

**Слет научных обществ учащихся образовательных организаций
общего и дополнительного образования детей г. Нижневартовска
в 2021-2022 учебном году**

**Секция 3. Информатика, вычислительная техника, телекоммуникации.
Информационная безопасность.**

Проектно-исследовательская работа

«Интерактивный квест «И тем полней люблю, чем глубже знаю»

Автор работы: Кондрюков Дмитрий Сергеевич,
ученик 11Б класса
муниципального бюджетного
общеобразовательного учреждения
«Средняя школа № 12»

Научный руководитель: Коренева Наталья Викторовна,
учитель информатики
муниципального бюджетного
общеобразовательного учреждения
«Средняя школа № 12»

Нижневартовск

2022 г.

Проектно-исследовательская работа
«Интерактивный веб-квест «И тем полней люблю, чем глубже знаю»
Кондрюкова Дмитрия, муниципальное бюджетное общеобразовательное
учреждение «Средняя школа №12», ученика 11Б класса

Аннотация к работе

В современном мире информационных технологий сеть Интернет переживает крутой подъем и огромную востребованность. Множество серверов с хранящейся на них информацией способствует росту популярности и доступности различных ресурсов. На сегодняшний день интернет – это возможности свободного получения и распространения различной информации.

Сайтостроение – процесс создания сайтов. Он начинается с разработки дизайна и заканчивается написанием программного кода, а также публикацией готового продукта в глобальной сети. Создание сайта позволяет донести информацию максимально быстро до огромного количества людей.

Ученик считает, что в Нижневартовске большое количество достопримечательностей и культурных заведений, которые будет полезно и интересно посетить как взрослым путешественникам, так и детям. В этом году городу Нижневартовску исполняется 50 лет – юбилей. В честь такого события обучающийся, проявляя заинтересованность к сайтостроению, самостоятельно изучив языки программирования создал интерактивный веб-квест. Веб-квест направлен на систематизацию знаний об исторически значимых объектах города. Для жителей и гостей города данный ресурс станет путеводителем по объектам Нижневартовска для досуга и культурного развития.

Цель данного проекта является создание интерактивного веб-квеста о достопримечательностях Нижневартовска с применением языков программирования для расширения информационно-образовательного пространства и повышения уровня краеведческих знаний школьников, горожан и гостей города Нижневартовска

Задачи:

- 1) познакомиться с понятием «сайтостроение»;
- 2) определить отличительные признаки и преимущества способов построения сайта;
- 3) познакомиться с этапами веб-разработки;
- 4) изучить возможности языков программирования, проанализировать их и выбрать самые удобные в создании сайтов;
- 5) реализовать интерактивную карту на платформе «Яндекс.Карты»;

- 6) разработать внешний вид сайта;
- 7) разместить (готовый сайт) интерактивный веб-квест на хостинге;
- 8) презентовать свою исследовательскую работу учителям школы и одноклассникам;

Методы исследования:

Работа с источниками информации (Интернет-ресурсы, литература).

экспериментально-теоретического уровня: изучение и анализ языков программирования; создание веб-квеста на основе полученных знаний;

теоретического уровня: анализ и синтез полученных результатов.

Проектно-исследовательская работа
«Интерактивный веб-квест «И тем полней люблю, чем глубже знаю»
Кондюкова Дмитрия, муниципальное бюджетное общеобразовательное
учреждение «Средняя школа №12», ученика 11Б класса

План исследований

В наш век стремительного развития технологий и Интернета, очень популярным становится умение создавать сайты. В 12 лет меня заинтересовало сайтостроение. Я начал подробно изучать алгоритмы разработки сайтов.

Актуальность данной работы заключается в том, что на сегодняшний день интернет – это возможности свободного получения и распространения различной информации. Сайты служат для получения нужной информации для широкого круга пользователей.

В этом году нашему городу исполняется 50 лет – юбилей. В Нижневартовске огромное количество достопримечательностей и культурных заведений, которые будет полезно и интересно посетить как взрослым путешественникам, так и детям. Но в городе недостаточное количество информационных источников, доступных для школьников, горожан и гостей города Нижневартовска, об истории и знаменитых людях города.

Решить поставленную мною проблему можно с помощью создания интерактивного веб-квест «И тем полней люблю, чем глубже знаю». Для жителей Нижневартовска веб-квест направлен на систематизацию знаний об исторически значимых объектах города. Для гостей города данный ресурс также станет путеводителем по объектам Нижневартовска для досуга и культурного развития.

Гипотеза: изучив возможности языков программирования для создания сайтов, можно создать интерактивный веб-квест «Путешествие по Нижневартовску», разместив его в сети Интернет. В этом случае жители и гости города легко смогут понять расположение всех достопримечательностей, узнать интересные факты о городе.

Объект исследования – интерактивный веб-квест

Предмет исследования – языки программирования сайтов

Методы исследования:

Работа с источниками информации (Интернет-ресурсы, литература).

экспериментально-теоретического уровня: изучение и анализ языков программирования; создание веб-квеста на основе полученных знаний;

теоретического уровня: анализ и синтез полученных результатов.

В работе были использованы следующие источники:

1. snipp.ru – Для веб-разработчиков [электронный ресурс]: URL: <https://www.snipp.ru/>.

2. php.net – Документация по PHP [электронный ресурс]: URL: <https://www.php.net/>.
3. JavaScript и JQuery: исчерпывающее руководство / Дэвид Макфарланд – 3 издание – Москва: Эксмо, 2017. – 880 с.
4. Новая большая книга CSS / Дэвид Макфарланд – Санкт-Петербург: Питер СПб, 2019. – 720 с

Содержание:

Введение	7-8
1. Основная (теоретическая) часть.....	9
1.1. Что такое сайтостроение	9
1.2. Отличительные признаки способов сайтостроения	9
1.3. Виды языков программирования. База данных	9
1.4. . Хостинг. Яндекс Карты. Yandex API	10
2. Основная (практическая) часть. Создание интерактивного веб-квеста	10
2.1. Разработка интерактивной карты.....	10
2.2. Frontend разработка сайта (визуальная часть).....	13
2.3. Backend разработка сайта (серверная часть).....	14
Заключение	16
Библиография	16

Проектно-исследовательская работа

«Интерактивный веб-квест «И тем полней люблю, чем глубже знаю»

Кондюкова Дмитрия, муниципальное бюджетное общеобразовательное учреждение «Средняя школа №12», ученика 11Б класса

Введение

В наш век стремительного развития технологий и Интернета, очень популярным становится умение создавать сайты. В 12 лет меня заинтересовало сайтостроение. Я начал подробно изучать алгоритмы разработки сайтов.

Актуальность данной работы заключается в том, что на сегодняшний день интернет – это возможности свободного получения и распространения различной информации. Сайты служат для получения нужной информации для широкого круга пользователей. В этом году нашему городу исполняется 50 лет – юбилей. В Нижневартовске огромное количество достопримечательностей и культурных заведений, которые будет полезно и интересно посетить как взрослым путешественникам, так и детям. В честь такого события я решил создать интерактивный квест. Для жителей Нижневартовска веб-квест направлен на систематизацию знаний об исторически значимых объектах города. Для гостей города данный ресурс также станет путеводителем по объектам Нижневартовска для досуга и культурного развития.

Проблема: недостаточное количество информационных источников, доступных для школьников, горожан и гостей города Нижневартовска, об истории и знаменитых людях города.

Гипотеза: изучив возможности языков программирования для создания сайтов, можно создать интерактивный веб-квест «Путешествие по Нижневартовску», разместив его в сети Интернет. В этом случае жители и гости города легко смогут понять расположение всех достопримечательностей, узнать интересные факты о городе.

Объект исследования – интерактивный веб-квест

Предмет исследования – языки программирования сайтов

Цель моей работы - создание интерактивного веб-квеста о достопримечательностях Нижневартовска с применением языков программирования для расширения информационно-образовательного пространства и повышения уровня краеведческих знаний школьников, горожан и гостей города Нижневартовска

Задачи:

- 1) познакомиться с понятием «сайтостроение»;
- 2) определить отличительные признаки и преимущества способов построения сайта;

- 3) познакомиться с этапами веб-разработки;
- 4) изучить возможности языков программирования, проанализировать их и выбрать самые удобные в создании сайтов;
- 5) реализовать интерактивную карту на платформе «Яндекс.Карты»;
- 6) разработать внешний вид сайта;
- 7) разместить (готовый сайт) интерактивный веб-квест на хостинге;
- 8) презентовать свою исследовательскую работу учителям школы и одноклассникам;

Методы исследования:

Работа с источниками информации (Интернет-ресурсы, литература).

экспериментально-теоретического уровня: изучение и анализ языков программирования; создание веб-квеста на основе полученных знаний;

теоретического уровня: анализ и синтез полученных результатов.

1. Основная (теоретическая) часть

1.1. Сайтостроение

Сайтостроение – процесс создания сайтов. Он начинается с разработки дизайна и заканчивается написанием программного кода, а также публикацией готового продукта в глобальной сети.

1.2. Отличительные признаки видов создания сайтов

Есть 3 способа сайтостроения:

- 1) Использование конструктора (*wix, tilda, ucoz*) – когда разработчик собирает всю конструкцию с помощью готовых «кубиков» конструктора.
- 2) Построение на CMS (*bitrix, wordpress, joomla*) – система управления веб-контентом. Базовый каркас и дополнительные инструменты позволяют поддерживать его работу, обновлять контент и взаимодействовать с пользователями.
- 3) Собственная разработка - творческий, но самый трудоемкий процесс. Написание сайта требует углубленных знаний по языкам программирования и многому другому. Готовый сайт будет решать его задачи и не заставит тратить время на лишние процессы.

Мною создан сайт способом собственной разработки.

1.3. Виды языков программирования. База данных

В IT-сфере большой выбор среди языков программирования, применяемых в создании сайтов.

В своей работе использовал такие языки, как *HTML, CSS* – при оформлении внешнего вида сайта (frontend), а также *PHP, JavaScript* и библиотека *jQuery* – при разработке серверной части (backend). База данных (БД) *MySQL* помогают в удобном хранении массивной информации. PHP язык имеет преимущество – в него встроено множество функций для совместной интеграции с БД.

PHP (Hypertext Preprocessor) – скриптовый язык программирования для создания сайтов, web-приложений. Он простой в изучении синтаксиса.

JavaScript – язык программирования для frontend и backend разработки. Он также удобен в создании мобильных приложений.

Библиотека *jQuery* – популярная JavaScript-библиотека, предназначенная для минимализации сценариев HTML. Она имеет возможность получать доступ к элементам DOM, обращаться к атрибутам элементов и содержимому DOM, контролировать их.

HTML – язык гипертекстовой разметки, который применяется при создании сайтов в сети Интернет.

CSS – язык стилей. Предназначен для описания внешнего вида документа, написанного с использованием HTML.

1.4. Хостинг. Яндекс Карты. Yandex API

В качестве хостинга был выбран Veget. Данный ресурс предоставляет бесплатное размещение сайта в сети Интернет и систему управления с базой данных. Однако данный хостинг ограничивает клиента в выборе домена на бесплатной основе.

Интерактивная карта от «Яндекс» помогает пользователю продемонстрировать визуальное прохождение веб-квеста. Для работы с картой использовался способ описаний API.

2. Основная (практическая) часть. Создание интерактивного веб-квеста.

Для написания кода воспользовался программой Microsoft Visual Code.

Страница интерактивного веб-квеста разделена на две части. Одна занимает место под интерактивную карту, а другая под задания веб-квеста.

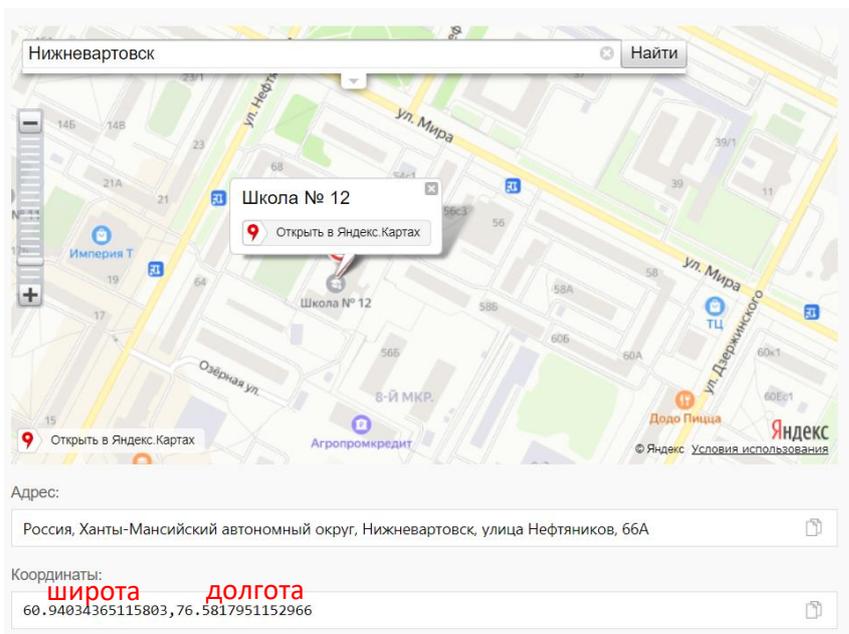
Для разработки интерактивной карты я пользовался документацией инструментов сервиса Yandex API (<https://yandex.ru/dev/maps/>).

2.1. Разработка интерактивной карты

Сначала нужно создать массив, содержащий в себе объекты, которые должны располагаться на карте. Из-за того, что этих меток будет достаточно много, не рационально прописывать отдельный код параметров для каждой из них. В упорядоченном хранении поможет база данных MySQL. В ней создаем таблицу *landmarks*, которая будет хранить основные параметры каждого объекта.

	id	latitude	longitude	hintContent	balloonContent
<input type="checkbox"/>   	1	60.969243323194206	76.531531977768	Памятник покорителям Самотлора	<div class="map_balloon"> <div class="map_img">...
<input type="checkbox"/>   	2	60.95043349182499	76.50123554664427	Аллея почёта авиационной техники	<div class="map_balloon"> <div class="map_img">...
<input type="checkbox"/>   	3	60.94363645178745	76.57393863513059	Памятник воинам-интернационалистам	<div class="map_balloon"> <div class="map_img">...
<input type="checkbox"/>   	4	60.939896089106135	76.56047394587577	Памятник Дворнику	<div class="map_balloon"> <div class="map_img">...

Чтобы карта понимала, где должна находиться метка, следует указать ее координаты (**latitude** – широта, **longitude** - долгота). Для их определения помогает сайт <https://snipp.ru/tools/address-coord/>.



Следующие колонки в таблице это:

hintContent – название метки при наведении курсором мыши,

balloonContent – содержание метки, открывающееся при нажатии на нее.

После того, как внесли все параметры меток, приступаем к разметке карты на сайте. Сначала подключаем JS – скрипты, которые инициализирует карту по заданным параметрам.

Добавление скрипта-шаблона карты через удаленное подключение (CDN):

```
<script src="https://api-maps.yandex.ru/2.1/?lang=ru_RU" type="text/javascript"></script>
```

Собственный скрипт(**map.js**), содержащий в себе параметры для нашей карты:

```
<script src="../map.js" type="text/javascript"></script>
```

Перед тем как написать параметры карты, получим все данные о метках, содержащихся в MySQL. Для этого соединяемся с базой данных на языке php (в целях безопасности содержимое файла db.php не демонстрируется).

```
include $_SERVER['DOCUMENT_ROOT'].'/assets/db.php';
```

После соединения, отправляем запрос на получение всех данных из таблицы

landmarks.

```
$q_getLandmarks = mysqli_query($db, "SELECT * FROM landmarks");
```

Полученные строки формируем в массив для удобства в дальнейшей работе.

```
while($landmarksData = mysqli_fetch_assoc($q_getLandmarks)){
    $placemarks[] = $landmarksData;
}
```

Так как работать с данным будем на языке JavaScript, преобразуем PHP массив в JSON формат, который понятен языку JavaScript. А также записываем полученный массив в переменную *placemarks*.

```
var placemarks = <?=json_encode($placemarks, JSON_UNESCAPED_UNICODE);?>, geoObjects = [];
```

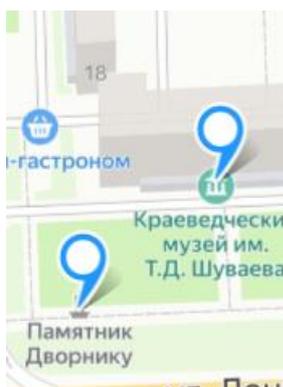
Дальше создаем главную функцию *init()* для инициализации Яндекс Карты. В ней мы укажем такие параметры как:

Координаты центрирования карты: `center: [60.93900010309049, 76.57740505353557],`

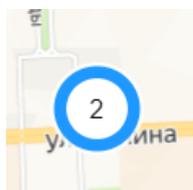
Значение увеличения карты: `zoom: 12`

Из-за большого количества меток и их близкого расположения друг к другу, при уменьшении масштаба карты, метки могут накладываться друг на друга. Чтобы избежать такой проблемы, вызываем метод кластеризации. С помощью него карта будет помнить о группировке меток.

Пример: две достопримечательности – Краеведческий музей и Памятник дворнику. Они находятся на близком расстоянии друг от друга.



При уменьшении масштаба расстояние между метками также уменьшается. Карта будет брать это во внимание и при критическом расстоянии объединит метки в одну.



Чтобы вернуть метки на исходные места, нажимаем на общую метку. Карта увеличивает масштаб до нужного размера и расставляет метки по их координатам.

Вернемся к коду

Теперь занимаемся размещением меток на карте. *Задача карты* – отобразить на себе информацию о достопримечательностях, но не всех сразу. Должен размещаться определенный объект, служащий правильным ответом к описанию

достопримечательности, если пользователь сделал правильный выбор из списка предложенных ответов.

Когда посетитель сделал выбор, он может проверить его правильность. Если ответ некорректный, то система даст возможность подумать еще раз. А при правильном исходе станет доступна кнопка перехода к следующему заданию. При нажатии на которую, система поймет, что пользователь верно решил задание и отобразит нужную метку на карте.

После этого даем команды на размещение меток на карте и добавляем их в кластер для возможности объединения при уменьшении масштаба.

```
clusterer.add(geoObjects);  
map.geoObjects.add(clusterer);
```

Последний шаг – вызвать функцию *init()* после полной загрузки страницы.

```
ymaps.ready(init);
```

2.2. Frontend разработка сайта (визуальная часть)

Вторым этапом является создание внешнего стиля сайта и формы для заданий квеста. Для создания формы, содержащей описание достопримечательности, использовался язык гипертекстовой разметки -HTML. Она включает в себя несколько контейнеров (div) с разными классами для размещения:

- заголовка** – task_header,
- содержания** – task_content,
- вариантов ответа** – task_answers,
- кнопок** – task_buttons,
- символики юбилея** – logo_birth

```
<div class="task"> flex  
  <div class="task_wrapper">  
    <div class="task_header">...</div>  
    <div class="task_content">...</div>  
    <div class="task_answers">...</div>  
    <div class="result_answer"> </div>  
    <div class="buttons">...</div> flex  
    <div class="logo_bitrh">...</div>  
  </div>  
</div>
```

Для красивого оформления на сайте применяется язык стилей – CSS. С помощью него можно задать свойства внешнего вида объектов HTML: цвет заднего фона, контура; название и размер шрифта; начертание текста; величину скругления углов; размер

картинки и т.д. CSS может воплотить любые задумки веб – дизайнера в реальности.

```
31  .task_header{
32      margin-bottom: 10px;
33      width: 100%;
34      padding: 6px;
35      font-size: 22px;
36      text-align: left;
37  }
38  .task_title{
39      font-size: 30px;
40      font-weight: 600;
41  }
42  .task_content{
43      font-size: 20px;
44      margin-bottom: 15px;
45  }
46
```

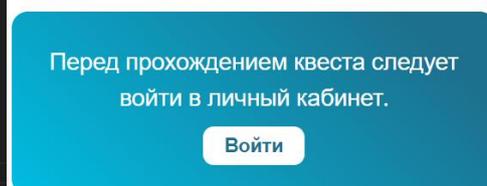
2.3. Backend разработка сайта (серверная часть)

При прохождении веб-квеста у посетителя должен быть личный кабинет. Для входа была создана форма авторизации. Данные об учетных записях хранятся в базе данных MySQL. Когда посетитель желает пройти квест система проверяет наличие сессии¹ учетной записи.

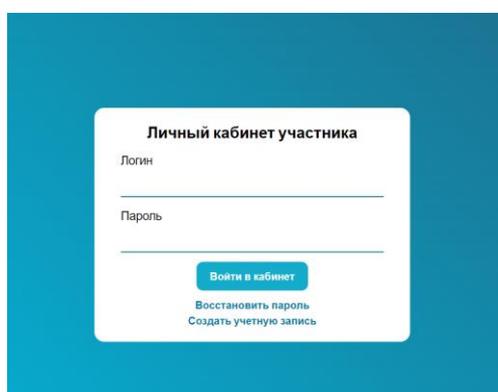
```
if(!isset($_SESSION['user'])){
    include $_SERVER['DOCUMENT_ROOT'].'/info.php';
}
```

Если она отсутствует, то браузер открывает страницу **info.php** с просьбой об авторизации на сайте.

```
<div class="information">
  <div class="info_box">
    <div class="info_content">
      <p>
        Перед прохождением квеста следует <br> войти в личный кабинет.
      </p>
      <div>
        <a href="/login/?back_url=/quest/" id="btn_success">Войти</a>
      </div>
    </div>
  </div>
</div>
```



При нажатии на кнопку «Войти» пользователь попадает на страницу входа.



¹ способ хранения нужной информации и работы с ней на языке PHP.

Когда ввели логин, пароль и была нажата кнопка «Войти в кабинет», данные отправляются методом **POST** на проверку и дальнейшую обработку.

Полученные данные проверяются на пустоту.

```
if($_POST['login'] == '' or $_POST['pass'] == ''){  
    $msg = 'Заполните все поля';
```

Если логин и пароль содержат какие-то символы, то они записываются в переменные. Причем пароль шифруется алгоритмом **md5** и в таком виде хранится в MySQL. Это помогает защитить аккаунты от легкого взлома паролей.

Например: пароль **111** при шифровании **md5** выглядит так:

698d51a19d8a121ce581499d7b701668.

```
$login = $_POST['login'];  
$pass = md5($_POST['pass']);
```

После этого в базу данных отправляется запрос на поиск строки с таким логином(email) и паролем (в зашифрованном виде).

```
$query_checkUser = mysqli_query($db, "SELECT * FROM users WHERE `email` = '$_login.'" AND `pass` = '$_pass.'");
```

Если такая строка найдена и все значения совпадают, то авторизация прошла успешно. Найденная запись формируется в массив.

```
$dataUser = mysqli_fetch_assoc($query_checkUser);
```

После чего система начинает создавать сессию пользователя. Она включает в себя ID пользователя, email и роль в системе.

```
$_SESSION['user']['id'] = $dataUser['id'];  
$_SESSION['user']['email'] = $dataUser['email'];  
$_SESSION['user']['role'] = $dataUser['role'];
```

При успешной генерации сессии сайт перебрасывает пользователя на ту страницу, откуда он попал на форму входа. В нашем случае это страница квеста.

```
if(isset($_GET['back_url'])){  
    header("Location: $_GET[back_url]");  
}else{  
    header("Location: /");
```

Система снова перепроверяет наличие сессии аккаунта и выдает доступ посетителю на выполнение заданий квест-игры.

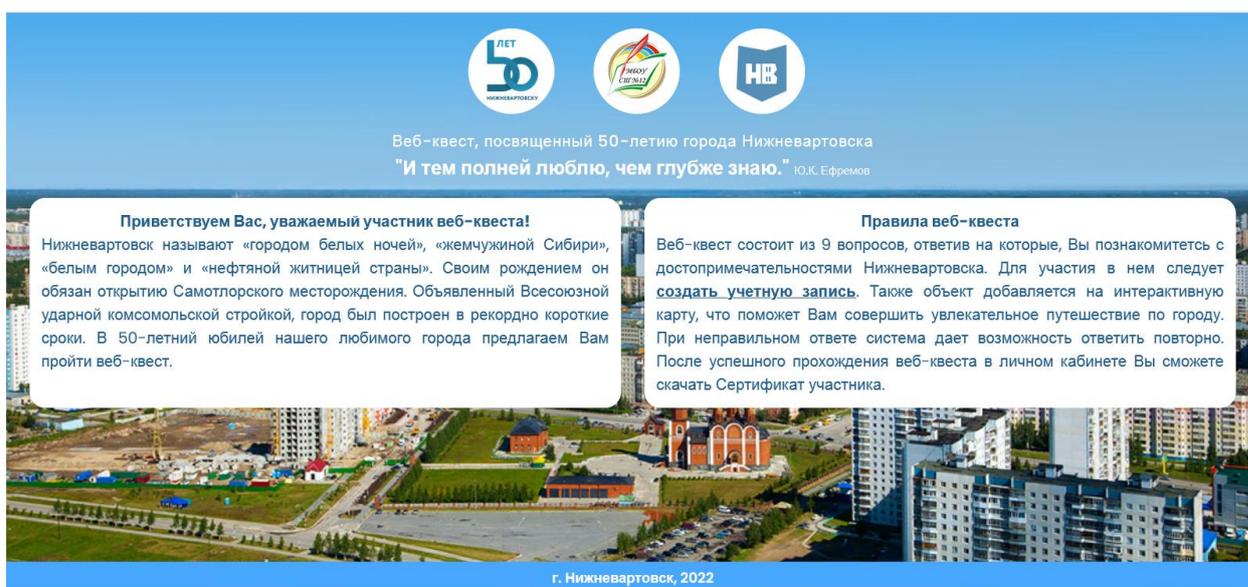
Заключение

Поставленная цель, задачи, - достигнуты, выдвинутая в начале гипотеза подтвердилась.

Изучив языки программирования PHP, JavaScript, библиотеку JQuery, а также язык гипертекстовой разметки HTML и язык стилей CSS мною создан интерактивный веб-квест «И тем полней люблю, чем глубже знаю» о достопримечательностях города Нижневартовска.

Веб-квест размещен в интернете в свободном доступе, его может пройти любой желающий в удобное для него время. Информация об интерактивном веб-квесте «И тем полней люблю, чем глубже знаю» размещена на сайте школы и портале системы образования города Нижневартовска. На сегодняшний день получили сертификаты об участии в веб-квесте 350 школьников общеобразовательных организаций города Нижневартовска.

Принять участие в веб-квесте можно по ссылке <http://c98288ut.beget.tech>



Практическая значимость заключается в том, что настоящий проект является самоучителем по созданию веб-квеста.

Библиография

1. snipp.ru – Для веб-разработчиков [электронный ресурс]: URL: <https://www.snipp.ru/>.
2. php.net – Документация по PHP [электронный ресурс]: URL: <https://www.php.net/>.
3. JavaScript и JQuery: исчерпывающее руководство / Дэвид Макфарланд – 3 издание – Москва: Эксмо, 2017. – 880 с.
4. Новая большая книга CSS / Дэвид Макфарланд – Санкт-Петербург: Питер СПб, 2019. – 720 с