Частное учреждение образования

«Колледж бизнеса и права»

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

ПО ВЕБ-ПРОГРАММИРОВАНИЮ

СОЗДАНИЕ САЙТА «название»

ОП Т.ХХХХХХ

Руководитель практики ( М.О.Кудрявцева )

Учащийся ( А.О.Зеневич )

2023

Содержание

Изм.

Лист

№ докум.

Подпись

Дата

Лист

2

ОП Т.ХХХХХХ

Разраб.

Провер.

Т. контр.

Н. контр.

Утверд.

Отчет по учебной практике по разработке и сопровождению программного обеспечения

Лит.

Листов

КБП

[ВВЕДЕНИЕ 3](#_Toc106699083)

[1. Сущность, задачи создания сайта 4](#_Toc106699084)

[1.1 Цели создания сайта 4](#_Toc106699085)

[1.2 Проектирование содержимого сайта 4](#_Toc106699086)

[2 Проектирование сайта 5](#_Toc106699087)

[2.1 Структура сайта 5](#_Toc106699088)

[2.2 Дизайн сайта 5](#_Toc106699089)

[2.3 Создание макета сайта 5](#_Toc106699090)

[3 Верстка сайта 6](#_Toc106699091)

[ЗАКЛЮЧЕНИЕ 7](#_Toc106699092)

[Список информационных источников 8](#_Toc106699093)

[Приложение А 9](#_Toc106699094)

# ВВЕДЕНИЕ

В настоящее время развитие информационных технологий оказывает существенное влияние на различные сферы жизни человека, в том числе и на животноводство. Современные технологии позволяют создавать программные средства, способные оптимизировать процессы управления и контроля за животным миром, а также предоставлять эффективные инструменты для отслеживания перемещения и жизненных циклов животных.

HTML — стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

CSS — формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Также может применяться к любым XML-документам, например, к SVG или XUL.

JavaScript — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений.

RESTful API Service — это тип веб-сервиса, который использует архитектурный стиль REST (Representational State Transfer), чтобы обмениваться данными между клиентом и сервером. RESTful API Service предоставляет универсальный набор правил для создания и обмена информацией через сеть, используя HTTP протокол.

Основные принципы RESTful API Service включают использование URI (Uniform Resource Identifier) для идентификации ресурсов, использование HTTP методов (GET, POST, PUT, DELETE) для выполнения операций с ресурсами, а также использование форматов данных, таких как JSON или XML, для обмена информацией.

RESTful API Service обеспечивает более простой и легковесный подход к созданию веб-сервисов, что позволяет разработчикам быстрее создавать и внедрять приложения. Этот подход также облегчает интеграцию между различными приложениями и платформами, так как RESTful API Service является независимым от языка и платформы решением для обмена данными.

В рамках данного курсового проекта ставится задача разработки RESTful API сервиса, который будет предоставлять возможность просмотра информации о животных, их типах, местоположении и перемещениях. Реализуемый сервис позволит осуществлять поиск, создание, изменение и удаление данных о животных, их типах, точках местоположения и местах посещения животных.

Актуальность поставленной задачи обусловлена необходимостью создания эффективных инструментов для управления и контроля за животным миром, а также необходимостью автоматизации процессов сбора и обработки информации, связанной с животными. Кроме того, разработка подобных сервисов способствует развитию индустрии информационных технологий и расширению возможностей их применения в различных сферах деятельности.

К курсовому проекту также прилагается пояснительная записка, состоящая из четырех разделов и графической части.

# 1. Сущность, задачи создания сайта

Целями практики по разработке и сопровождению программного обеспечения являются:

* + закрепление знаний, связанных с технологией создания веб-приложений;
  + приобретение навыков разработки веб-приложений с использованием современных средств обработки информации;
  + выработка умений применять средства стандартных библиотек при создании сайтов.

Задачами практики по разработке и сопровождению программного обеспечения являются:

* + углубленное изучение принципов организации программного обеспечения и технологии его проектирования;
  + владение методами надежного программирования;
  + умение разрабатывать веб-приложения в соответствии с промышленными требованиями, обеспечивая высокий уровень качества программного обеспечения и экономической эффективности;
  + оформление комплекта документации на созданное программное обеспечение.

## 1.1 Цели создания сайта

1. Цели создания сайта:

* исследование предметной области: осуществить изучение и анализ требований и особенностей отслеживания перемещения животных, включая сбор и обработку данных.
* разработка RESTful API сервиса: создать веб-приложение, которое предоставляет программный интерфейс для отслеживания перемещения животных.
* улучшение эффективности и эффективности деятельности заказчика: обеспечить заказчику возможность точного отслеживания и управления перемещением животных в их предприятиях или на открытой местности.

1. Задачи, которые планируется решать посредством сайта:

* предоставление доступа к информации о перемещении животных: пользователи смогут получать актуальные данные о текущем местоположении и состоянии животных.
* определение географических координат: веб-приложение будет определять координаты животных с использованием GPS или других технологий геопозиционирования.
* управление перемещением животных: пользователи смогут задавать определенные параметры для перемещения животных и получать уведомления о них.

1. Целевая аудитория сайта:

* владельцы ферм и животноводческих предприятий: основная группа пользователей, которые заинтересованы в отслеживании и контроле перемещения своих животных.
* персонал и управляющие предприятий: пользователи, которые ответственны за контроль и координацию перемещения животных на предприятии.
* ветеринарные службы и спасательные организации: пользователи, которые могут использовать информацию о перемещении животных для предоставления необходимой помощи и поддержки.

1. Задачи, решаемые при помощи сайта:

* мониторинг здоровья животных: позволяет владельцам и ветеринарным службам отслеживать состояние животных, например, температуру, пульс, активность и другие показатели, которые могут указывать на заболевания или проблемы.
* планирование маршрутов перемещения: пользователи могут создавать оптимальные маршруты для перемещения животных, учитывая факторы, такие как расстояние, доступность пастбищ, условия питания и другие факторы.
* оповещения и уведомления: владельцы животных и персонал предприятий могут получать уведомления о важных событиях, например, о выходе животного за пределы определенной зоны или о появлении аварийных ситуаций.
* аналитика и отчетность: предоставление пользователю статистической информации и отчетов о перемещении животных, их активности, продуктивности и других параметрах для анализа и принятия управленческих решений.
* интеграция с другими системами: возможность интеграции с другими информационными системами предприятия, такими как системы учета и управления скотом, для обмена данными и совместной работы.
* аутентификация и авторизация: обеспечение безопасного доступа к информации о перемещении животных только уполномоченным пользователям, защита данных от несанкционированного доступа.

## 1.2 Проектирование содержимого сайта

подготовка текстовых материалов для сайта (приводится перечень документов с информацией, которую необходимо разместить на сайте);

подготовка графических материалов для сайта (приводится перечень графических материалов, которые необходимо разместить на сайте).

необходимые для разработки сайта документы и материалы должны быть приведены в приложении к проекту.

Приводится первоначальная информационная структура сайта в виде иерархического списка или ментальной карты и карта пути пользователя для решения основных задач на сайте.

# 2 Проектирование сайта

## 2.1 Структура сайта

внутренняя структура сайта (предоставляется список страниц сайта в виде структуры (список, организационная диаграмма);

карта сайта (приводится структурная схема сайта (карта сайта) и динамических блоков, если они имеются). Карту сайта разместить в графической части.

список модулей сайта (приводится физическая структура сайта в виде организационной диаграммы и таблицы, приводится список всех используемых модулей). Разрабатывается диаграмма развертывания сайта.

## 2.2 Дизайн сайта

обоснование типа макета сайта (фиксированный, резиновый, адаптивный, комбинированный);

внешняя структура сайта (обосновывается выбор модульной схемы сайта, модульной сетки)

главная страница сайта (перечисляются разделы главной страницы сайта и приводится модульная схема главной страницы в графическом виде с указанием всех элементов)

внутренние страницы сайта (описывается содержание всех внутренних страниц сайта и приводятся модульные схемы страниц в графическом виде)

шапка сайта (описывается, что должно быть размещено в шапке сайта)

подвал сайта (описывается, что должно быть размещено в нижней части сайта)

цветовое решение сайта (определяется цветовая схема, перечисляются основные цвета, используемые на сайте, приводятся коды цветов для элементов дизайна – фон, шрифт, ссылки, элементы навигации и т.д.)

## 2.3 Создание макета сайта

выбор программного обеспечения для создания макета сайта;

определение размера макета сайта;

определение фона для макета сайта;

разметка макета сайта (указывается размер разделов макета в пикселях)

* + создание шапки сайта (описывается процесс создания, используемые графические элементы);
  + создание навигационного меню (описывается процесс создания, используемые графические элементы);
  + создание подвала сайта (описывается процесс создания, используемые графические элементы);
  + создание контента сайта (описывается процесс создания, используемые графические элементы);
  + графическое представление макетов страниц сайта размещается в приложении к ПЗ.

# 3 Верстка сайта

Для разработки макета сайта в соответствии с выбранными технологиями (Visual Studio, Visual Studio Code, HTML, CSS, JavaScript, React, Bootstrap, C#, ASP.NET Web API, Entity Framework, ADO.NET и PostgreSQL), рекомендуется использовать следующие инструменты:

1. Интегрированная среда разработки (IDE) Visual Studio предоставляет мощные инструменты для разработки веб-приложений на базе .NET. Она поддерживает язык программирования C# и обеспечивает возможности разработки серверной части приложения на основе ASP.NET Web API.
2. Visual Studio Code (VS Code) — лёгкий и гибкий редактор кода, который поддерживает различные языки программирования, включая HTML, CSS и JavaScript. Он предлагает широкий выбор расширений и инструментов для удобной разработки веб-приложений.
3. HTML, CSS, JavaScript: Эти языки программирования широко используются для разработки пользовательского интерфейса веб-приложений. HTML используется для определения структуры и содержимого веб-страницы, CSS — для оформления и стилизации элементов, а JavaScript — для создания интерактивности и динамического поведения сайта.
4. React — это библиотека JavaScript, которая упрощает создание пользовательского интерфейса. Она позволяет разбить интерфейс на независимые компоненты и обеспечивает эффективное обновление интерфейса при изменении данных.
5. Bootstrap — это популярный фреймворк CSS, который предоставляет готовые стили и компоненты для создания современного и отзывчивого дизайна веб-приложений. Он упрощает разработку и обеспечивает согласованный внешний вид для различных устройств и разрешений экрана.
6. C#, ASP.NET Web API, Entity Framework, ADO.NET: Эти технологии являются основой для разработки серверной части вашего web-приложения. C# — это язык программирования, используемый в экосистеме .NET. ASP.NET Web API позволяет создавать RESTful API для взаимодействия с клиентской частью. Entity Framework и ADO.NET предоставляют инструменты для работы с базой данных PostgreSQL.
7. PostgreSQL — это открытая реляционная система управления базами данных (СУБД). Она обладает мощными возможностями, высокой производительностью и широким набором функций. PostgreSQL поддерживает стандарт SQL и предоставляет надежное хранение и обработку данных. Вы можете использовать PostgreSQL для создания и управления базой данных, которая будет хранить информацию о перемещении животных.

В результате использования вышеуказанного программного обеспечения, вы сможете создать макет сайта для вашего web-приложения "RESTful API сервиса для отслеживания перемещения животных". Это позволит вам разработать интерфейс пользователя с использованием HTML, CSS, JavaScript, React и Bootstrap, а также реализовать серверную часть приложения с помощью C#, ASP.NET Web API, Entity Framework, ADO.NET и PostgreSQL. Определение структуры проекта и разработка макета сайта позволят вам создать пользовательский интерфейс, который будет удобным и эффективным для отслеживания перемещения животных.

Важно также учесть аспекты безопасности, масштабируемости и производительности при выборе и использовании указанного программного обеспечения.

Описание используемых функций и процедур

описание верстки (таблицы, блоки, фреймы, стили) структура и формат страниц, цветовая разметка страниц, фоновое изображение, подключение звука;

описание последовательности верстки макета сайта;

взаимосвязи функций;

код всех модулей с комментариями и указанием элементов управления (событиями), которые инициируют их выполнение, должны быть представлены в приложении.

описание применяемых технологий;

оптимизация графики для размещения на сайте;

кроссплатформенность, кроссбраузерность.

# ЗАКЛЮЧЕНИЕ

В «Заключение» необходимо отразить:

* как выполнена поставленная задача;
* степень соответствия проектных решений поставленной задаче;
* причины несоответствия (если таковы имеются).

Удобнее всего компоновать заключение из готовых выводов, сформулированных в конце соответствующих разделов отчета.

напрмер

За время практики по созданию и сопровождению программного обеспечения были изучены процессы создания, внедрения и сопровождения программных приложений в различных средах программирования.

Во время практики удалось накопить неоценимый опыт в разработке приложений в интегрированной среде визуального программирования Misrosoft Visual Studio 2019 с помощью технологии WPF и использованными языками С# и XAML, развить умения самостоятельно находить оптимальный метод решения задачи. Кроме того, изучить принципы организации и технологии реализации программного обеспечения, развить умения разработки программ в соответствии с требованиями технического задания, обеспечивая высокий уровень качества программного обеспечения и экономической эффективности. Получить навыки оформления комплекта документации на созданное программное обеспечение.

При создании программных приложений, входящих в обязательную программу практики, были отработаны навыки владения методами надежного программирования и изучены способы определения экономической себестоимости и эффективности разработки программного обеспечения.

Рассказать про свой проект

# Список информационных источников

1. Багласова, Т.Г. Методические указания по оформлению курсовых и дипломных проектов / Т.Г. Багласова, К.О. Якимович. – Минск : КБП, 2013. – 29 c.
2. Вендров, А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. / А.М. Вендров. – 2-е изд., перераб. и доп.– М. : Финансы и статистика, 2006. — 544 с.
3. Макфарланд, Д. Большая книга CSS3 / Д. Макфарланд. – 3-е изд. – СПб. : Питер, 2014. — 608 с.
4. Мацяшек. Анализ и проектирование информационных систем с помощью UML 2.0 / Мацяшек, А. М. Лешек: – М. – ООО “И.Д. Вильямс”, 2008. – 816 с.
5. Орлов С. А. Технологии разработки программного обеспечения: Учебник для вузов. 4-е изд. / С. А. Орлов, Б. Я. Цилькер – СПб. : Питер, 2012. — 608 с.
6. Просницкий А. Самоучитель «Управление проектами в Microsoft Project 2010» / А. Просницкий. – Київ, Казань, Красноярск, Рига, Санкт-Петербург, Львів, Чернівці, Івано-Франківськ, 2011.
7. Рудаков А.В., Технология разработки программных продуктов. Практикум: учеб. пособие для студ. учреждений сред. проф. образования / А.В. Рудаков, Г.Н. Федорова. 4-е изд., стер. – М. : Издательский центр «Академия»; 2014. — 192 с.
8. Тепляков С. Паттерны проектирования на платформе .NET. / С. Тепляков – СПб. : Питер, 2018. – 317 с.
9. Экономика предприятия. Практикум / Э. В. Крум [и др.] ; под ред. Э. В. Крум. – Минск : Издательство Гревцова, 2009. – 355 с.
10. Язык программирования C#. Классика Computers Science / А. Хейлсберг [и др]. – 4-е изд. – СПб. : Питер, 2012. — 784 с.
11. Общие требования к тестовым документам : ГОСТ 2.105-95. – Введ. 01.01.1996. – Минск : Межгос. совет по стандартизации, метрологии и сертификации, 1995. – 84 с.
12. Описание программы. Требования к содержанию, оформлению и контролю качества : ГОСТ 19.402-2000. – Введ. 01.09.2001. – Минск : Межгос. совет по стандартизации, метрологии и сертификации, 2000. – 14 с.
13. Текст программы. Требования к содержанию, оформлению и контролю качества : ГОСТ 19.401-2000. – Введ. 01.09.2001. – Минск : Межгос. совет по стандартизации, метрологии и сертификации, 2000. – 16 с.
14. Полное руководство по языку программирования С# 9.0 и платформе .NET 5 [Электронный ресурс] / METANIT.COM. Сайт о программировании. – metanit.com, 2012-2021. – Режим доступа : http://metanit.com/sharp/tutorial. – Дата доступа : 15.04.2021.

# Приложение А

(обязательное)

Код модулей сайта

*(отдельные куски кода с комментариями)*