователей	требования к функциям (задачам), выполняемым системой	Необходимо
16	Предоставление различного набора прав доступа в зависимости от типа учетной записи	требования к системе в целом	Необходимо
17	Возможность у системного администратора подключение к тонкому и толстому клиенту с полным уровнем доступа	требования к функциям (задачам), выполняемым системой	Необходимо
18	Возможность у системного администратора управления сервером	требования к функциям (задачам), выполняемым системой	Необходимо
19	Возможность у системного администратора регистрации с назначением прав доступа	требования к функциям (задачам), выполняемым системой	Необходимо
20	Поддержка у мобильного приложения Android версии не ниже 6.0	требования к видам обеспечения.	Необходимо
21	Поддержка у десктопного приложения системы Windows 7,8,10	требования к видам обеспечения.	Необходимо

В результате исследования предметной области были выявлены основные особенности объекта автоматизации, а именно:

- Выявлены предпосылки для создания автоматизированной системы.
- Проанализирована деятельность в прикладной сфере, требующей автоматизации.
- Проведен анализ существующего бизнес-процесса, как объекта автоматизации.
- Определены требования к системе, сформулированы цели и задачи разработки информационной системы.
- Оформлено техническое задание на создание автоматизированной системы организации оказания удаленных экскурсий на объектах культурного наследия Республики Хакасия.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Осуществить поиск информации с целью определения технических характеристик проекта разработки.
2. Сформулировать цель и задачи разработки проекта.
3. Обосновать выбор средств проектных решений.
4. Провести моделирование проекта веб-приложения.
5. Разработать прототип веб приложения (главной страницы).
6. Оценить экономическую эффективность внедрения разработанной информационной системы.

1.6 Анализ литературы и других источников информации по функционированию систем автоматизации экскурсионных услуг

На рынке программного обеспечения представлены различные продукты для проведения виртуальных экскурсий и подробнее остановимся на наиболее популярных:

1) Tour Builder, разработанный Google, позволяет самостоятельно создавать виртуальные экскурсии. Инструмент работает на основе сервиса Google Earth. Для создания тура нужно выбрать маршрут, добавить описание, фотографии и видео. Например, можно рассказать о какой-то стране, поделиться впечатлениями о поездке на каникулах, рассказать о путешествии знаменитых людей и многое другое.

Особенности:

- англоязычный;
- необходимо иметь аккаунт Google;
- бесплатный;
- нет возможность обратной связи;
- имеется возможность поделиться созданной работой через ссылку или скачать KML файл и встроить в Google Earth;
- групповая работа на сервисе не предусмотрена.

2) Geteach –сервис создан на основе приложения Google. Автором приложения является учитель из г. Остина (штат Техас, США) Джош Уильямс. Для создания маршрута необходимо выбрать на карте объект, внести его описание и вставить фото. Сервис дает возможность сопоставить две карты, использовать чертежные и измерительные инструменты, метки, технологии Google Street, возможность работать со слоями, можно вставить подобранные карты и фотографии.

Особенности:

- англоязычный;
- необходимо иметь аккаунт Google;
- бесплатный;
- нет возможности обратной связи;

- имеется возможность поделиться созданной работой в Google Earth.

- групповая работа на сервисе не предусмотрена.

3) Tour Creator, сервис создан на основе Google и предназначен для создания виртуальных туров на основе карт Google. Сервис позволяет за короткое время создать виртуальный тур с использованием фотографий Google Street Map или собственные 360-градусные фотографии. Имеется возможность для создания тура с использованием картонного VR-шлема Google Cardboard и сервиса Expeditions. Созданным туром можно поделиться для всеобщего обозрения, а также отправить ссылку через Google Classroom, Google Meet или Gmail.

Для начала работы в сервисе необходим аккаунт Google. Перед началом работы по созданию экскурсии необходимо дать описание тура и вставить фото обложки. Сервис позволяет использовать технологии Google Street, имеется возможность добавить на изображение описание отдельных элементов здания или сооружения.

Особенности:

- англоязычный;

- бесплатный;

- нет возможности обратной связи;

- имеется возможность поделиться созданной работой через ссылку, вставить в блог или сайт, опубликовать в социальных сетях.

- групповая работа на сервисе не предусмотрена.

4) TimelineJS manual – генератор интерактивных таймлайнов для web. Сразу после начала создания своего проекта, сервис предложит пользователям подробную инструкцию со скриншотами по работе с инструментарием сервиса. Есть и видеоролик, который покажет все действия пользователя.

После перехода на сайт выбирается вкладка «Инструменты для рассказывания историй», затем вкладку StoriMap и начинается создание тура.

К каждому месту на карте можно добавить описание и фотографии. После создания тура можно поделиться созданной работой через ссылку или вставить на сайт или блог.

Особенности:

- сервис англоязычный;
- необходимо иметь аккаунт Goggle;
- бесплатный;
- нет возможности обратной связи;
- имеется возможность поделиться созданной работой через ссылку;
- групповая работа на сервисе не предусмотрена.

Как видно из информации, представленной выше, данные продукты не подходят к специфике нашего региона, они не располагают возможностью диалога с экскурсоводом, не предусматривают возможности оплаты за проведенную экскурсию. При использовании сторонних сервисов для видеосвязи туриста и гида возникают различные трудности, в первую очередь связанные с неудобством использования нескольких программных продуктов одновременно, неудобством оплаты и просмотра спроса на определенную экскурсию. Поэтому создание автоматизированной системы для организации удаленных экскурсий по объектам культурного наследия Республики Хакасия необходимо будет произвести собственными силами.

1.7Обоснование и выбор средств проектных решений

Для разработки и реализации системы были выбраны варианты мобильного и веб - приложения.

Реализация веб-приложения предполагает использование системы на персональных компьютерах, поэтому желательно, чтобы данная система работала из браузера, уже установленного на компьютере.

Для такой разработки лучше всего подходит язык программирования Python, а именно его реализация через фреймворк Flask, в котором разрабатываются веб-приложения.

Основные преимущества фреймворка Flask:

- ORM;
- миграции базы данных;
- аутентификация пользователя;
- панель администратора;
- формы;

Для реализации базы данных используется облачная база данных Firebase, поскольку система будет работать в двух вариациях (мобильное и веб-приложения), организации хранения данных и подключения базы данных должны быть вынесены за пределы серверного оборудования Министерства культуры.

Технология для передачи аудио и видео сигнала была выбрана webrtc, как наиболее распространенная и не требующая настроек со стороны пользователей, поскольку использует собственную пакетную передачу информации напрямую без серверного шифрования.

Преимущества webrtc:

- Высокий уровень безопасности: все соединения защищены и зашифрованы согласно протоколам TLS и SRTP.
- Есть встроенный механизм захвата контента, например, рабочего стола.
- Возможность реализации любого интерфейса управления на основе HTML5 и Python.
- Возможность интеграции интерфейса с любыми back-end системами с помощью WebSockets.
- Проект с открытым исходным кодом – можно внедрить в свой продукт или сервис.

– Настоящая кросс-платформенность: одно и то же WebRTC приложение будет одинаково хорошо работать на любой операционной системе, десктопной или мобильной, при условии, что браузер поддерживает WebRTC.

1.8 Выводы по разделу «Анализ предметной области для создания информационной системы онлайн экскурсий на объектах культурного наследия Республики Хакасия»

В ходе выполнения аналитического раздела были описаны следующие подразделы:

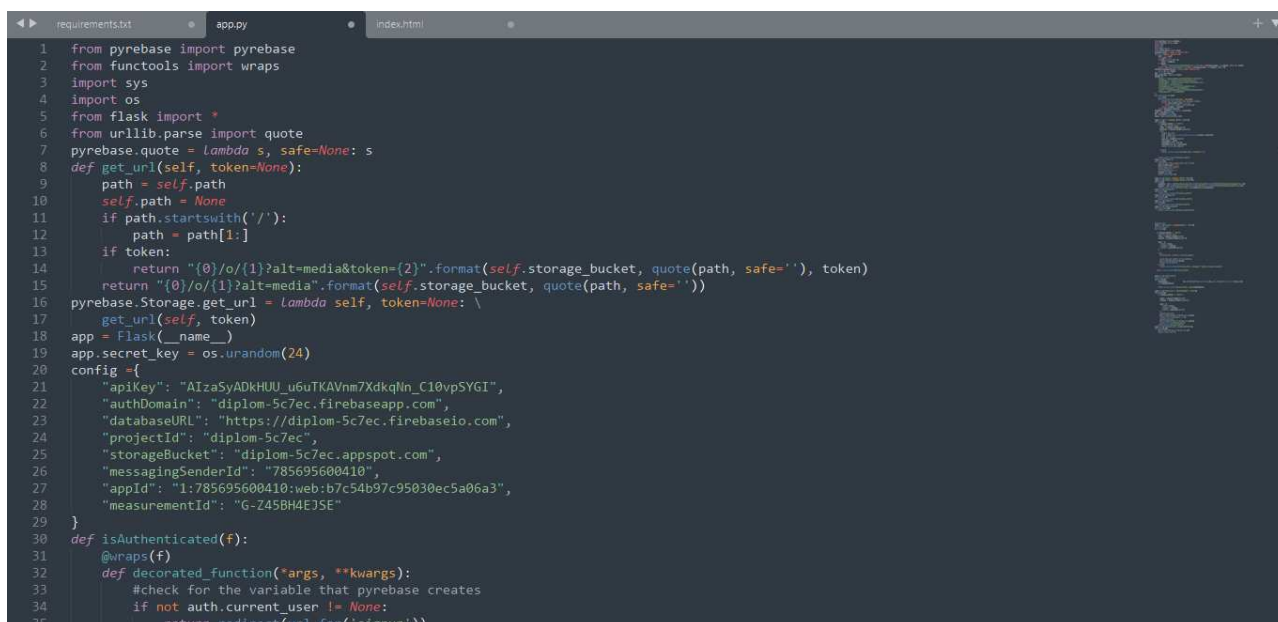
- Организационная характеристика предметной области.
- Построение структурной схемы Министерства культуры Республики Хакасия.
- Бизнес-процессы предметной области.
- Характеристика IT-инфраструктуры Министерства культуры Республики Хакасия.
- Постановка цели и задач проектирования.
- Анализ литературы и других источников информации по функционированию подобных систем.
- Обоснование и выбор средств проектных решений.

На основании вышеизложенного можно утверждать, что разработка информационной системы онлайн экскурсий на объектах культурного наследия Республики Хакасия может быть продолжена и реализована.

2 Разработка веб-приложения для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия

2.1 Подготовка рабочей среды

Для создания информационной системы использовался редактор исходного кода Sublime, распространяющийся по бесплатной лицензии. Текстовый редактор показан на рисунке 4.



```
1 from pyrebase import pyrebase
2 from functools import wraps
3 import sys
4 import os
5 from flask import *
6 from urllib.parse import quote
7 pyrebase.quote = lambda s, safe=None: s
8 def get_url(self, token=None):
9     path = self.path
10    self.path = None
11    if path.startswith('/'):
12        path = path[1:]
13    if token:
14        return "{0}/o/{1}?alt=media&token={2}".format(self.storage_bucket, quote(path, safe=''), token)
15    return "{0}/o/{1}?alt=media".format(self.storage_bucket, quote(path, safe=''))
16 pyrebase.Storage.get_url = lambda self, token=None: \
17     get_url(self, token)
18 app = Flask(__name__)
19 app.secret_key = os.urandom(24)
20 config = {
21     "apiKey": "AIzaSyADkHUU_u6uTKAVnm7XdkqMn_C10vpSYGI",
22     "authDomain": "diplom-5c7ec.firebaseio.com",
23     "databaseURL": "https://diplom-5c7ec.firebaseio.com",
24     "projectId": "diplom-5c7ec",
25     "storageBucket": "diplom-5c7ec.appspot.com",
26     "messagingSenderId": "785695600410",
27     "appId": "1:785695600410:web:b7c54b97c95030ec5a06a3",
28     "measurementId": "G-2458H4EJ5E"
29 }
30 def isAuthenticated(f):
31     @wraps(f)
32     def decorated_function(*args, **kwargs):
33         #check for the variable that pyrebase creates
34         if not auth.current_user != None:
35             return redirect(url_for('signup'))
```

Рисунок 4 – Главный экран Sublime

Для корректной работы с веб-приложением на компьютере, где происходит разработка, установлены следующие модули и программы:

- Python: программный комплекс для разработки приложений на данном языке программирования.
- Pip3: Расширение, которое необходимо для установки и загрузки библиотек для среды разработки python.
- Flask: Один из самых популярных фреймворков и инструментов для веб-разработки на языке программирования python.

- Pyrebase4: модуль для работы с облачными базами данных.
- Ngrok: модуль для отладки и тестирования веб-приложений без публикации их на своем сервере.

```
bcrypt==3.1.7
firebase-admin==5.0.0
google-api-core==1.26.3
google-api-python-client==2.3.0
google-auth==1.30.0
google-auth-httpplib2==0.1.0
google-cloud-core==1.6.0
google-cloud-firestore==2.1.1
google-cloud-storage==1.38.0
google-crc32c==1.1.2
google-resumable-media==1.2.0
googleapis-common-protos==1.53.0
grpcio==1.37.1
httpplib2==0.14.0
oauth2client==3.0.0
oauthlib==3.1.0
SecretStorage==2.3.1
```

Рисунок 5 – Список используемых пакетов

Для создания информационной системы использовались пакеты, показанные на рисунке 5:

- Bcrypt: используется для шифрования пароля при сохранении пользователя и для проверки совпадения пароля при авторизации пользователя.
- Firebaseadmin: используется для администрирования базы данных.
- Google-сервисы: используются для работы с файлами, получаемыми и отправляемые в систему базы данных реального времени.
- Httpplib2: используется для работы с файлами httpв среде разработки python

- OAuth2client и oauthlib: библиотеки для обработки запросов авторизации и аутентификации пользователей приложения
- Sercetstorage: библиотека для работы с облачными хранилищами и правильно зашифрованной передачей файлов между сервером и клиентом.

Таким образом, была подготовлена рабочая среда для создания веб-приложения и установлены все необходимые модули для функционирования рабочей среды.

2.2 Разработка проекта

Для начала работы был создан проект в сервисе firebase, который будет служить сервисом обработки авторизационных запросов, хранилищем данных о пользователях (логины, пароли), базой данных реального времени и хранилищем файлов это показано на рисунках 6, 7.

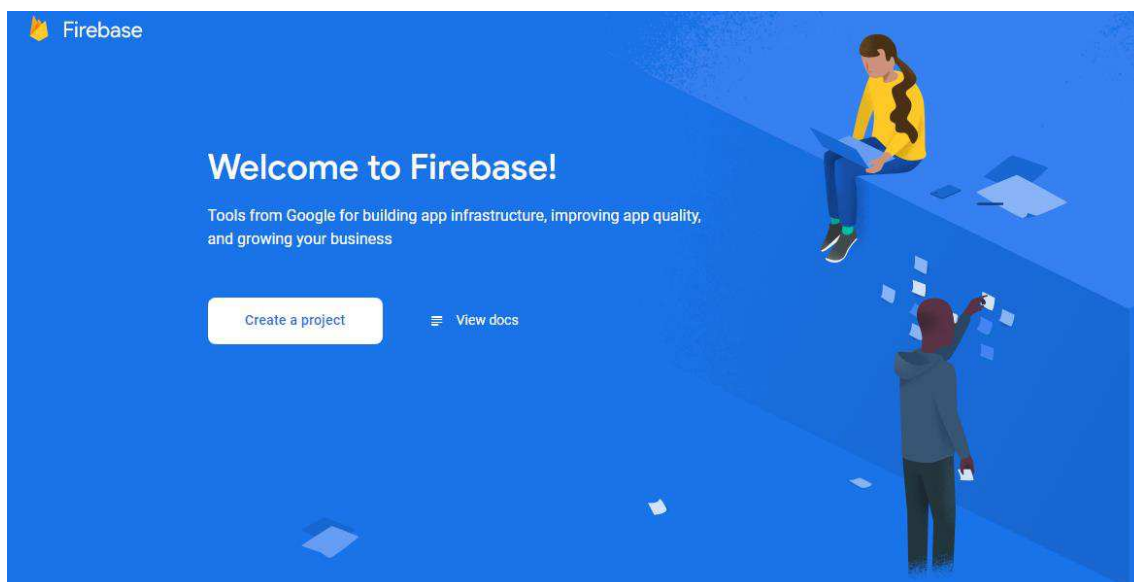


Рисунок 6 – Главная страница сайта FireBase



<input type="checkbox"/>	Name	Size	Type	Last modified
<input type="checkbox"/>	news_image/	–	Folder	–
<input type="checkbox"/>	profile_image/	–	Folder	–
<input type="checkbox"/>	showplaces_image/	–	Folder	–

Рисунок 7 – Хранилище файлов Firebase

После создания базы данных была сгенерирован специальный ключ приложения для возможности подключения базы данных к приложению.

Параллельно с разработкой авторизации велась и разработка коллекций для хранения информации. Список коллекций представлен на рисунке 8. А пример коллекции показан на рисунке 9.

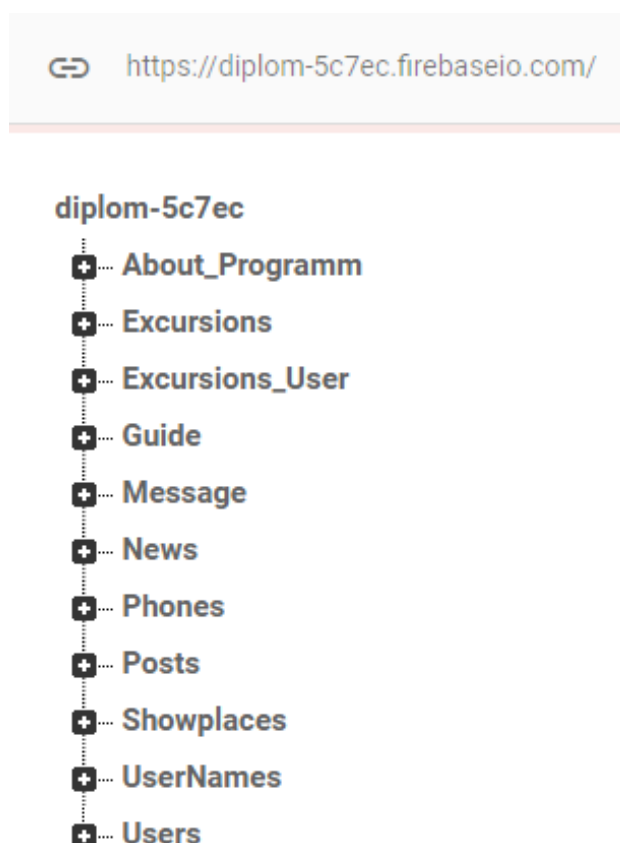


Рисунок 8 – Список коллекций базы данных

FireBase – облачная СУБД. Firebase имеет огромную масштабируемость. Для данного проекта такой уровень масштабируемости не нужен, поэтому создавались коллекции, по которым система обращалась к базе данных. Данные коллекции помогают указать строгую структуру для базы данных системы. В моделях указываются поля, которые находятся в базе данных и типы значения полей. Более строгая модель необходима для исключения ошибок в выполнении различных запросов с неправильными наименованиями полей.

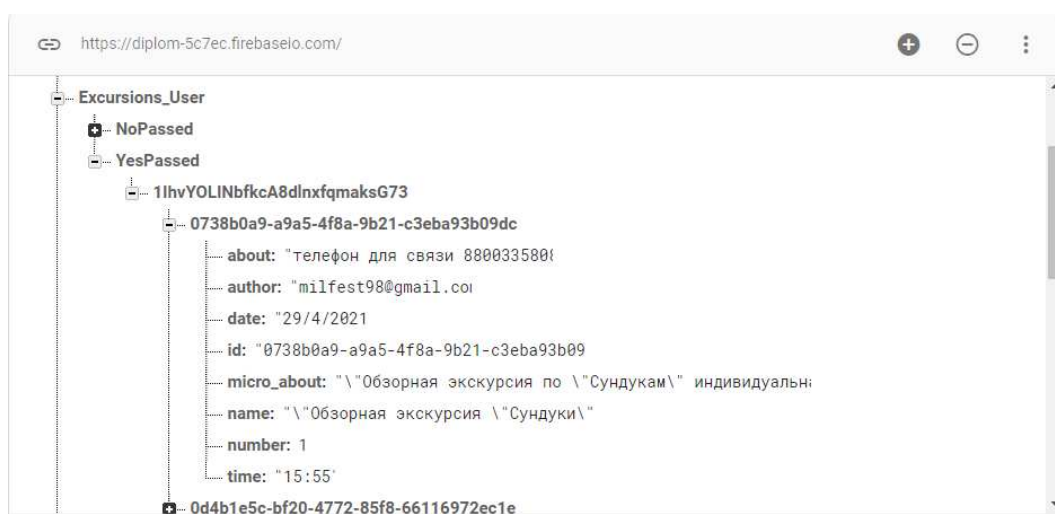


Рисунок 9 – Пример коллекции экскурсии

Далее была разработана основа серверной части приложения, где осуществляется подключение к базе данных при помощи пакета rugebase. Основная серверная часть приложения содержится на рисунке 10.

```

from urllib.parse import quote
pyrebase.quote = lambda s, safe=None: s
def get_url(self, token=None):
    path = self.path
    self.path = None
    if path.startswith('/'):
        path = path[1:]
    if token:
        return "{0}/o/{1}?alt=media&token={2}".format(self.storage_bucket, quote(path, safe=''), token)
    return "{0}/o/{1}?alt=media".format(self.storage_bucket, quote(path, safe=''))
pyrebase.Storage.get_url = lambda self, token=None: \
    get_url(self, token)
app = Flask(__name__)
app.secret_key = os.urandom(24)
config = {
    "apiKey": "AIzaSyADkHUJ_u6uTKAVnm7XdkqIn_C10vpSYGI",
    "authDomain": "diplom-5c7ec.firebaseio.com",
    "databaseURL": "https://diplom-5c7ec.firebaseio.com",
    "projectId": "diplom-5c7ec",
    "storageBucket": "diplom-5c7ec.appspot.com",
    "messagingSenderId": "785695600410",
    "appId": "1:785695600410:web:b7c54b97c95030ec5a06a3",
    "measurementId": "G-Z45B4E75E"
}
def isAuthenticated(f):
    @wraps(f)
    def decorated_function(*args, **kwargs):
        #check for the variable that pyrebase creates
        if not auth.current_user != None:
            return redirect(url_for('signup'))
        return f(*args, **kwargs)
    return decorated_function
firebase = pyrebase.initialize_app(config)
auth = firebase.auth()
db = firebase.database();

```

Рисунок 10 –Создание сервера для приложения

В основной серверной части подключаются основные пакеты приложения и происходит описание работы приложения при появлении запросов и ответов от сервера. Такие ответы и запросы в разработке называются роутами. Далее для каждого роута описывается логика обработки ее со стороны сервера и ответа на него. В конце файла серверной части задается порт для сервера, для данного приложения был установлен порт 5000. Далее объявляется асинхронная функция start, в которой происходит подключение к базе данных, если подключение не происходит, то сервер не стартует, а выдает ошибку сервера и завершает процесс.

Роут авторизации и входа. За авторизацию отвечает роут «/», он показан на рисунке 11.

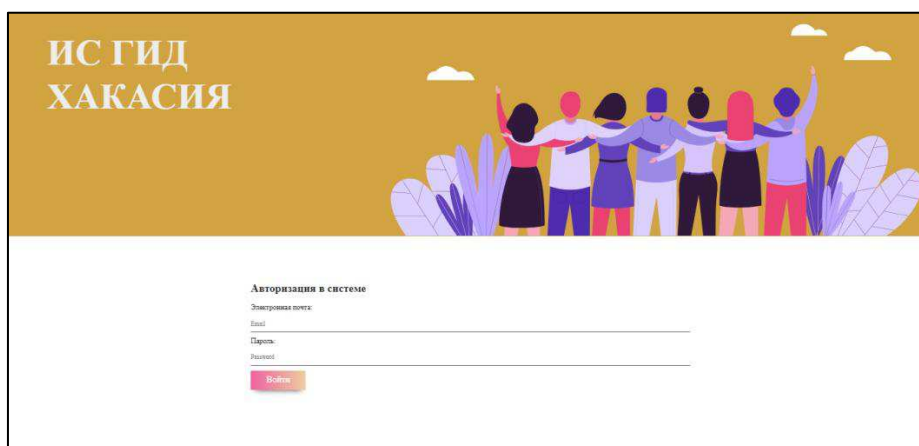
Для авторизации в приложении данные, попадая в роут, разбиваются на две переменные: login и password. Далее в базе данных при помощи запроса производится поиск пользователя с логином, записанным в переменную login. Если пользователь не найден, то на сервер отправляется статус 400 и сообщение о том, что пользователь не найден. Если пользователь найден, то запрос выдает его параметры, и далее идет проверка его пароля. Пароль, взятый из формы, проходит шифрование и сравнивается с зашифрованным паролем, который хранился в базе данных, и если они

совпадают, то пользователю выдается токен, на основании которого он попадает в систему. Если же пароли не совпадают, то в систему возвращается ответ 400, и сообщение: «Неверный пароль, попробуйте снова».

```
37
38 @app.route("/", methods= ['GET', 'POST'])
39 def login():
40     if request.method == "POST":
41         #get the request data
42         email = request.form["email"]
43         password = request.form["password"]
44         try:
45             #login the user
46             user = auth.sign_in_with_email_and_password(email, password)
47             #set the session
48             user_id = user['idToken']
49             user_email = email
50             session['usr'] = user_id
51             session["email"] = user_email
52             return redirect("/index")
53         except:
54             return render_template("login.html", message="1" )
55
56
57
58 return render_template("login.html")
```

Рисунок 11 – Роут «Auth»

Страница для входа имеет два поля, в одно вводится логин, а во второе пароль. Далее данные с формы отправляются на обработку в роут «/». Страница авторизации показана на рисунке 12.



ИС ГИД
ХАКАСИЯ

Авторизация в системе

Электронная почта:

Пароль:

Remember

Войти

Рисунок 12 – Страница авторизации

После авторизации пользователь попадает в систему. Пользователю выдается сообщение о том, что он успешно авторизовался. Далее пользователю становятся доступны все функции системы.

Далее была создана главная страница приложения. Со страницей также взаимодействует роут. Роут использует гиперссылки и специальные переменные хранилища и начинает внесение данных, полученных с формы в базу данных путем запроса. Главная страница приложения показана на рисунке 13.

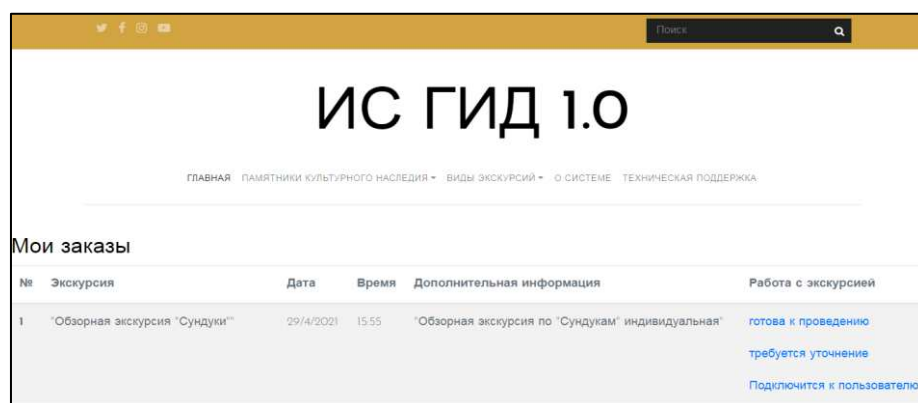


Рисунок 13 – Главная страница приложения

Для отображения данных по заказам были созданы 2 таблицы под специальными тэгами, размещенными на главной странице html(рисунок 14).

Доступные заказы						
№	Экскурсия	Статус	Дата	Время	Дополнительная информация	Взять в работу
1	Экскурсия по курганам Салбыкской степи	заявка отправлена	31/5/2021	13:32	Обзорная экскурсия Курганы Салбыкской степи	Взять в работу
2	"Экскурсия по музею "Казановка"	заявка отправлена	18/6/2021	13:04	Обзорная экскурсия музей Казановка	Взять в работу
3	"Обзорная экскурсия "Сундуки"	заявка отправлена	1/6/2021	12:30	"Обзорная экскурсия "Сундуки"	Взять в работу

Рисунок 14 – Таблица отображения информации о доступных заказах

Используется получение данных из коллекции экскурсий для внесения данных в массив post. Далее при помощи функции, прикрепленной к событию нажатие на кнопку, срабатывает функция Rendertemplate. В функции Rendertemplate происходит запрос для нахождения всех экскурсий, которые не имеют статус «взяты в работу», далее выполняется второй запрос,

который по id всегда отображает информацию о взятых заказах на экскурсии. Также на данной странице присутствует useEffect, который при загрузке страницы выполняет функцию searchReports. Функция searchReports выполняет запрос к роуту search и далее к роутеру searchReports, который находится в самом роуте. Отправка запроса происходит методом POST. Данная страница показана на рисунке 15.

```
@app.route("/status/<id>", methods=["GET", "POST"])
def edit0(id):
    if request.method == "POST":
        title = 'взять в работу'
        post = {
            "status": status,
            "author": session["email"]
        }

        #update the post
        db.child("Posts").child(id).update(post)
        return redirect("index.html")
    return redirect("index.html")
@app.route("/status1/<id>", methods=["GET", "POST"])
def edit1(id):
    if request.method == "POST":
        title = 'готова к проведению'
        post = {
            "status": status
        }

        #update the post
        db.child("Posts").child(id).update(post)
        return redirect("index.html")
    return redirect("index.html")
@app.route("/status2/<id>", methods=["GET", "POST"])
def edit2(id):
    if request.method == "POST":
        title = 'требуется уточнения'
        post = {
            "status": status
        }
```

Рисунок 15 – Страница вывода информации о заказах

Страницы, на которых отображается детальная информация о каждом заказе или же новости на главной странице, строятся с помощью специального запроса. Страница, которая выдает подобные результаты, должна иметь в своей структуре динамическую ссылку на объект в базе данных. Результат запроса формируется в виде специальных тэгов. В данной странице использовался компонент. Компонент — это заранее подготовленный элемент, который выгружается на страницу при помощи вызова компонента. На рисунке 16 показан код для обращения к конкретной записи из базы данных. На строке 25 виден вызов компонента, в который

передается массив данных. Далее на рисунке 17 показан сам компонент, который формируется из полученного массива и выводится на страницу.

```
@app.route("/post/<id>")
@isAuthenticated
def post(id):
    orderedDict = db.child("Posts").order_by_key().equal_to(id).limit_to_first(1).get()
    print(orderedDict)

    return render_template("post.html", data=orderedDict)
```

Рисунок 16 – Страница для вывода информации о заказе

```
<section class="site-section pt-5">
<h2>Мои заказы</h2>
<table class="table table-striped table-hover">
<thead>
<tr>
<th scope="col">№</th>
<th scope="col">Экскурсия</th>
<th scope="col">Дата</th>
<th scope="col">Время</th>
<th scope="col">Дополнительная информация</th>
<th scope="col">Работа с экскурсией</th>
</tr>
</thead>
<tbody>
{% for post in posts.each() %}
    {% if session["email"] == post.val().author %}
<tr>
<th scope="row">{{post.val().number}}</th>
<td>{{post.val().name}}</td>
<td>{{post.val().date}}</td>
<td>{{post.val().time}}</td>
<td>{{post.val().micro_about}}</td>
<td><a href="/status/{{post.key()}}">готова к проведению</a>
<p></p>
<a href="/status/{{post.key()}}">требуется уточнение</a>
<p></p>
    {% endif %}
    {% endfor %}
</tbody>
</table>
<h2>Доступные заказы</h2>
<table class="table table-striped table-hover">
<thead>
<tr>
<th scope="col">№</th>
```

Рисунок 17 – Компонент вывода списка экскурсий

Страница редактирования информации. На данной странице имеются специальные поля для редактирования данных. По логике работы системы обращения к системе устроены так, что только пользователь - создатель информации может вносить в нее изменения, исключение составляет учетная запись администратора.

Данные записываются и удаляются с помощью отправки данных в роуты, в которых выполняются запросы с данными, поступающими со страницы.

Роуты нужны для функционирования системы, и они играют огромную роль. Роут создания новости показан на рисунке 18. Данный роут получает данные с главной страницы и проводит валидацию данных и выполняет запрос в базу данных на сохранение информации.

```
#create form
@app.route("/create", methods=["GET", "POST"])
@isAuthenticated
def create():

    if request.method == "POST":
        #get the request data
        title = request.form["title"]
        content = request.form["content"]

        post = {
            "title": title,
            "content": content,
            "author": session["email"]
        }

        try:
            #print(title, content, file=sys.stderr)

            #push the post object to the database
            db.child("Posts").push(post)
            return redirect("/")
        except:
            return render_template("create.html", message= "Ошибка создания записи")

    return render_template("create.html")
```

Рисунок 18 – Роут создания новости

Далее на рисунке 19 показаны два роута для редактирования записи в базе данных и удаления. После получения запроса к роутам они берут тело

запроса и формируют запрос к базе данных, совмещая его с телом запроса. Далее идет процедура сохранения.

```
@app.route("/edit/<id>", methods=["GET", "POST"])
def edit(id):
    if request.method == "POST":

        title = request.form["title"]
        content = request.form["content"]

        post = {
            "title": title,
            "content": content,
            "author": session["email"]
        }
        #update the post
        db.child("Posts").child(id).update(post)
        return redirect("/post/" + id)
        #update the post
        db.child("Posts").child(id).update(post)
        return redirect("index.html")
    return redirect("index.html")
@app.route("/delete/<id>", methods=["POST"])
def delete(id):
    db.child("Posts").child(id).remove()
    return redirect("/")
```

Рисунок 19 – Роуты для редактирования и удаления записей

Роут для удаления записи получает запрос, в теле которого находится `id` записи и `id` пользователя, для удостоверения полномочий на удаление данных. Далее путем запроса данные удаляются из базы данных.

На данный момент, поскольку приложение не развернуто на серверах заказчика, для проверки работоспособности временно используется сторонний сервис видеосвязи Linkello. Данный видеосервис позволяет подключать мобильное и веб-приложение без использования своего сервера для обработки видеопотока. Для того чтобы данный сервис работал одновременно на разных платформах использовалась технология API данного сервиса и подключалась как в мобильное так и в веб-приложение. На рисунке 20 представлен код для подключения сервиса видеосвязи в веб-приложение. Необходимыми условиями работы являются наличие учетной

записи сервиса и доступные порты для приема и отправки видеосигнала. После добавления данного кода при нажатии на кнопку взять в работу экскурсию для каждой экскурсии формируется уникальная ссылка, которая добавляется в базу данных реального времени и затем в мобильное приложение, поэтому дублирование или случайного входа на экскурсию быть не может.

```
1
2 onBistriConferenceReady = function () {
3     var localStream;
4     bc.init( {
5         appId: "38077edb",
6         appKey: "4f304359baa6d0fd1f9106aaeb116f33"
7     } );
8     if ( !bc.isCompatible() ) {
9         alert( "your browser is not WebRTC compatible !" );
10        return;
11    }
12    bc.signaling.bind( "onConnected", function () {
13        bc.startStream( "640x480", function( stream ){
14            localStream = stream;
15            bc.attachStream( localStream, document.body, { mirror: true } );
16            bc.joinRoom( "conference_gid" );
17        } );
18    } );
19    bc.signaling.bind( "onJoinedRoom", function ( result ) {
20        var roomMembers = result.members;
21        for ( var i=0, max=roomMembers.length; i<max; i++ ) {
22            bc.call( roomMembers[i].id, "conference_gid", { stream: localStream, 'video-codec': 'H264/9000' } );
23        }
24    } );
25    bc.streams.bind( "onStreamAdded", function ( remoteStream ) {
26        bc.attachStream( remoteStream, document.body );
27    } );
28    bc.streams.bind( "onStreamClosed", function ( stream ) {
29        bc.detachStream( stream );
30    } );
31    bc.connect();
32
33 }
34
35
```

Рисунок 20 – Код для подключения видеосвязи в приложения

На этапе разработки приложения при переходе по различным ссылкам, а именно ссылкам видеосвязи и загрузки и выгрузки рисунков с базы данных

реального времени было замечено, что фреймворк автоматически добавляет специальные символы, которые не распознаются кодировкой UTF-8. Для решения данной проблемы было решено воспользоваться функциями декорирования строк, которые присутствуют в языке программирования python, но доработать их и использовать в виде лямбда функции. На рисунке 21 представлен код лямбда функции декорирования ссылок на ресурсы приложения.

```
6 from urllib.parse import quote
7 pyrebase.quote = Lambda s, safe=None: s
8 def get_url(self, token=None):
9     path = self.path
10    self.path = None
11    if path.startswith('/'):
12        path = path[1:]
13    if token:
14        return "{0}/o/{1}?alt=media&token={2}".format(self.storage_bucket, quote(path, safe=''), token)
15    return "{0}/o/{1}?alt=media".format(self.storage_bucket, quote(path, safe=''))
16 pyrebase.Storage.get_url = Lambda self, token=None: \
17    get_url(self, token)
```

Рисунок 21 – Лямбда функция для декорирования ссылок в приложении

По итогу проделанной работы в данном пункте было разработано веб-приложение, установлена связь с базой данной реального времени и произведена настройка сервиса видеосвязи для оказания удаленных экскурсий на объектах культурного наследия республики Хакасия.

2.3 Описание информационной системы

Для корректной работы с информационной системой в веб-приложении пользователю необходим компьютер с устойчивым проводным интернет соединением, наличием звукозаписывающего устройства (микрофон) и устройства аудиовывода (колонки, наушники). Наличие веб-камеры желательно, но необязательно, ведь при проведении экскурсии самое главное чтобы гид видел происходящее вокруг пользователя, а видеть самого гида не

всегда нужно. Компьютеры, которые используются в Министерстве культуры и подведомственных ему учреждениях, смогут корректно отображать информацию и работать в данной системе.

При открытии страницы приложения пользователю предлагается авторизоваться в системе. Страница показана на рисунке 22.

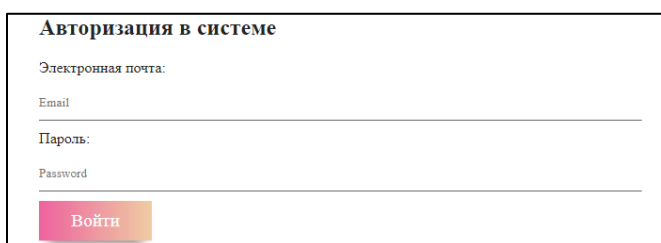


Рисунок 22 – Окно авторизации в системе

После авторизации открывается главная страница, где пользователю открывается компонент NavBar. В компоненте NavBar присутствуют кнопки для навигации по системе. Нажимая на соответствующую кнопку можно попадать на разные страницы системы. Страница поиска показана на рисунке 23.

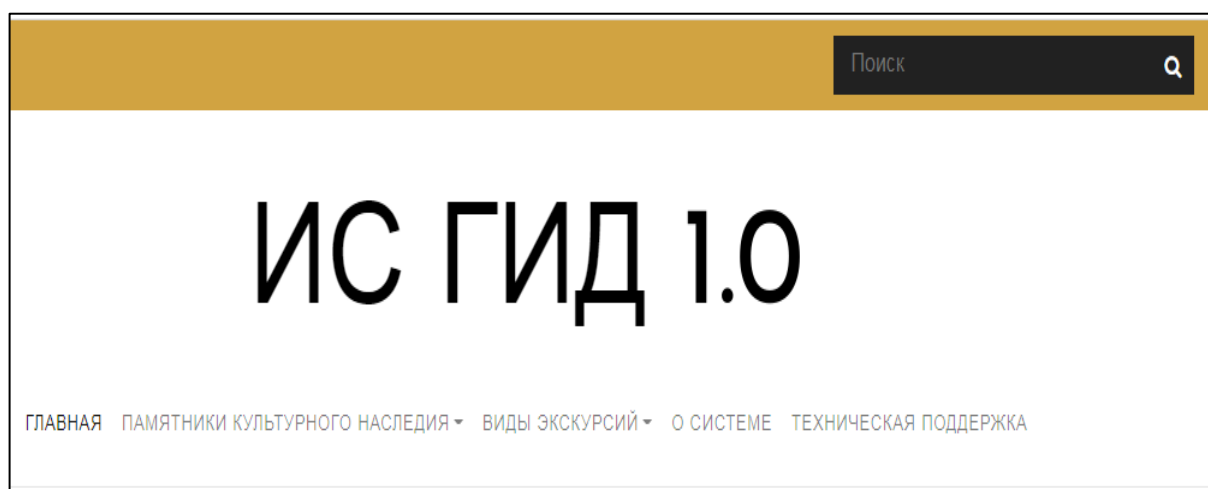


Рисунок 23 – Главное меню приложения

Затем на главной странице пользователю (гиду) сверху представлены 2 таблицы с доступными и взятыми заказами на экскурсии, которые

представлены на рисунке 24. В каждой таблице в крайнем правом поле находятся кнопки для работы с заявкой. При необходимости заявку можно взять в работу, попросить уточнение данных (используется связь с пользователем), либо перейти к странице, на которой происходит оказание услуги.

Мои заказы						
№	Экскурсия	Дата	Время	Дополнительная информация	Работа с экскурсией	
1	"Обзорная экскурсия "Сундуки"	29/4/2021	15:55	"Обзорная экскурсия по "Сундукам" индивидуальная"	готова к проведению требуется уточнение Подключится к пользователю	
2	"Обзорная экскурсия по "Салбыкский курган"	21/7/2021		Экскурсия по курганам Салбыкской степи	готова к проведению требуется уточнение Подключится к пользователю	
3	"групповая экскурсия "музей под открытым небом Усть-Сос"	30/8/2021	13:19	Обзорная экскурсия по музею	готова к проведению требуется уточнение Подключится к пользователю	
Доступные заказы						
№	Экскурсия	Статус	Дата	Время	Дополнительная информация	Взять в работу
1	Экскурсия по курганам Салбыкской степи	заявка отправлена	31/5/2021	13:32	Обзорная экскурсия Курганы Салбыкской степи	Взять в работу
2	"Экскурсия по музею "Казановка"	заявка отправлена	18/6/2021	13:04	Обзорная экскурсия музей Казановка	Взять в работу
3	"Обзорная экскурсия "Сундуки"	заявка отправлена	1/6/2021	12:30	"Обзорная экскурсия "Сундуки"	Взять в работу

Рисунок 24 – Страница с таблицами заказов

При нажатии кнопки «подключится к пользователю» происходит открытие страницы видеосвязи, где оказывается сама услуга, там имеется внутренний чат. Система видеосвязи позволяет подключать более 100 пользователей одновременно, окно видеосвязи с пользователем представлено на рисунке 25.

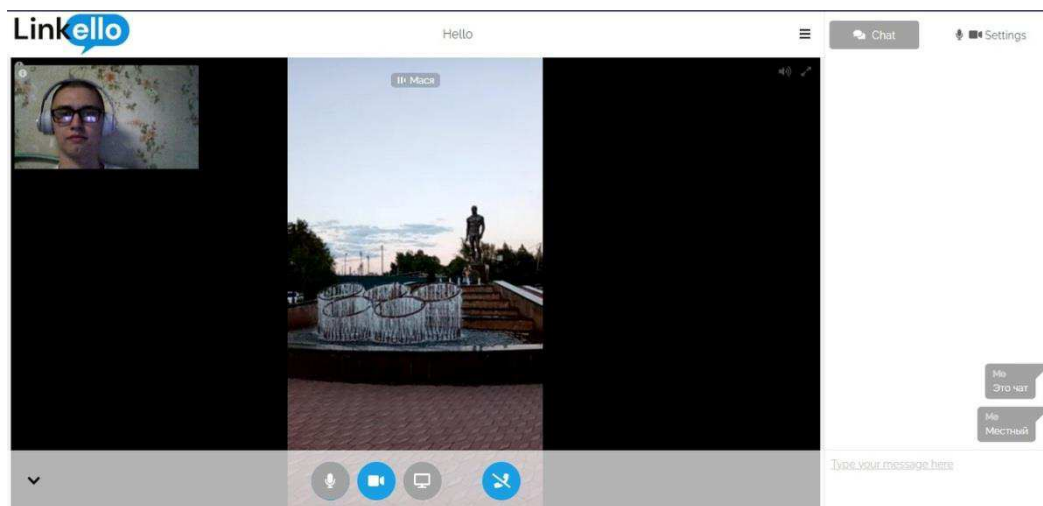


Рисунок 25 – Страница видеосвязи с пользователем

В меню на главной странице имеется кнопка «Виды экскурсий», при нажатии на эту кнопку открывается страница, на которой можно ознакомиться с информацией об оказываемых услугах и условиях их оказания.

При нажатии на кнопку «О системе» открывается страница с информацией о системе, разработчиках и контактных данных держателей системы. Данный раздел будет оформлен после передачи информационной системы заказчику. В разделе «Техническая поддержка» представлены данные для связи со службой поддержки данной системы.

Следующим разделом, который располагается после таблиц с заказами, является раздел «Новости» системы, представленные на рисунке 26.

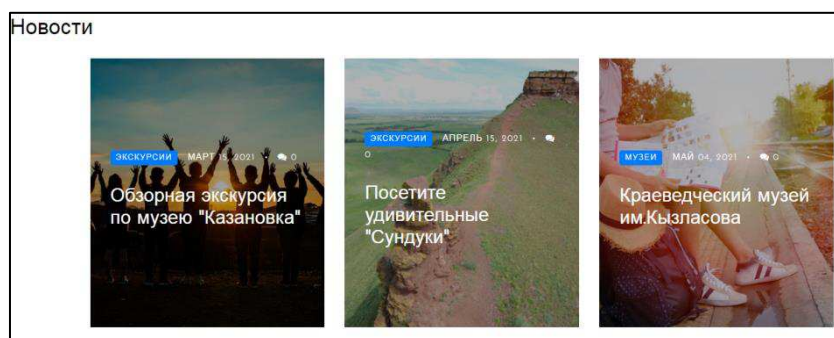


Рисунок 26 – Раздел «Новости» главной страницы сайта

За создание данной вкладки и ее наполнение отвечает администратор системы и на данный момент в нее внедрены тестовые записи, которые в последующем будут удалены.

Для внутренней работе на сайте кроме учетных записей для гидов были созданы учетные записи администратора. Учетная запись администратора позволяет добавлять новости, информацию об экскурсиях, а также информацию о памятниках культурного наследия. Для добавления информации используются кнопки в таблице с необходимым типом информации, представленные на рисунке 27.

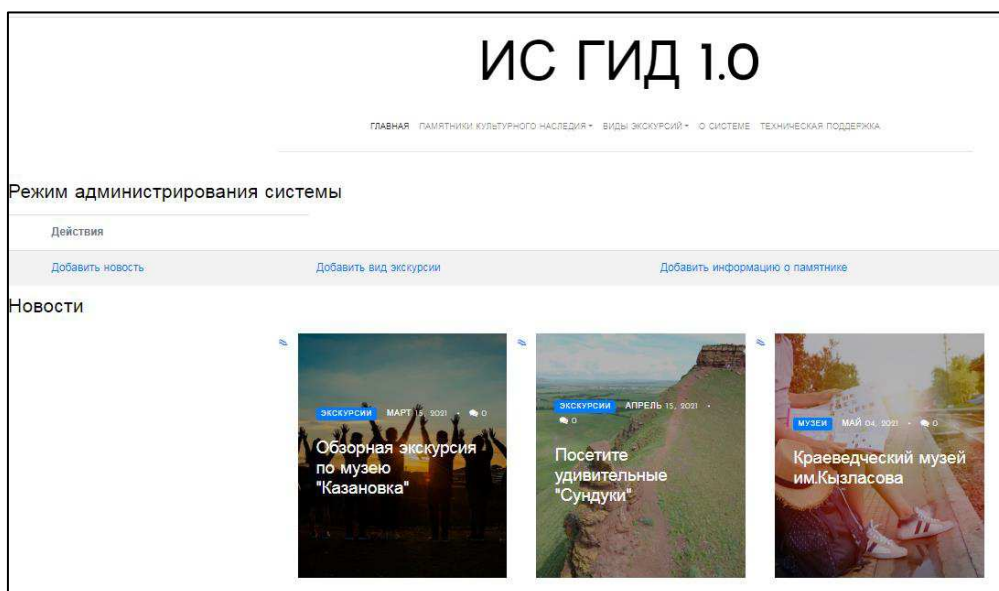


Рисунок 27 – Главная страница администратора приложения

При нажатии кнопки «Добавить запись»или же «Редактировать запись»(значок карандаша возле записи) открывается страница работы с данной информацией, где присутствуют поля для ввода информации и кнопка для сохранения информации, после нажатия данной кнопки информация загружается в базу данных реального времени, страница ввода информации представлена на рисунке 28.

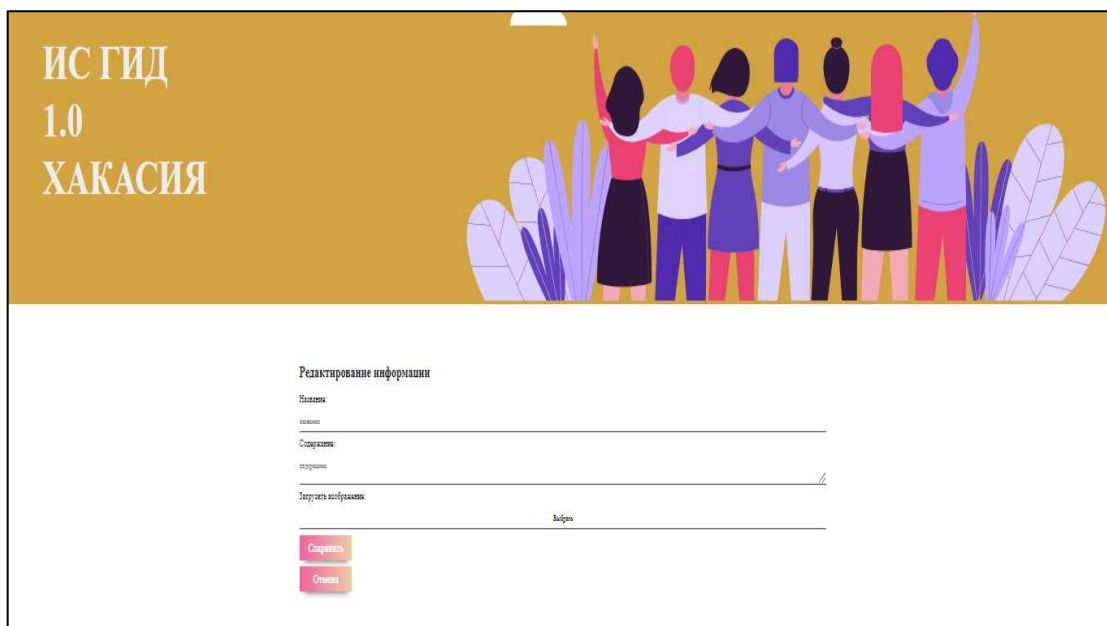


Рисунок 28 – Страница создания и редактирования информации

Так же, поскольку приложением будут пользоваться сотрудники министерства культуры, для них создаются специальные учетные записи, с помощью которых они смогут получать информацию для заполнения статистических форм по оказанным платным экскурсионным услугам в республике Хакасия и получение рейтинга самых активных гидов, работающих через приложение. На рисунке 29 представлена главная страница приложения для сотрудников министерства.

ИС ГИД 1.0

[ГЛАВНАЯ](#)
[ПАМЯТНИКИ КУЛЬТУРНОГО НАСЛЕДИЯ](#)
[ВИДЫ ЭКСКУРСИЙ](#)
[О СИСТЕМЕ](#)
[ТЕХНИЧЕСКАЯ ПОДДЕРЖКА](#)

Режим мониторинга системы

Данные для АИС "Статистическая отчетность отрасли"

период	на 01.04.21	на 01.05.21	на 01.06.21	на 01.07.21	на 01.08.21
Количество платных оказанных экскурсионных услуг	1	1	1	-	-

Рейтинг гидов, работающих в системе

№ п/п	эл.почта	кол-во оказанных услуг
1	milfest98@gmail.com	3
2	guster@mail.ru	0

Рисунок 29– Главная страница для сотрудников министерства культуры

По итогам пункта были описаны основные функции и страницы веб приложения информационной системы, рассказаны основные характеристики, необходимые для работы в системе, описаны возможности по работе с данными для различных групп пользователей.

2.4 Выводы по разделу «Разработка веб-приложения для информационной системы оказания удаленных экскурсий на объектах культурного наследия республики Хакасия»

Созданная система имеет доступ к данным в облачной базе данных. Может выполнять запросы на создание данных и редактирование данных. В данной системе можно отслеживать информацию и оказывать удаленные экскурсии по видеосвязи. Что и требовалось для заказчика. В системе есть функции:

- Вывод списка экскурсий.
- Редактирование работы с экскурсией.
- Редактирование записей.
- Добавление записей.
- Удаление записей.

В ходе создания информационной системы были использованы Python, FireBase, Flask. FireBase использовалась как облачная база данных, в ходе разработки использовалось облачное хранилище с бесплатным тарифом, но при вводе в эксплуатацию сервер будет локальным на серверах заказчика, что позволит хранить в базе конфиденциальную информацию. В показанных примерах была внесена информация, случайно сгенерированная генератором записей. Flask был использован для работы в веб среде с помощью языка программирования python. Flask использовался для серверной части. Ngrok запускал сервер и подключался к базе данных. Также с помощью Flask создавались роуты в системе, которые позволили обрабатывать

информацию и выполнять запросы к базе данных. Итогом разработки получено веб-приложение, которое удовлетворяет задачи заказчика.

3 Оценка экономической эффективности разработки информационной системы оказания онлайн экскурсий на объектах культурного наследия Республики Хакасия

3.1 Расчет затрат реализации проекта информационной системы Республики Хакасия

При расчете затрат на создание информационной системы для организации оказания удалённых экскурсий на объектах культурного наследия Республики Хакасия должно учитываться, что капитальные и эксплуатационные затраты рассчитываются для мобильного и веб-приложения отдельно, а совокупная стоимость владения (ТСО) и экономическая эффективность являются общими составляющими разработанной системы.

3.1.1 Капитальные затраты

Затраты на разработку информационной системы вычисляются по формуле 1.

$$K=K_{\text{пр}}+K_{\text{тс}}+K_{\text{лс}}+ K_{\text{ло}}+K_{\text{ио}}+K_{\text{об}}+K_{\text{оз}}, \quad (1)$$

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

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

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

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

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

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

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

Затраты на разработку. Затраты на проектирование ИС рассчитываются по формуле 2.

$$K_{пр} = K_{зп} + K_{ппс} + K_{свт} + K_{проч}, \quad (2)$$

где $K_{зп}$ – затраты на заработную плату программиста;

$K_{ппс}$ – затраты на инструментальные программные средства проектирования;

$K_{свт}$ – затраты на средства вычислительной техники для проектирования;

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

Разработку данного продукта будет выполнять студент и для оплаты его работы будет использована тарифная система оплаты труда, соответствующая оплаты штатного программиста Министерства культуры Республики Хакасия.

Для расчета заработной платы программиста необходимо составить таблицу заработной платы и посчитать ФОТ (фонд оплаты труда), который составляет 30,2% от заработной платы. Оклад работы программиста за месяц равен 19.500 рублей. Продолжительность работы в месяц 22 дня. Программист работает над проектом 66 дней, значит, оклад необходимо посчитать на 66 дней работы. Следовательно, заработная плата программиста равна 58476 рублей (без учета северной надбавки и регионального коэффициента, а также без учета НДФЛ).

Расчет заработной платы программиста с учетом северной надбавки (30%) и регионального коэффициента (30%) произведен по тарифному способу оплаты, но поскольку студент не работает в данной организации более 1,5 года, то применять к нему северный надбавку не требуется.

Оплата за всю проделанную работу с учетом северной надбавки и регионального коэффициента, а также с учетом НДФЛ представлена в таблице 2.

Таблица 2– Расчет заработной платы программиста

Доходы, руб.		Расходы, руб.	
Заработная плата	58476		
Северная надбавка	-		
Районный коэффициент	17542		
Все доходы	76018	НДФЛ	9882
Итого на руки	66136		

Итого: $K_{зп}=76018*1,302=98975$ рублей.

После расчета ФОТ необходимо так же рассчитать другие затраты, связанные с проектированием ИС.

Для создания данной информационной системы программисту необходимо приобрести следующее оборудование (Таблица 3).

Таблица 3 – Стоимость оборудования

Наименование оборудования	Наименование элементов оборудования	Количество элементов	Стоимость ед., руб.	Стоимость всего, руб.
Компьютер для разработки ИС и ее дальнейшей работы	18.5" Монитор ASUS VS197DE	1	4699	4699
	Корпус Zalman Z3 черный	1	3299	3299
	Процессор Intel Core i5 Haswell	1	10140	10140

Окончание таблицы 3

Оперативная памятьSODIMM KingstonValueRam [KVR16LS11/8]	2	5399	10798
Жесткийдиск Western Seagate BarraCuda Pro [ST1000LMO49] 1 TB	1	4399	4399
ВидеокартаMSI PCI-E GTX10502GOC nVidia GeForce GTX 1050	1	9300	9300
Материнская плата MSIB250MGamingPRO	1	5999	5999
Мышь проводная DEXPCM-904BUсиний	1	250	250
Блокпитания Cooler Master Master Watt 650W	1	5899	5899
Клавиатура Sven KB- C7100EL	1	650	650
Итого	11		55433

Расчет амортизации оборудования на срок эксплуатации в проекте. Для каждой детали компьютера средний срок службы различен и составляет:

- процессор – 4 года;
- материнская плата – 5 лет;
- жесткий диск – 8 лет;
- модуль оперативной памяти – 5 лет;
- блок питания – 6 лет;
- видеокарта – 4 года;
- монитор – 5 лет;
- корпус – 8 лет.

Поэтому срок эксплуатации компьютера равен 5 годам – $T_{эк}$.

$$A_{год} = C_б * H_{ам}, \quad (3)$$

где $C_б$ – это стоимость балансовая;

$H_{ам}$ – норма амортизации.

$$H_{ам} = \frac{100\%}{T_{эк}}, \quad (4)$$

где $T_{эк} = 5$, значит $H_{ам} = 20\%$, $C_б = 55433$ руб., $A_{год} = 0,20 * 55433 = 11087$.

Амортизация проекта рассчитывается по формуле 5.

$$A_{пр} = \frac{A_{год}}{K_{рдр}} * K_{дэ}, \quad (5)$$

где $A_{пр}$ – амортизация проекта;

$K_{рдр}$ – количество рабочих дней в году, которое равно 247;

$K_{дэ}$ – количество дней эксплуатации, которое равно 66.

$$A_{пр} = (11087/247) * 66 = 2962 \text{ руб.}$$

После приобретения компьютера необходимо приобрести лицензионные версии данного программного обеспечения (Таблица 4).

Таблица 4 – Стоимость программного обеспечения

Наименование программного обеспечения	Стоимость, руб.	Срок использования
Microsoft Windows 10 Professional (64 bit)	8499	Бессрочно
Office 2016	4068	1 год
Firebase	0	Бессрочно

Окончание таблицы 4

Flask framework	0	Бессрочно
Итого	12567	1 год

Для расчета амортизации программного обеспечения необходимо стоимость программного обеспечения разделить на количество реализованных программных продуктов за этот срок. За период использования программного обеспечения возможно создать 8 программных продуктов, поэтому стоимость всех программ необходимо уменьшить в 8 раз. Следовательно, амортизация равна $12567/8=1570$ рублей.

Для поддержания имеющихся комплектующих компьютера и программного обеспечения в рабочем состоянии необходимы дополнительные затраты в размере 470 рублей в месяц.

Затраты на проектирование ИС представлены в таблице 5.

Таблица 5 – Затраты на проектирование ИС

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на проектирование ИС	Затраты на заработную плату программиста	98975
	Затраты на ПО	1570
	Затраты на средства вычислительной техники	2962
	Прочие затраты (3% всех затрат)	3108
Итого		106615

Итого: $K_{пр}=98975 + 1570 + 2962 + 3108 = 106615$ рублей.

Затраты на технические средства управления. Затраты на технические средства управления включают в себя стоимость компьютера и

серверного ПК. Стоимость компьютера приведена в таблице 1, и данный компьютер будет служить сервером для данной ИС, итого:

$K_{тс}=55433=55433$ рублей.

Затраты на создание линий связи. Для вычисления затрат на создание линий связи необходимо приобрести роутер, поскольку будет необходим дополнительный маршрутизатор для обработки обращений с сервером и выделенная линия связи, отличная от правительственной и оплатить интернет с оплатой 500руб./месяц, т.к. тестирование ИС будет проводиться 3 месяца, значит затраты на интернет будут равны 1500 рублей. Затраты на создание линий связи приведены в таблице 6.

Таблица 6 –Затраты на создание линий связи

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на создание линий связи	Затраты на роутер	1000
	Затраты на интернет	1500
Итого		2500

Итого: $K_{лс}=2500$ рублей.

Затраты на программные средства. Затраты на программные средства управления отсутствуют, поскольку разработка ведется уже в данной системе, следовательно $K_{по}=0$ рублей.

Затраты на формирование информационной базы. Затраты на формирование информационной базы учитываются в расчете заработной платы программиста, следовательно, её можно не учитывать, значит $K_{ио}=0$ рублей.

Затраты на обучение персонала. Т.к. на обучение персонала необходимо выделить 4 часа, а работа специалиста по обучению работы с системой в среднем тарифицируется по 100 рублей в час, следовательно:

$$K_{об} = 400 (1,3 * 1,3) * 1,302 = 833 \text{ рублей.}$$

Затраты на опытную эксплуатацию. Затраты на опытную эксплуатацию включают в себя зарплату руководителя организации за время тестирования программы. Оклад руководителя компании равен 50000 рублей, следовательно, чтобы вычислить заработную плату директора за 6 часов тестирования программы необходимо найти заработную плату за 1 час и увеличить в 6 раз. В рабочем дне у директора компании 8 часов рабочего времени, в месяце 25 рабочих дней, следовательно, необходимо оклад разделить на 25 и умножить на 0,75, где 0,75 — это 6 часов тестирования программы.

$$З/п = 50000 / 25 * 0,75 = 1500 \text{ рублей.}$$

Данные затраты приведены без учета северного коэффициента и районной надбавки, а также без учета НДС.

$$\text{Итого: } K_{ос} = 1500 * 1,302 * 1,3 * 1,3 = 3296 \text{ рублей.}$$

Капитальные затраты, которые рассчитываются по формуле 6 равны:

$$K = 98875 + 55433 + 2500 + 3108 + 833 + 3296 = 164045 \text{ рублей.}$$

Список капитальных затрат представлен в таблице 7 и рисунке 6.

Таблица 7 – Список капитальных затрат

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на разработку ИС	Затраты на заработную плату разработчика	98875
	Затраты на программные	1570
	Затраты на средства вычислительной техники	2962
	Прочие затраты на разработку	3108
Затраты на технические средства управления		55433
Затраты на создание линий связи локальных сетей		2500
Затраты на программные средства		0
Затраты на формирование информационной базы	ЗП разработчика	0
Затраты на обучение персонала	ЗП обучаемого	833
Затраты на опытную эксплуатацию	ЗП директора	3296

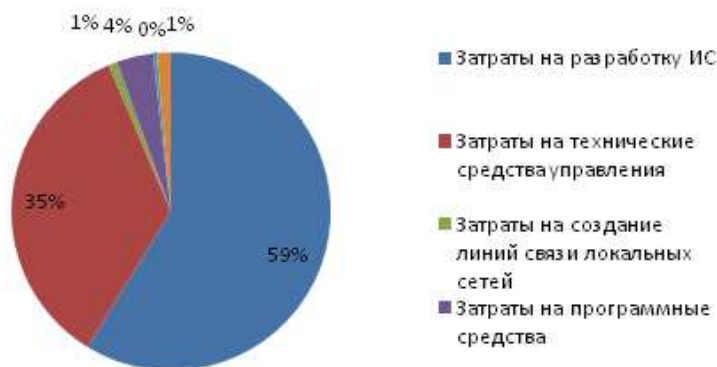


Рисунок 7– Капитальные затраты

3.1.2 Эксплуатационные затраты

Эксплуатационные затраты рассчитываются с помощью формулы 6.

$$C = C_{зп} + C_{ао} + C_{то} + C_{лс} + C_{ни} + C_{проч}, (6)$$

где $C_{зп}$ – зарплата персонала, работающего с ИС;

$C_{ао}$ – амортизационные отчисления;

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

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

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

$C_{проч}$ – прочие затраты.

Заработная плата персонала, работающего с ИС. Для того, чтобы поддерживать ИС в рабочем состоянии нужен 1 инженер-программист, который будет заниматься обслуживанием информационной системы и вносить доработки в программный продукт, следовательно дополнительных затрат на заработную плату для данного сотрудника не потребуется. Также, поскольку с данной ИС будут работать другие специалисты и данная работа уже входит в их должностные инструкции (работа по оказанию услуг), а ИС используется для автоматизации данного процесса, дополнительных затрат на заработную плату данных работников тоже не потребуется.

$$C_{зп} = 0 \text{ руб.}$$

Затраты на амортизационные отчисления. Для использования ИС необходим компьютер, указанный в таблице 2.

Средний срок эксплуатации всего компьютера равен:

$$T_{эк} = \frac{5+4+5+5+3+4}{6} = 4,3, \quad (7)$$

где $T_{эк}$ – срок эксплуатации компьютера.

Следовательно, норма амортизации равна 23% по формуле:

$$N_{\text{ам}} = \frac{100\%}{T_{\text{эк}}} \quad (8)$$

$$A_{\text{год}} = C_{\text{б}} * N_{\text{ам}} = 55433 * 0,23 = 12749 \text{ рублей}$$

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

$C_{\text{б}}$ – балансовая стоимость (55433 руб.);

$N_{\text{ам}}$ – норма амортизации.

Итого: $C_{\text{ао}}=12749$ рублей.

Затраты на техническое обслуживание. Для того, чтобы поддерживать ИС в рабочей форме необходим компьютер, который приобретен в процессе разработки вместе с сервером, а следовательно дополнительных затрат не потребуется.

$C_{\text{то}}=0$ рублей

Затраты на использование глобальных сетей. Для функционирования системы не потребуется постоянного соединения с глобальной сетью, а значит данные затраты будут равны 0 рублей.

$C_{\text{лс}} = 0$ рублей.

Затраты на носители информации. Для данной ИС внешние носители информации не нужны.

$C_{\text{ни}}=0$ рублей

Прочие затраты. На прочие затраты и издержки были взяты финансы в размере 5 % от общей стоимости эксплуатации системы.

$C_{\text{проч}} = 0$ рублей

Таблица 8 – Эксплуатационные затраты

Эксплуатационные затраты	Сумма, руб.
Зарплата персонала	0
Амортизационные отчисления	12749
Затраты на техническое обслуживание	0
Затраты на использование глобальных сетей	0
Затраты на носители информации	0
Прочие затраты	638
Итого	13387

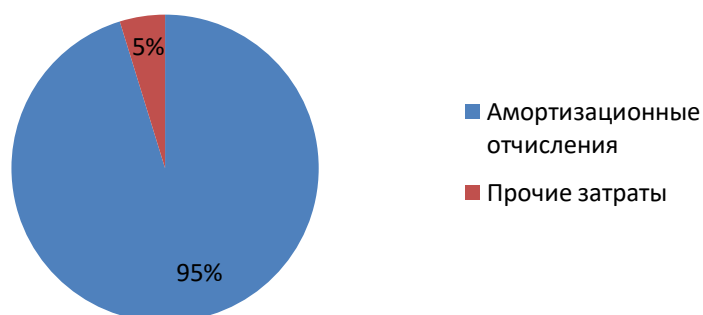


Рисунок 8 – Эксплуатационные затраты

3.2 Расчет по методике TCO

Для расчета совокупности затрат на внедрение проекта необходимо использовать методику, которая могла бы собрать все данные вместе и предоставлять возможность для изучения рентабельности внедрения. Такой является популярная методика TCO (*Total cost of ownership*) – совокупная стоимость владения. Данный метод предполагает количественную оценку на внедрение и сопровождение программного обеспечения. Остальные методики были отброшены ввиду их сложности применения к данному проекту (методика выбора, ELECTRE).

Расчет совокупной стоимости владения ИС.

Показатель совокупной стоимости владения ИС рассчитывается по формуле 9.

$$TCO = DE + IC_1 + IC_2, (9)$$

где DE (direct expenses) – прямые расходы;

IC₁ (indirect costs) – косвенные расходы первой группы;

IC₂ (indirect costs) – косвенные расходы второй группы.

Для данного проекта значимыми являются только косвенные расходы второй группы, следовательно, $TCO \approx DE + IC_2$.

$$DE = DE_1 + DE_2 + DE_3 + DE_4 + DE_5 + DE_6 + DE_7 + DE_8, (10)$$

где DE₁ – капитальные затраты;

DE₂ – расходы на управление информационными технологиями;

DE₃ – расходы на техническую поддержку автоматизированного обеспечения и программного обеспечения;

DE₄ – расходы на разработку прикладного программного обеспечения внутренними силами;

DE₅ – расходы на аутсорсинг;

DE₆ – командировочные расходы;

DE₇ – расходы на услуги связи;

DE₈ – другие группы расходов.

Собрав все данные вместе получим:

– $DE1 = K = 164045$ руб.;

– $DE2 = C_{зп} = 0$ руб.;

– $DE3 = C_{то} + C_{ао} = 0 + 12749 = 12749$ руб.;

– $DE4 = 0$ руб. ;

– $DE5 = 0$ руб., т.к. для данной ИС не требуются внешние источники;

– $DE6 = 0$ руб., т.к. для разработки ИС командировки не потребуются;

– $DE7 = C_{лс} = 0$ руб.;

– $DE8 = C_{проч} + C_{ни} = 638 + 0 = 638$ руб.

Прямые затраты равны:

$$DE = 164045 + 12749 + 0 + 0 + 0 + 0 + 0 + 638 = 177432 \text{ руб.}$$

ИС₂ для разрабатываемой системы будет включать средства на дополнительную разработку программиста внедренной системы после опытной эксплуатации и затраты на рекламу информационной системы.

Реклама информационной системы будет стоить 20000 рублей, при условии рекламы с помощью сети интернет (контекстная реклама).

Дополнительные затраты на внесение правок со стороны программиста во время опытной эксплуатации при условии, что правки будут внесены не в архитектуру проекта, а в условия предоставления информации и внешнего вида приложения будут равняться 0 рублей, поскольку данное условие отражено в техническом задании на разработку информационной системы.

Только при условии серьезных изменений в проекте может потребоваться оплата услуг программиста, но на данном этапе можно считать, что таких правок не потребуется.

$$IC_2=20000+0=20000 \text{ руб.}$$

$$ТСО =197432 \text{ руб.}$$

3.3 Оценка рисков при реализации проекта создания ИС

Для оценки рисков при реализации проекта необходимо провести анализ, идентификацию рисков, разбить обнаруженные риски по приоритету их воздействия на данную ИС и определить мероприятия по разрешению рисков или сведению их влияния к минимуму.

3.3.1 Идентификация рисков

Рассмотрев проект и подобные ИС, были идентифицированы следующие риски:

- Риск денежных потоков.
- Лимитированное время разработки.
- Изменение требований заказчика к проекту.
- Реализационный риск.
- Человеческий фактор.

Данные риски имеют место быть в данной ИС, так как являются обобщенными для всех ИС на этапе разработки проектов.

3.3.2 Анализ рисков

После проведения идентификации необходимо более подробно рассмотреть каждый из рисков, для их сортировки по приоритету влияния на данный проект.

Риск денежных потоков – этот риск может вызвать задержку разработки информационной системы, т.к. заказчиком является государственное учреждение, и имеют место неоплаты проекта своевременно.

Лимитированное время разработки – поскольку может потребоваться изменить проект под другие платформы или ПО, то это может повлечь увеличение времени на разработку проекта и данный риск присутствует в данном проекте.

Изменение требований заказчика к проекту – из-за того, что прямых аналогов система не имеет и может потребоваться адаптация системы и изменение требований при опытной эксплуатации. Могут также измениться требования заказчика к разрабатываемому проекту и, как следствие возникает данный риск.

Реализационный риск присутствует, так как ИС предполагает одновременную работу с данной ИС нескольких пользователей по различным каналам связи (десктопное и мобильное приложение). При этом присутствует вероятность того, что бюджетные учреждения, в которых будет эксплуатироваться система, могут перейти на ПО Российской разработки или же на Unix-системы, что может привести к нерентабельности данной ИС, ввиду слишком кардинальных изменений в структуре работы серверов и клиентского приложения.

Дополнительные расходы на доработку проекта – поскольку за время проекта могут возникнуть удорожание ПО или же компьютерного оборудования, необходимо также учитывать данный риск, поскольку дополнительного финансирования от госучреждения можно не дожидаться.

"Человеческий фактор" – этот риск имеет место, так как эксплуатация ИС предполагает работу персонала и граждан, обращающихся в учреждения за получением услуги, и ошибки при работе в системе пользователей разных уровней могут быть.

3.3.3 Планирование риск-менеджмента и разрешение рисков

Исходя из предыдущих пунктов, были определены риски, которые могут повлиять на проект, их список указан в таблице 9.

Также риски были разбиты на следующие категории:

- Риски инвестирования в разработку проекта.
- Риски внедрения проекта.

К первой категории относятся риски:

- Риск денежных потоков.
- Лимитированное время разработки.
- Изменение требований заказчика к проекту.

Ко второй категории относятся:

- Реализационный риск.
- "Человеческий фактор".

Далее были определены уровни влияния на проект и вероятность возникновения каждого риска, возможности разрешения рисков или снижения их влияния, результаты по данной работе представлены в таблице 9.

Таблица 9 – Риски при создании проекта

№	Группы рисков	Перечень рисков проекта	Уровень влияния риска на проект	Вероятность риска	Возможность предотвращения или снижения риска
1		Лимитированное время разработки	Низкий	Низкая	На тестирование программы выделена треть

Окончание таблицы 9

					времени разработки, это является так называемой «временной подушкой», которая позволит избежать этого риска
		Изменение требований заказчика к проекту	Средний	Низкий	Правильно составленное техническое задание
2	Риски внедрения проекта	Риск не востребоваемости проекта у получателей услуг	Высокий	Средняя	Во избежание этого риска будет произведена реклама информационной системы для привлечения туристов
		"Человеческий фактор"	Средний	Средний	С ростом активности в ИС необходимо наращивать количество специалистов, которые будут подключены к данной системе и позволят обслуживать большее количество граждан.

Реализационный риск является наиболее влиятельным на проект, так как от него напрямую зависит, будет ли ИС ликвидная, и оправдает ли средства и ресурсы, необходимые для ее реализации, ведь в случае ее провала придется производить разработку новой ИС под другие среды или же полностью отказаться от использования ИС.

3.4 Анализ рынка продуктов – аналогов. Установление стоимости программного продукта

Проведен анализ аналогичных программных продуктов, которые представлены на рынке и имеют похожие технические характеристики и функции (Таблица 10).

Таблица 10 – Сравнение продуктов – аналогов

	PeakFinder	OsmAnd+	Гид по Хакасии	Наша система
Дизайн	4	3	3	4
Средняя оценка	4,5	4,7	4	4
Возможность онлайн видеозаписей	0	0	0	4
Возможность чата с гидами	0	0	0	4
Стоимость создания	154100	147700	183574	168228
Стоимость	289	1250	Бесплатно	Бесплатно

Проанализировав таблицу, следует сделать вывод, что разработанный продукт находится на разных уровнях с конкурентами в основных характеристиках, поскольку у него новый функционал для данной сферы и другая применяемость.

Для определения стоимости продукта проверим разные методы установления стоимости:

1. Затратный метод.

Определяется на основании капитальных затрат (К) определяется себестоимость, количества экземпляров, а также прибавление планируемой прибыли (10% от себестоимости), стоимость рекламы и продвижения продукта.

$$\text{Стоимость} = \text{К} + \text{К} * 10\% + \text{стоимость продвижения}$$

Для мобильного приложения стоимость рекламы и продвижения примерно равна 8000 рублей.

Итоговая стоимость программного продукта равна:

Стоимость = $160975 + 160975 * 0,1 + 8000 = 185\ 072,5$ руб.

2. Метод сравнения.

Разработанный продукт находится на разных уровнях с конкурентами в основных характеристиках, поскольку у него новый функционал для данной сферы и другая применимость, поэтому ему можно указать стоимость равную бесплатной, но с возможностью приобретения платных покупок внутри приложения, для получения прибыли.

Стоимость создания всех приложений примерно на одном уровне, следовательно, и создаваемое приложение необходимо ставить ценой примерно на этом же уровне. То есть определить стоимость по затратному методу, чтобы оправдать потраченные средства на разработку и внедрение игрового приложения.

3. Доходный метод.

Увеличение стоимости значительно выше себестоимости нецелесообразно из-за возможной низкой окупаемости приложения в будущем, следовательно, стоимость ставится на уровне двух предыдущих пунктов.

3.5 Источник получения дохода

Источником получения дохода от данного приложения будут непосредственно онлайн видеоэкскурсии, которые будут покупаться конечными пользователями. Для расчета конечной стоимости данной экскурсии необходим анализ существующих на данный момент аналогичных экскурсий.

Основная проблема, что на данный момент, отсутствие экскурсий в онлайн формате как таковых, поскольку данное приложения является

первооткрывателем в данной сфере. Следовательно, будут анализироваться экскурсии в «живом формате», которые проводятся непосредственно сейчас:

– Экскурсия "Саянская мозаика" - Маршрут: Саяно-Шушенская ГЭС + Мраморная гора + Красное озеро + Фуникулер на вершину горы + Набережная Енисея - Стоимость 1500 рублей;

– Ретро экскурсия Минусинск+Шушенское - Стоимость 1500 рублей;

– Экскурсия по Саяно-Шушенскому водохранилищу - Стоимость 1100 рублей (без учета доставки к месту проведения);

– Экскурсия в горы Ергаки - "Висячий камень" - Стоимость 1500 рублей;

– Экскурсия на скальный массив Тогыс–Ас "ТРОПА ПРЕДКОВ" - Стоимость 1700 рублей;

– Экскурсия на ОГЛАХТЫ "Галерея древних рисунков" - Стоимость 1100 рублей [9].

Как видно из данных, представленных выше, стоимость обычных экскурсий находится на уровне 1500 рублей, с небольшими отклонениями. Для привлечения пользователей к данной разработке следует на начальных этапах выхода приложения «в свет» сделать цены ниже рыночных на 30%, что позволит приложению быстро выйти в свет и привлечь пользователей. Далее, по мере нарастания общего количества пользователей можно варьировать ценовую политику по желанию. Следовательно, если стоимость экскурсии составляет 1000 рублей, стоимость для приложения будет равна 700 руб. и так далее.

3.6 Экономическая эффективность реализации проекта

Внедрение информационных технологий сопряжено с капитальными вложениями как на приобретение техники, так и на разработку проектов, выполнение подготовительных работ и подготовку кадров.

Под эффективностью автоматизированного преобразования экономической информации понимают целесообразность применения средств вычислительной и организационной техники при формировании, передаче и обработке данных. Различают расчетную и фактическую эффективность. Первую (расчетную) определяют на стадии проектирования автоматизации информационных работ, т. е. разработки рабочего проекта, вторую (фактическую) – по результатам внедрения рабочего проекта.

Обобщенным критерием экономической эффективности является минимум затрат живого и общественного труда. При этом установлено, что чем больше участков управленческих работ автоматизировано, тем эффективнее используется техническое и программное обеспечение.

Экономический эффект от внедрения вычислительной и организационной техники подразделяют на прямой и косвенный.

Под прямой экономической эффективностью программного продукта понимают экономию материально-трудовых ресурсов и денежных средств, полученную в результате сокращения численности управленческого персонала, фонда заработной платы, расхода основных и вспомогательных материалов вследствие автоматизации конкретных видов планово-учетных и аналитических работ.

Под косвенной экономической эффективностью программного продукта понимают экономию времени и затрат человеческих ресурсов на выполнение определенной задачи, то есть не конкретно ощутимые денежные средства, а увеличение скорости работы предприятия и уменьшение трудозатрат.

Данное приложение будет приносить прямой доход, поскольку внутри него будут продаваться экскурсии за определенную плату, следовательно, денежные средства будут поступать на счет Заказчика.

Для расчета экономической эффективности необходимо рассчитать через какое количество экскурсий будут покрыты расходы, затраченные на создание приложения.

Средняя стоимость экскурсий равна 1500, следовательно, для вычисления стоимости экскурсии в приложении необходимо $1500 \cdot 30\% = 450$ рублей. Таким образом, средняя стоимость экскурсий при использовании данного приложения равна 1000 рублей.

Сделаем два прогноза на расчет эффективности данного приложения.

Оптимистический прогноз: в среднем проводится 70 экскурсий в месяц, следовательно, для вычисления среднего количества заработных средств за месяц необходимо, $1000 \cdot 70 = 70000$ рублей. Так как стоимость программы равна 145378, следовательно, данная программа окупится за $145378 / 70000 = 2,07$ месяцев. Таким образом, получается, что через 2,1 месяца данное приложение выйдет в плюс, окупит все расходы и начнет приносить доход. Таким образом, на третьем месяце эксплуатации приложение окупится, поскольку месяц необходим на рекламу приложения начала пользования приложением.

Пессимистичный прогноз: в среднем проводится 10 экскурсий в месяц, следовательно, для вычисления среднего количества заработных средств за месяц необходимо, $1000 \cdot 10 = 10000$ рублей. Так как стоимость программы равна 145378, следовательно, данная программа окупится за $145378 / 10000 = 14,53$ месяца. Таким образом, получается, что через 14,6 месяцев данное приложение выйдет в плюс, окупит все расходы и начнет приносить доход. Таким образом, через год с небольшим эксплуатации приложение окупится, поскольку месяц необходим на рекламу приложения начала пользования приложением. Так как сезон экскурсионных услуг длится с апреля по октябрь, следовательно, во время второго сезона данное приложение начнет приносить доход.

3.7 Выводы по экономическому разделу

В данном разделе были оценены капитальные затраты, которые составляют 164 045 (руб.). Эксплуатационные затраты, которые составляют 13 387 (руб.). Данные сведения позволили рассчитать совокупную стоимость владения информационной системой (ТСО), сумма которого составляет 197 432 (руб.). Помимо этого были предугаданы наиболее вероятные риски и минимизировано их влияние на данный проект. Проведена оценка экономической эффективности внедрения проекта.

ЗАКЛЮЧЕНИЕ

Цель выпускной квалификационной работы – разработать информационную систему организации онлайн экскурсий на объектах культурного наследия Республики Хакасия, достигнута. Поставленные задачи выполнены в полном объёме.

Изучив сферу разработки веб-приложений и изучив место внедрения программного продукта, были выбраны следующие программные средства:

- Python.
- Flask.
- Firebase.

Выбранные программные средства являются бесплатными. Также их внедрение будет значительно комфортнее, так как хранение данных будет осуществляться в облачном хранилище.

Результатом проделанной работы является информационная система, которая может использоваться для оказания удалённых экскурсий на объектах культурного наследия Республики Хакасия с использованием видеосвязи, для осуществления мониторинга сотрудниками Министерства культуры оказания платных экскурсионных услуг в данной системе и для популяризации туристического потенциала Хакасии в сети «интернет»

Кроме создания информационной системы была выполнена оценка экономической эффективности, оценены риски. Затраты на разработку проекта составляют 164045 руб. Совокупная стоимость владения системой, включающая затраты на создание системы, внедрение и эксплуатацию составляют 197432 руб.

Информационная система будет использоваться учреждениями культуры, подведомственными Министерству культуры Республики Хакасия для оказания удалённых экскурсионных услуг.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Сайт Министерства культуры Республики Хакасия [Электронный ресурс]. – Режим доступа: <https://www.culture19.ru/>
2. Разработка Web-приложений [Электронный ресурс]. – Режим доступа: <http://en.bookfi.net/book/632117>
3. Базы данных. Проектирование, реализация и сопровождение [Электронный ресурс]. – Режим доступа: <https://www.studmed.ru/>
4. Кутикова, К. В. Методика проектирования информационных систем для сферы государственных и муниципальных услуг / К.В. Кутикова. – М.: Синергия, 2014. – 92 с.
5. Разработка Web-приложений [Электронный ресурс]. – Режим доступа: <http://en.bookfi.net/book/632117>
6. ГОСТ 34.602-89. Техническое задание. На создание автоматизированной системы [Электронный ресурс]. – Режим доступа: <https://studfiles.net/preview/5711663/>
7. Контрольный список тестирования веб-приложений: примеры тестовых случаев для веб-сайта [Электронный ресурс] : Онлайн система обучения Guru99. – Режим доступа: <https://www.guru99.com/complete-web-application-testing-checklist.html>
8. Государственные стандарты серии 34 [Электронный ресурс]. – Режим доступа: <https://www.prj-exp.ru/gost-34>
9. Федеральный закон от 27.07.2006 N 149-ФЗ (ред. от 19.07.2018) "Об информации, информационных технологиях и о защите информации".
10. Положение о Министерстве культуры Республики Хакасия: Постановление Правительства Республики Хакасия от 23 октября 2011 г. № 260 (в ред. от 07.03.2018 N 168) // Вестник Хакасии. – 2019. – 23 июня.
11. Documentation–Materialize[Электронный ресурс]. – Режим доступа: <https://materializecss.com>

12. Flask – A Framework library for building user interfaces [Электронный ресурс]. – Режим доступа: <https://Flask.org/>
13. Что такое стек MERN, и как с ним работать? [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/piter/blog/458096>
14. Flask – фреймворк веб-приложений Python [Электронный ресурс]. – Режим доступа: <https://flaskapp.com/ru/>
15. The most popular database for modern apps | FireBase[Электронный ресурс]. – Режим доступа: <https://www.firebaseio.com/>
16. Visual Studio Code - Code Editing. Redefined [Электронный ресурс]. – Режим доступа: <https://code.visualstudio.com>.
17. Туристская компания Discovery[Электронный ресурс]. – Режим доступа: <https://www.discovery-khakasia.ru/ekskursii.html>.
18. Выполнение и защита выпускной квалификационной работы по направлению 09.03.03 «Прикладная информатика» [Электронный ресурс] : метод. указания / сост. Е. Н. Скуратенко, В. И. Кокова, И. В. Янченко ; Сиб. федер. ун-т, ХТИ – филиал СФУ. – Электрон.текстовые, граф. дан. (0,71 МБ). – Абакан : ХТИ – филиал СФУ, 2017. – 1 файл. – Режим доступа: https://e.sfu-kras.ru/pluginfile.php/1368122/mod_reso.
19. СТО 4.2–07–2014 Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности – Введ. 02.07.2014. – Красноярск: ИПК СФУ, 2014. – 60 с.
20. Управление IT-проектом. Курсовая работа [Электронный ресурс] : методические указания / Е. Н. Скуратенко, И. В. Янченко, В. И. Кокова ; Сиб. федер. ун-т; ХТИ - филиал СФУ. - Электрон.текстовые дан. Электрон.граф. дан. (файла : 0,04 Мбайтов). - Абакан : ХТИ - филиал СФУ, 2018. – Режим доступа:http://89.249.130.59/docs/Met_1082.pdf.

ПРИЛОЖЕНИЕ А

Листинг кода веб – приложения для оказания онлайн-экскурсий на объектах культурного наследия Республики Хакасия

Компонент app.py

```
from pyrebase import pyrebase
from functools import wraps
import sys
import os
from flask import *
from urllib.parse import quote
pyrebase.quote = lambda s, safe=None: s
def get_url(self, token=None):
    path = self.path
    self.path = None
    if path.startswith('/'):
        path = path[1:]
    if token:
        return
    "{0}/o/{1}?alt=media&token={2}".format(self.storage_bucket, quote(path, safe=""), token)
    return
    "{0}/o/{1}?alt=media".format(self.storage_bucket, quote(path, safe=""))
    pyrebase.Storage.get_url = lambda self, token=None: \
        get_url(self, token)
    app = Flask(__name__)
    app.secret_key = os.urandom(24)
    config = {
        "apiKey":
            "AIzaSyADkHUU_u6uTKAVnm7XdkqNn_C10vpSYGI",
        "authDomain":
            "diplom-5c7ec.firebaseio.com",
        "databaseURL":
            "https://diplom-5c7ec.firebaseio.com",
        "projectId": "diplom-5c7ec",
        "storageBucket":
            "diplom-5c7ec.appspot.com",
        "messagingSenderId": "785695600410",
        "appId":
            "1:785695600410:web:b7c54b97c95030ec5a06a3",
        "measurementId": "G-Z45BH4EJSE"
    }
    def isAuthenticated(f):
        @wraps(f)
        def decorated_function(*args, **kwargs):
            #check for the variable that pyrebase
            creates
            if not auth.current_user != None:
                return redirect(url_for('signup'))
            return f(*args, **kwargs)

        return decorated_function
    firebase = pyrebase.initialize_app(config)
    auth = firebase.auth()
    db = firebase.database();
    users = db.child("Posts").get()

    @app.route("/", methods= ['GET','POST'])
    def login():
        if request.method == "POST":
            #get the request data
            email = request.form["email"]
            password = request.form["password"]
            try:
                #login the user
                user =
                auth.sign_in_with_email_and_password(email, password)
                #set the session
                user_id = user['idToken']
                user_email = email
                session['usr'] = user_id
                session["email"] = user_email
                return redirect("/index")
            except:
                return render_template("login.html", message="1" )

        return render_template("login.html")
    @app.route("/logout")
    def logout():
        #remove the token setting the user to
        None
        auth.current_user = None
        #also remove the session
        #session['usr'] = ""
        #session["email"] = ""
        session.clear()
        return redirect("/");

    @app.route("/home", methods=
    ['GET','POST'])
    @app.route("/index", methods=
    ['GET','POST'])
    def main():
```



```

        allposts =
db.child("Excursions_User").child("YesPassed").child("1IhvYOLINbfkcA8dlnxfqmaksG73").get()
        gidpost =
db.child("Excursions_User").child("NoPassed").child("1IhvYOLINbfkcA8dlnxfqmaksG73").get()
        return render_template('index.html',
gid=gidpost,posts=allposts)
        @app.route("/contact")
def contact():
        return
render_template('contact.html')
        @app.route("/category")
def category():
        return
render_template('category.html')
        @app.route("/about")
def about():
        return render_template('about.html')
        @app.route("/blog_single")
def blog_single():
        return render_template('blog-
single.html')

#create form
@app.route("/create", methods=["GET",
"POST"])
@isAuthenticated
def create():

if request.method == "POST":
        #get the request data
title = request.form["title"]
content = request.form["content"]

post = {
        "title": title,
        "content": content,
        "author": session["email"]
        }

try:
        #print(title, content, file=sys.stderr)

        #push the post object to the database
db.child("Posts").push(post)
return redirect("/")
except:
        return render_template("create.html",
message= "Ошибка создания записи")

return render_template("create.html")

@app.route("/post/<id>")
@isAuthenticated
def post(id):

```

```

        orderedDict =
db.child("Posts").order_by_key().equal_to(id).limit_t
o_first(1).get()
        print(orderedDict)

        return render_template("post.html",
data=orderedDict)

@app.route("/edit/<id>", methods=["GET",
"POST"])
def edit(id):
if request.method == "POST":

title = request.form["title"]
content = request.form["content"]

post = {
        "title": title,
        "content": content,
        "author": session["email"]
        }
        #update the post
db.child("Posts").child(id).update(post)
return redirect("/post/" + id)
        #update the post
db.child("Posts").child(id).update(post)
return redirect("index.html")
return redirect("index.html")
        @app.route("/delete/<id>",
methods=["POST"])
def delete(id):
db.child("Posts").child(id).remove()
return redirect("/")

```

Компонент index.html

```

<thead>
<tr>
<th scope="col">№</th>
<th scope="col">Экскурсия</th>
<th scope="col">Дата</th>
<th scope="col">Время</th>
<thscope="col">Дополнительная
информация</th>
<th scope="col">Работа с экскурсией</th>
</tr>
</thead>
<tbody>
        {% for post in posts.each() %}

```

```

                {% if session["email"] ==
post.val().author %}
                <tr>
                <th
scope="row">{{ post.val().number }}</th>
                <td>{{ post.val().name }}</td>
                <td>{{ post.val().date }}</td>
                <td>{{ post.val().time }}</td>
                <td>{{ post.val().micro_about }}</td>
                <td><a
href="/status/{{ post.key() }}">готова к
проведению</a>
                <p></p>
                <a href="/status/{{ post.key() }}">требуется
уточнение</a>
                <p></p>
                {% endif %}
                {% endfor %}
            </tbody>
        </table>
        <h2>Доступные заказы</h2>
        <table class="table table-striped table-
hover">
        <thead>
        <tr>
        <th scope="col">№</th>
        <th scope="col">Экскурсия</th>
        <th scope="col">Статус</th>
        <th scope="col">Дата</th>
        <th scope="col">Время</th>
        <th scope="col">Дополнительная
информация</th>
        <tr>
        <th scope="col">Взять в работу</th>
        </tr>
        </thead>
        <tbody>
        {% for post in gid.each() %}
        {% if post.val().statusExcursion ==
"заявка отправлена" %}
        <tr>
        <th
scope="row">{{ post.val().number }}</th>
        <td>{{ post.val().name }}</td>
        <td>{{ post.val().statusExcursion }}</td>
        <td>{{ post.val().date }}</td>
        <td>{{ post.val().time }}</td>
        <td>{{ post.val().micro_about }}</td>
        <td><a href="/status/{{ post.key() }}">Взять
в работу</a></td>
        </tr>
        {% endif %}
        {% endfor %}
        </tbody>
        </table>
        <h2 class="mb-4">Новости</h2>
        <div class="container">
        <div class="row">
        <div class="col-md-12">
        <div class="owl-carousel owl-theme home-
slider">
        <div>

```

```

                <a href="blog-single" class="a-block d-flex
align-items-center height-lg" style="background-
image:url('{{ url_for('static',
filename='images/img_1.jpg') }}')">
                <div class="text half-to-full">
                <div class="post-meta">
                <span class="category">Lifestyle</span>
                <span class="mr-2">Март 15, 2021
</span>&bullet;
                <span class="ml-2"><span class="fa fa-
comments"></span> 0</span>
                </div>
                <h3>Обзорная экскурсия по музею
"Казановка"</h3>
                <p>{{ post.content.key }}</p>
                </div>
                </a>
            </div>
            <a href="blog-single" class="a-block d-flex
align-items-center height-lg" style="background-
image:url('{{ url_for('static',
filename='images/{{ post.id.key }}.jpg') }}')">
            <div class="text half-to-full">
            <div class="post-meta">
            <span class="category">Lifestyle</span>
            <span class="mr-2">March 15, 2018
</span>&bullet;
            <span class="ml-2"><span class="fa fa-
comments"></span> 3</span>
            </div>
            <h3>Обзорная экскурсия по музею
"Казановка"</h3>
            <p>{{ post.content.key }}</p>
            </div>
            </a>
        </div>
        <a href="blog-single" class="a-block d-flex
align-items-center height-lg" style="background-
image:url('{{ url_for('static',
filename='images/{{ post.id.key }}.jpg') }}')">
        <div class="text half-to-full">
        <div class="post-meta">
        <span class="category">Lifestyle</span>
        <span
class="mr-2">{{ post.date.key }}</span>&bullet;
        <span class="ml-2"><span class="fa fa-
comments"></span> 3</span>
        </div>
        <h3></h3>
        <p></p>
        </div>
        </a>
    </div>
</div>
<div class="row">
<div class="col-md-6 col-lg-4">

```



```

cy="24" r="22" fill="none" stroke-width="4"
stroke="#e6e6e6"/><circle class="path" cx="24"
cy="24" r="22" fill="none" stroke-width="4" stroke-
miterlimit="10" stroke="#f4b214"/></svg></div>
</body>
</html>

```

Компонент conference.js

```

bc.joinRoom( "conference_gid" );
    });
    bc.signaling.bind(
"onJoinedRoom", function ( result ) {
        var roomMembers =
result.members;
        for ( var i=0,
max=roomMembers.length; i<max; i++ ) {
            bc.call(
roomMembers[i].id, "conference_gid", { stream:
localStream, 'video-codec': 'H264/9000' } );
        }
    });
    bc.streams.bind( "onStreamAdded",
function ( remoteStream ) {
        bc.attachStream(
remoteStream, document.body );
    });
    bc.streams.bind(
"onStreamClosed", function ( stream ) {
        bc.detachStream( stream );
    });
    bc.connect();
}

```

Компонент Style.css

```

body {
background: #fff;
font-family: "Josefin Sans", arial, sans-serif;
font-weight: 300;
font-size: 18px;
line-height: 1.9;
color: #6c757d;
}

a {
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
text-decoration: none;
}

a:hover {
text-decoration: none;
}

h1, h2, h3, h4, h5 {
color: #000;
font-family: "Josefin Sans", arial, sans-serif;
}

header {
position: relative;
top: 0;

```

```

left: 0;
width: 100%;
z-index: 5;
font-weight: 400;
background: #fff !important;
}

```

```

header .navbar-brand {
text-transform: uppercase;
letter-spacing: .2em;
font-weight: 400;
color: #fff !important;
}

```

```

header .navbar-collapse {
position: relative;
}

```

```

header .navbar {
padding-top: 0;
padding-bottom: 0;
background: transparent !important;
}

```

```

@media (max-width: 767.98px) {
header .navbar {
background: #e6e6e6 !important;
}
}

```

```

header .navbar .navbar-nav.absolute-right {
position: absolute;
right: 0;
}

```

```

@media (max-width: 991.98px) {
header .navbar .navbar-nav.absolute-right {
position: relative;
right: inherit;
}
}

```

```

header .navbar .nav-link {
padding: 1.7rem 1rem;
font-size: 13px;
outline: none !important;
text-transform: uppercase;
letter-spacing: .05em;
}

```

```

@media (max-width: 991.98px) {
header .navbar .nav-link {
padding: 1.7rem 1rem;
}
}

```

```

@media (max-width: 767.98px) {
header .navbar .nav-link {
padding: .5rem 0rem;
}
}

```

```

header .navbar .dropdown-menu {
font-size: 14px;
border-radius: 0px;
border: none;
-webkit-box-shadow: 0 2px 30px 0px
rgba(0, 0, 0, 0.2);
box-shadow: 0 2px 30px 0px rgba(0, 0, 0,
0.2);
min-width: 13em;
margin-top: -10px;
}

header .navbar .dropdown-menu:before {
bottom: 100%;
left: 10%;
border: solid transparent;
content: " ";
height: 0;
width: 0;
position: absolute;
pointer-events: none;
border-bottom-color: #fff;
border-width: 10px;
}

@media (max-width: 1199.98px) {
header .navbar .dropdown-menu:before {
display: none;
}
}

header .navbar .dropdown-menu .dropdown-
item:hover {
background: #007bff;
color: #fff;
}

header .navbar .dropdown-menu .dropdown-
item.active {
background: #007bff;
color: #fff;
}

header .navbar .dropdown-menu a {
padding-top: 7px;
padding-bottom: 7px;
}

.absolute-toggle {
position: absolute;
left: 15px;
top: 5px;
}

.absolute-toggle .burger-lines {
width: 30px;
display: inline-block;
height: 2px;
background: #000;
position: relative;
}

.absolute-toggle .burger-lines:before,
.absolute-toggle .burger-lines:after {
position: absolute;
height: 2px;
content: "";
background: #000;
width: 100%;
left: 0;
}

.absolute-toggle .burger-lines:before {
top: -10px;
}

.absolute-toggle .burger-lines:after {
bottom: -10px;
}

.site-logo a {
color: #000;
font-size: 90px;
}

@media (max-width: 991.98px) {
.site-logo a {
font-size: 40px;
}
}

.top-bar {
background: #d1a341;
padding: 10px 0;
}

.top-bar .social a, .top-bar .search-icon a {
color: #fff;
opacity: .5;
padding: 5px;
}

.top-bar .social a:hover, .top-bar .search-icon
a:hover {
opacity: 1;
}

.top-bar .social {
text-align: left;
}

.top-bar .search-icon {
text-align: right;
}

.navbar > .container {
border-bottom: 1px solid #e6e6e6;
}

@media (max-width: 767.98px) {
.navbar > .container {
border-bottom: none;
}
}

```

```

.category {
display: inline-block;
background: #007bff;
padding: 2px 8px;
line-height: 1.5;
font-size: 12px;
border-radius: 4px;
text-transform: uppercase;
color: #fff !important;
margin-right: 10px;
}

.a-block {
display: block;
background-size: cover;
background-position: center center;
padding: 30px;
position: relative;
margin-bottom: 30px;
height: 300px;
}

.a-block.height-lg {
height: 500px;
}

.a-block.height-md {
height: 400px;
}

.a-block:before {
background: #000;
position: absolute;
content: "";
z-index: 1;
opacity: .3;
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
top: 0;
left: 0;
right: 0;
bottom: 0;
}

.a-block .text {
position: relative;
z-index: 2;
max-width: 100%;
}

.a-block .text.half-to-full {
max-width: 50%;
}

@media (max-width: 767.98px) {
.a-block .text.half-to-full {
max-width: 100%;
}
}

.a-block .text .post-meta {
color: #fff;
text-transform: uppercase;
letter-spacing: .1em;
font-size: 13px;
margin-bottom: 30px;
}

.a-block .text h3 {
color: #fff;
border: 1px solid red;
}

.a-block .text p {
color: #fff;
}

.a-block:hover:before {
opacity: .4;
}

.bio {
padding: 15px;
background: #fff;
border: 1px solid #e6e6e6;
font-weight: 400;
}

.bio img {
max-width: 100px;
margin-top: -4em;
border-radius: 50%;
position: relative;
margin-bottom: 30px;
}

.bio h2 {
font-size: 20px;
}

.bio .bio-body .social a {
color: #000;
}

.site-hero {
background-size: cover;
background-position: center center;
min-height: 750px;
height: 100vh;
}

.site-hero a {
color: #fff;
opacity: .5;
}

.site-hero a:hover {
opacity: 1;
}

.site-hero .post-meta {
font-size: 13px;
}

```

```

text-transform: uppercase;
letter-spacing: .2em;
}

.site-hero h1, .site-hero p {
color: #fff;
line-height: 1.5;
}

.site-hero h1 {
text-transform: uppercase;
font-size: 60px;
font-weight: 900;
line-height: 1.2;
margin-bottom: 30px;
}

.site-hero .lead {
font-size: 30px;
color: #fff;
opacity: .8;
margin-bottom: 30px;
font-weight: 300;
}

.site-hero > .container {
position: relative;
z-index: 2;
}

.site-hero.overlay {
position: relative;
}

.site-hero.overlay:before {
content: "";
background: rgba(0, 0, 0, 0.5);
width: 100%;
z-index: 1;
position: absolute;
top: 0;
left: 0;
right: 0;
bottom: 0;
}

.site-hero .site-hero-inner {
min-height: 750px;
height: 100vh;
}

.box {
padding: 30px;
background: #fff;
}

.episodes .episode a {
color: #000;
}

.episodes .episode a:hover {
opacity: .7;
}
}

.episodes .meta {
text-transform: uppercase;
font-size: 14px;
letter-spacing: .2em;
color: #cccccc;
}

.episodes .episode-number {
border: 2px solid #0062cc;
text-align: center;
display: inline-block;
width: 80px;
height: 80px;
text-transform: uppercase;
line-height: 74px;
border-radius: 50%;
background: #007bff;
color: #fff;
font-size: 30px;
}

.site-hero-innerpage, .site-hero-innerpage
.site-hero-inner {
min-height: 550px;
height: 50vh;
}

.site-section {
padding: 7em 0;
}

.site-section.py-md {
padding: 3em 0;
}

.site-section.py-sm {
padding: 0em 0;
}

.school-features,
.school-instructors {
background-size: cover;
background-position: center center;
position: relative;
overflow: hidden;
}

@media (min-width: 576px) {
.school-features,
.school-instructors {
width: 100%;
padding-right: 15px;
padding-left: 15px;
margin-right: auto;
margin-left: auto;
}
}

@media (min-width: 576px) {
.school-features .inner,

```

```

        .school-instructors .inner {
display: -webkit-box;
display: -ms-flexbox;
display: flex;
    -ms-flex-wrap: wrap;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
    }
    }

.school-features .feature, .school-features
.instructor,
.school-instructors .feature,
.school-instructors .instructor {
position: relative;
width: 100%;
min-height: 1px;
padding-right: 15px;
padding-left: 15px;
    -webkit-box-flex: 0;
    -ms-flex: 0 0 100%;
flex: 0 0 100%;
max-width: 100%;
    }

@media (min-width: 576px) {
.school-features .feature, .school-features
.instructor,
.school-instructors .feature,
.school-instructors .instructor {
position: relative;
width: 100%;
min-height: 1px;
padding-right: 15px;
padding-left: 15px;
    -webkit-box-flex: 0;
    -ms-flex: 0 0 50%;
flex: 0 0 50%;
max-width: 50%;
    }
    }

@media (min-width: 768px) {
.school-features .feature, .school-features
.instructor,
.school-instructors .feature,
.school-instructors .instructor {
position: relative;
width: 100%;
min-height: 1px;
padding-right: 15px;
padding-left: 15px;
    -webkit-box-flex: 0;
    -ms-flex: 0 0 50%;
flex: 0 0 50%;
max-width: 50%;
    }
    }

@media (min-width: 992px) {
        .school-features .feature, .school-features
.instructor,
.school-instructors .feature,
.school-instructors .instructor {
position: relative;
width: 100%;
min-height: 1px;
padding-right: 15px;
padding-left: 15px;
    -webkit-box-flex: 0;
    -ms-flex: 0 0 25%;
flex: 0 0 25%;
max-width: 25%;
padding: 30px;
border-right: 1px solid rgba(255, 255, 255,
0.1);
    }
    }

.school-features .feature h3, .school-features
.instructor h3,
.school-instructors .feature h3,
.school-instructors .instructor h3 {
font-size: 18px;
color: #fff;
    }

.school-features .feature p, .school-features
.instructor p,
.school-instructors .feature p,
.school-instructors .instructor p {
color: #fff;
opacity: .5;
    }

.school-features .feature img, .school-
features .instructor img,
.school-instructors .feature img,
.school-instructors .instructor img {
width: 100px;
border-radius: 50%;
margin: 0 auto;
    }

.school-features .feature .icon > span:before,
.school-features .instructor .icon > span:before,
.school-instructors .feature .icon >
span:before,
.school-instructors .instructor .icon >
span:before {
font-size: 50px;
margin-left: 0;
color: #fff;
    }

.school-features.text-dark.last,
.school-instructors.text-dark.last {
border-bottom: none;
    }

.school-features.text-dark.last .feature,
.school-instructors.text-dark.last .feature {

```



```

border-bottom: none;
}

@media (min-width: 992px) {
.school-features.text-dark .feature,
.school-instructors.text-dark .feature {
border-right: 1px solid #e6e6e6;
border-bottom: 1px solid #e6e6e6;
}
}

.school-features.text-dark .feature:last-child,
.school-instructors.text-dark .feature:last-
child {
border-right: none;
}

.school-features.text-dark .feature h3,
.school-instructors.text-dark .feature h3 {
font-size: 18px;
color: #000;
}

.school-features.text-dark .feature p,
.school-instructors.text-dark .feature p {
color: #000;
opacity: .5;
}

.school-features.text-dark .feature p:last-
child,
.school-instructors.text-dark .feature p:last-
child {
margin-bottom: 0;
}

.school-features.text-dark .feature .icon >
span:before,
.school-instructors.text-dark .feature .icon >
span:before {
font-size: 50px;
margin-left: 0;
color: #007bff;
}

.img-bg {
background-size: cover;
}

@media (max-width: 991.98px) {
.img-md-fluid {
max-width: 100%;
}
}

.section-cover {
background-size: cover;
position: relative;
background-position: top left;
}

.section-cover, .section-cover .intro {
border-bottom: none;
}

height: 500px;
}

.section-cover p {
color: #fff;
}

.section-cover h2 {
color: #fff;
}

.blog-entries .blog-entry {
display: block;
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
margin-bottom: 30px;
}

.blog-entries .blog-entry:hover, .blog-entries
.blog-entry:focus {
opacity: .7;
}

.blog-entries .blog-entry .blog-content-body
{
padding: 20px;
border: 1px solid #efefef;
border-top: none;
}

.blog-entries .blog-entry img {
max-width: 100%;
}

.blog-entries .blog-entry h2 {
font-size: 18px;
line-height: 1.5;
}

.blog-entries .blog-entry p {
font-size: 13px;
color: gray;
}

.blog-entries .post-meta {
margin-bottom: 20px;
color: #b3b3b3;
}

.instructor-meta {
margin-bottom: 10px;
color: #999999 !important;
}

.btn, .form-control {
outline: none;
-webkit-box-shadow: none !important;
box-shadow: none !important;
border-radius: 0;
}

```

```

        .btn:focus, .btn:active, .form-control:focus,
.form-control:active {
    outline: none;
}

.form-control {
    -webkit-box-shadow: none !important;
    box-shadow: none !important;
}

textarea.form-control {
    height: inherit;
}

.btn {
    -webkit-transition: .3s all ease;
    -o-transition: .3s all ease;
    transition: .3s all ease;
    padding: 8px 20px;
}

.btn.btn-black {
    background: #000;
    color: #fff;
}

.btn.btn-primary {
    color: #fff;
    border-width: 2px;
}

.btn.btn-primary:hover, .btn.btn-
primary:active, .btn.btn-primary:focus {
    border-color: #3395ff;
    background: #3395ff;
}

.btn.btn-sm {
    font-size: 14px;
}

.btn.btn-outline-primary {
    border-width: 2px;
    color: #007bff;
}

.btn.btn-outline-primary:hover, .btn.btn-
outline-primary:focus, .btn.btn-outline-primary:active {
    color: #fff;
}

.btn.btn-outline-white {
    border-width: 2px;
    border-color: #fff;
    color: #fff;
}

.btn.btn-outline-white:hover, .btn.btn-
outline-white:focus {
    background: #fff;
    color: #000;
}

border-width: 2px;
}

.btn:hover {
    -webkit-box-shadow: 0 3px 10px -2px
rgba(0, 0, 0, 0.2) !important;
    box-shadow: 0 3px 10px -2px rgba(0, 0, 0,
0.2) !important;
}

.flaticon-custom {
    font-size: 70px;
}

.flaticon-custom:before, .flaticon-
custom:after {
    margin-left: 0;
}

.no-nav .owl-nav {
    display: none;
}

.owl-carousel .owl-item {
    opacity: .4;
}

.owl-carousel .owl-item.active {
    opacity: 1;
}

.owl-carousel .owl-nav {
    position: absolute;
    top: 50%;
    width: 100%;
}

.owl-carousel .owl-nav .owl-prev,
.owl-carousel .owl-nav .owl-next {
    position: absolute;
    -webkit-transform: translateY(-50%);
    -ms-transform: translateY(-50%);
    transform: translateY(-50%);
    margin-top: -10px;
}

.owl-carousel .owl-nav .owl-prev:hover,
.owl-carousel .owl-nav .owl-prev:focus, .owl-
carousel .owl-nav .owl-prev:active,
.owl-carousel .owl-nav .owl-next:hover,
.owl-carousel .owl-nav .owl-next:focus,
.owl-carousel .owl-nav .owl-next:active {
    outline: none;
}

.owl-carousel .owl-nav .owl-prev
span:before,
.owl-carousel .owl-nav .owl-next
span:before {
    font-size: 40px;
}

```

```

.owl-carousel .owl-nav .owl-prev {
left: 30px !important;
}

.owl-carousel .owl-nav .owl-next {
right: 30px !important;
}

#accordion .card h5 a {
display: block;
text-align: left;
text-decoration: none;
color: #007bff;
position: relative;
-webkit-box-shadow: 0 3px 10px -2px
rgba(0, 0, 0, 0.2) !important;
box-shadow: 0 3px 10px -2px rgba(0, 0, 0,
0.2) !important;
border-radius: 0;
}

#accordion .card h5 a .icon {
position: absolute;
right: 20px;
top: 50%;
-webkit-transform: translateY(-50%)
rotate(-180deg);
-ms-transform: translateY(-50%) rotate(-
180deg);
transform: translateY(-50%) rotate(-
180deg);
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
}

#accordion .card h5 a:hover {
text-decoration: none;
-webkit-box-shadow: 0 3px 10px -2px
rgba(0, 0, 0, 0.2) !important;
box-shadow: 0 3px 10px -2px rgba(0, 0, 0,
0.2) !important;
}

#accordion .card h5 a.collapsed {
color: #000;
-webkit-box-shadow: none !important;
box-shadow: none !important;
}

#accordion .card h5 a.collapsed .icon {
right: 20px;
top: 50%;
-webkit-transform: translateY(-50%);
-ms-transform: translateY(-50%);
transform: translateY(-50%);
}

#accordion .card h5 a.collapsed:hover {
text-decoration: none;
-webkit-box-shadow: 0 3px 10px -2px
rgba(0, 0, 0, 0.2) !important;
box-shadow: 0 3px 10px -2px rgba(0, 0, 0,
0.2) !important;
}

#accordion .card .card-body {
padding-top: 15px;
}

.testimonial {
font-size: 30px;
color: #000;
}

.media-testimonial img {
width: 100px;
border-radius: 50%;
}

.media-testimonial blockquote p {
font-size: 20px;
color: #000;
font-style: italic;
}

.comment-form-wrap {
clear: both;
}

.comment-list {
padding: 0;
margin: 0;
}

.comment-list .children {
padding: 50px 0 0 40px;
margin: 0;
float: left;
width: 100%;
}

.comment-list li {
padding: 0;
margin: 0 0 30px 0;
float: left;
width: 100%;
clear: both;
list-style: none;
}

.comment-list li .vcard {
width: 80px;
float: left;
}

.comment-list li .vcard img {
width: 50px;
border-radius: 50%;
}

```

```

    -webkit-box-shadow: 0 2px 20px -3px
    rgba(0, 0, 0, 0.1);
    box-shadow: 0 2px 20px -3px rgba(0, 0, 0,
    0.1);
    -webkit-transform: scale(1.05);
    -ms-transform: scale(1.05);
    transform: scale(1.05);
    z-index: 2;
    border-bottom: 10px solid #007bff;
}

.media-custom {
background: #fff;
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
margin-bottom: 30px;
position: relative;
top: 0;
}

.media-custom .meta-post {
color: #ced4da;
font-size: 13px;
text-transform: uppercase;
}

.media-custom > a {
position: relative;
overflow: hidden;
display: block;
}

.media-custom .meta-chat {
color: #ced4da;
}

.media-custom .meta-chat:hover {
color: #6c757d;
}

.media-custom img {
-webkit-transition: .3s all ease;
-o-transition: .3s all ease;
transition: .3s all ease;
}

.media-custom:focus, .media-custom:hover {
top: -2px;
-webkit-box-shadow: 0 2px 40px 0 rgba(0,
0, 0, 0.2);
box-shadow: 0 2px 40px 0 rgba(0, 0, 0, 0.2);
}

.media-custom:focus img, .media-
custom:hover img {
margin: 0;
float: left;
width: 100%;
}

.search-form-wrap {

```

```

margin-bottom: 5em;
display: block;
}

.search-form .form-group {
position: relative;
}

.search-form .form-group #s {
padding-right: 50px;
background: #f7f7f7;
padding: 15px 15px;
border: none;
}

.search-form .icon {
position: absolute;
top: 50%;
right: 20px;
-webkit-transform: translateY(-50%);
-ms-transform: translateY(-50%);
transform: translateY(-50%);
}

blockquote {
padding-left: 30px;
border-left: 10px solid #e6e6e6;
}

blockquote p {
font-size: 18px;
font-style: italic;
color: #000;
}

.list-unstyled.check li {
position: relative;
padding-left: 30px;
line-height: 1.3;
margin-bottom: 10px;
}

.list-unstyled.check li:before {
color: #17a2b8;
left: 0;
font-family: "Ionicons";
content: "\f122";
position: absolute;
}

#modalAppointment .modal-content {
border-radius: 0;
border: none;
}

#modalAppointment .modal-body,
#modalAppointment .modal-footer {
padding: 40px;
}

.overflow {
position: relative;
}

```

```

overflow-x: hidden;
}

.site-footer {
background-size: cover;
background-position: center center;
background-repeat: no-repeat;
padding: 5em 0;
background: #000;
}

.site-footer .post-entry-sidebar ul li a h4 {
color: #fff;
}

.site-footer .footer-social li a > span {
width: 30px;
}

.site-footer a {
color: #fff;
}

.site-footer a:hover {
opacity: .5;
}

.site-footer h3 {
color: #fff;
margin-bottom: 30px;
font-size: 14px;
text-transform: uppercase;
letter-spacing: .2em;
}

.site-footer p {
color: rgba(255, 255, 255, 0.5);
}

.blog-entry-footer .post-meta {
color: white;
font-size: 15px;
}

.border-t {
border-top: 1px solid #f8f9fa;
}

.copyright {
font-size: 14px;
}

.element-animate {
/*opacity:0;*/
visibility: inherit;
}

#loader {
position: fixed;
width: 96px;
height: 96px;
left: 50%;
top: 50%;
-webkit-transform: translate(-50%, -50%);
-ms-transform: translate(-50%, -50%);
transform: translate(-50%, -50%);
background-color: rgba(255, 255, 255, 0.9);
-webkit-box-shadow: 0px 24px 64px
rgba(0, 0, 0, 0.24);
box-shadow: 0px 24px 64px rgba(0, 0, 0,
0.24);
border-radius: 16px;
opacity: 0;
visibility: hidden;
-webkit-transition: opacity .2s ease-out,
visibility 0s linear .2s;
-o-transition: opacity .2s ease-out, visibility
0s linear .2s;
transition: opacity .2s ease-out, visibility 0s
linear .2s;
z-index: 1000;
}

#loader.fullscreen {
padding: 0;
left: 0;
top: 0;
width: 100%;
height: 100%;
-webkit-transform: none;
-ms-transform: none;
transform: none;
background-color: #fff;
border-radius: 0;
-webkit-box-shadow: none;
box-shadow: none;
}

#loader.show {
-webkit-transition: opacity .4s ease-out,
visibility 0s linear 0s;
-o-transition: opacity .4s ease-out, visibility
0s linear 0s;
transition: opacity .4s ease-out, visibility 0s
linear 0s;
visibility: visible;
opacity: 1;
}

#loader.circular {
-webkit-animation: loader-rotate 2s linear
infinite;
animation: loader-rotate 2s linear infinite;
position: absolute;
left: calc(50% - 24px);
top: calc(50% - 24px);
display: block;
-webkit-transform: rotate(0deg);
-ms-transform: rotate(0deg);
transform: rotate(0deg);
}

#loader .path {
stroke-dasharray: 1, 200;
}

```

```

    stroke-dashoffset: 0;
    -webkit-animation: loader-dash 1.5s ease-
in-out infinite;
    animation: loader-dash 1.5s ease-in-out
infinite;
    stroke-linecap: round;
}

@-webkit-keyframes loader-rotate {
    100% {
        -webkit-transform: rotate(360deg);
    }
}

@keyframes loader-rotate {
    100% {
        -webkit-transform: rotate(360deg);
    }
}

@-webkit-keyframes loader-dash {
    0% {
        stroke-dasharray: 1, 200;
        stroke-dashoffset: 0;
    }
    50% {
        stroke-dasharray: 89, 200;
        stroke-dashoffset: -35px;
    }
    100% {
        stroke-dasharray: 89, 200;
        stroke-dashoffset: -136px;
    }
}

@keyframes loader-dash {
    0% {
        stroke-dasharray: 1, 200;
        stroke-dashoffset: 0;
    }
    50% {
        stroke-dasharray: 89, 200;
        stroke-dashoffset: -35px;
    }
    100% {
        stroke-dasharray: 89, 200;
        stroke-dashoffset: -136px;
    }
}

```

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 20 наименований.

Один экземпляр сдан на кафедру.

« ____ » _____ 2021 г.

_____ Ленцов Николай Олегович

подпись

Федеральное государственное автономное
образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

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

Кафедра прикладной информатики, математики и естественно-научных
дисциплин

УТВЕРЖДАЮ

Заведующий кафедрой


Е. Н. Скуратенко

подпись

« 18 » июня 2021 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03 Прикладная информатика


Разработка информационной системы онлайн экскурсий по объектам
культурного наследия Республики Хакасия

Руководитель


18.06.21
подпись, дата

зав.кафедрой, канд. техн. наук Е.Н. Скуратенко

Выпускник


18.06.21
подпись, дата

Н.О. Ленцов

Консультанты
по разделам:

Экономический


18.06.21
подпись, дата

Е. Н. Скуратенко

Нормоконтролер


18.06.21
подпись, дата

В. И. Кокова

Абакан 2021

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

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

Кафедра прикладной информатики, математики и естественно-научных
дисциплин

УТВЕРЖДАЮ

Заведующий кафедрой


Е.Н. Скуратенко

подпись

« 08 » апреля 2021 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Ленцову Николаю Олеговичу

Группа ХБ 17-03

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.

Утверждена приказом по институту № 222 от 08.04.2021 г.

Руководитель ВКР: Е.Н. Скуратенко, зав. кафедрой, канд. техн. наук, ХТИ – филиал СФУ

Исходные данные для ВКР: информация о деятельности Министерства культуры Республики Хакасия, информация о локальном сервере.

Перечень разделов ВКР:

1. Анализ предметной области.
2. Разработка веб-приложения для информационной системы онлайн экскурсий по объектам культурного наследия Республики Хакасия.
3. Оценка экономической эффективности информационной системы.

Перечень графического материала: нет

Руководитель ВКР


_____ подпись

Е. Н. Скуратенко

Задание принял к исполнению


_____ подпись

Н.О. Ленцов

«08» апреля 2021 г.