**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ ДОПОЛНИТЕЛЬНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ ЦЕНТР ПРОФЕССИОНАЛЬНЫХ КВАЛИФИКАЦИЙ И СОДЕЙСТВИЯ ТРУДОУСТРОЙСТВУ «ПРОФЕССИОНАЛ»**

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к итоговой аттестационной работе на тему

**«Разработка web-ресурса с использованием технологий HTML, CSS, JavaScript. Bootstrap»**

**на примере web-ресурса:** http://srv191622.hoster-test.ru/dip/dip-test.html

слушателя **Андреевой Натальи Михайловны** группы №: **209**

программы профессиональной переподготовки

«Frontendразработка»

Москва, 2023

Оглавление

[Постановка задачи и план работы. 3](#_Toc103884852)

[ОСНОВНАЯ ЧАСТЬ. 4](#_Toc103884853)

[Назначение веб-ресурса. 4](#_Toc103884854)

[Описание этапов разработки, описание функционала с приложением листингов исходного программного кода основной функции, структурных модулей, пример кода-разметки **4**](#_Toc103884855)

[Список литературы 11](#_Toc103884856)

# ПОСТАНОВКА ЗАДАЧИ И ПЛАН РАБОТЫ.

Задачей является создание лендинга для рекламы услуг по тестированию группы выпускников курсов тестирования.При начале работы над проектом, необходимо определиться, как будет работать серверная часть, как будет работать проект со стороны клиента. После чего необходимо определиться какой будет внешний вид сайта, какой будет внешний вид личного кабинета пользователя.

Задачи к наполнению сайта: наличие информации о компании, «часто задаваемые вопросы», контакты, схема проезда к офису, каталог автомобилей, личный кабинет водителя, форма связи, описание условий аренды.

План работ:

1. Подключение фреймворка Bootstrap.

2. Создание базы данных на хостинге.

3. Верстка и программирование страницы сайта.

4. Подключение страниц сайта к базе данных.

# ОСНОВНАЯ ЧАСТЬ.

# Назначение веб-ресурса.

Лендинг (веб-ресурс) предназначен для предоставления информации потенциальным клиентам об услугах группы начинающих тестировщиков, видах и необходимости тестирования. Также для потенциальных клиентов разработана форма обратной связи, в котором они могут оставить свои контакты и отправить заявку со списком интересующих видов тестирования, или получить информацию о контактах и компании для самостоятельной связи и формировании заинтересованности в услугах.

## Описание этапов разработки, описание функционала с приложением листингов исходного программного кода основной функции, структурных модулей, пример кода-разметки

Этап разработки начинался с создания проекта с использованием фреймворка Bootstrapver 4.6 (далее “фреймворк”). Перед началом работы над проектом необходимо подключить фреймворк.

<script *src*="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"

*integrity*="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"

*crossorigin*="anonymous"></script>

<script *src*="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"

*integrity*="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN"

*crossorigin*="anonymous"></script>

<script *src*="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.min.js"

*integrity*="sha384-VHvPCCyXqtD5DqJeNxl2dtTyhF78xXNXdkwX1CZeRusQfRKp+tA7hAShOK/B/fQ2"

*crossorigin*="anonymous"></script>

Листинг 1. Библиотеки

Веб ресурс создавался с использованием системы сеток фреймворка “Container”, “Container-fluid”, а также дополнительных готовых компонентов фреймворка. Создается база данных на стороне хостинга для получения, записи данных пользователей.

В проект веб ресурса входят 1 страниц в формате html, 1 файл в формате php в папке php.Помимо них, входят папки с изображениями. Страницы веб проекта адаптивны и могут открываться на разных устройствах с разным разрешением экрана.

При переходе по адресу http://srv191622.hoster-test.ru/dip/dip-test.html пользователь попадает на страницу сайта, на которой в шапке расположено закрепленное меню для перехода к нужному разделу на странице. В меню присутствует логотип, кнопки-ссылки и выпадающее подменю со ссылками на конкретные виды услуг подробнее. При нажатии на пункт меню или подменю происходит переход к данному разделу страницы.

    <nav *class*="navbar navbar-expand-lg navbar-light bg-light fixed-top">

    <a *class*="navbar-brand" *href*="#">

      <img *src*="./img/testing.png" *width*="50" *height*="30" *class*="d-inline-block align-top" *alt*="">

      Тестеры

    </a>

    <button *class*="navbar-toggler" *type*="button" *data-toggle*="collapse" *data-target*="#navbarSupportedContent"

*aria-controls*="navbarSupportedContent" *aria-expanded*="false" *aria-label*="Переключатель навигации">

      <span *class*="navbar-toggler-icon"></span>

    </button>

    <div *class*="collapse navbar-collapse" *id*="navbarSupportedContent">

      <ul *class*="navbar-nav mr-auto">

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#what">Что происходит?</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#about">Зачем Вам?</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#why">Почему мы?</a>

        </li>

        <li *class*="nav-item dropdown">

          <a *class*="nav-link dropdown-toggle" *href*="#" *id*="navbarDropdown" *role*="button" *data-toggle*="dropdown"

*aria-haspopup*="true" *aria-expanded*="false">

            Что тестируем

          </a>

          <div *class*="dropdown-menu" *aria-labelledby*="navbarDropdown">

            <a *class*="dropdown-item" *href*="#front">Frontend</a>

            <a *class*="dropdown-item" *href*="#api">API</a>

            <a *class*="dropdown-item" *href*="#back">Backend</a>

            <a *class*="dropdown-item" *href*="#mob">Приложения</a>

            <a *class*="dropdown-item" *href*="#bez">Безопасность</a>

            <a *class*="dropdown-item" *href*="#stress">Нагрузку</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#also">Еще тестируем...</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#refresh">Отзывы и дипломы</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#forma">Форма заявки</a>

        </li>

        <li *class*="nav-item">

          <a *class*="nav-link" *href*="#contact">Как нас найти?</a>

        </li>

      </ul>

      <form *class*="form-inline my-2 my-lg-0">

        <input *class*="form-control mr-sm-2" *type*="search" *placeholder*="Поиск" *aria-label*="Поиск">

        <button *class*="btn btn-outline-success my-2 my-sm-0" *type*="submit">Поиск</button>

      </form>

    </div>

  </nav>

Листинг 2. Меню навигации

Под меню на странице расположен раздел с общей информацией об основных услугах и названием компании.   
Блок с информацией об услуге сделан на основе карточки bootstrap. Карточки располагаются адаптивно в зависимости от размера экрана. Карточка услуги обернута в ссылку на якорь раздела с подробным описанием услуги.

<div *class*="col mb-4">

        <a *href*="#front">

          <div *class*="card ">

            <div *class*=" ">

              <svg *xmlns*="http://www.w3.org/2000/svg" *width*="150" *height*="150" *fill*="currentColor"

*class*="bi bi-window-stack" *viewBox*="0 0 16 16">

                <path

*d*="M4.5 6a.5.5 0 1 0 0-1 .5.5 0 0 0 0 1ZM6 6a.5.5 0 1 0 0-1 .5.5 0 0 0 0 1Zm2-.5a.5.5 0 1 1-1 0 .5.5 0 0 1 1 0Z" />

                <path

*d*="M12 1a2 2 0 0 1 2 2 2 2 0 0 1 2 2v8a2 2 0 0 1-2 2H4a2 2 0 0 1-2-2 2 2 0 0 1-2-2V3a2 2 0 0 1 2-2h10ZM2 12V5a2 2 0 0 1 2-2h9a1 1 0 0 0-1-1H2a1 1 0 0 0-1 1v8a1 1 0 0 0 1 1Zm1-4v5a1 1 0 0 0 1 1h10a1 1 0 0 0 1-1V8H3Zm12-1V5a1 1 0 0 0-1-1H4a1 1 0 0 0-1 1v2h12Z" />

              </svg>

            </div>

            <div *class*="card-body">

              <h5 *class*="card-title">Тестирование Frontend</h5>

              <p *class*="card-text">Профессиональное тестирование вашего веб-интерфейса – убедитесь, что ваш сайт

                работает

                безупречно на любом устройстве и во всех популярных браузерах.</p>

            </div>

          </div>

        </a>

      </div>

Листинг 3. Краткая карточка об услуге

Далее идут разделы с разными карточками и медиа-списком по разному выстроенные с целью заинтересовать и рассказать подробнее о компании посетителям сайта.

На сайте реализована адаптация содержимого под размер экрана («адаптивная верстка») с помощью классов bootstrap в 3 точках останова для маленького размера экрана, среднего и большого. При достижении размера этих точек по максимальной стороне сетка перестраивается.

// Маленький экран / телефон sm: 576px,

// Средний экран / планшет md: 768px,

// Большой экран / настольный lg: 992px,

Реализована стилизация элементов с помощью CSS и псевдоклассов, а так же с помощью свойств в стилях bootstrap. Основная гамма черно-белая с цветовой стилизацией при наведении. Так же использовались градиенты Всем изображениям добавлены текстовые описания для ридеров, для обеспечения доступности.

form p {

      color: red;

    }

.bug2 {

      background: **rgb**(203, 223, 30);

      background: **radial-gradient**(circle, **rgba**(203, 223, 30, 1) 2%, **rgba**(171, 203, 100, 1) 49%, **rgba**(91, 201, 89, 1) 94%);

    }

.inline {

      display: inline;

    }

    .txt-plus,

    .txt-minus {

      margin-left: 5px;

      padding-top: 0px;

      padding-bottom: 0px;

      color: #000;

      text-shadow: white 0 0 10px;

      font-weight: 500;

      border: 3px double darkred;

      background-color: lightgrey;

    }

#also .card:hover h4,

#also .card:hover svg {

      text-shadow: **rgb**(240, 180, 16) 0px 0 15px;

      fill: **rgb**(235, 201, 93);

    }

Листинг 4. Стили

На странице использовано анимационное появление изображения при прокрутке с помощью добавления класса с непрозрачностью и анимирования перехода. В результате чего происходит постепенное появление картинки.

<img *src*="./img/Bag.jpg" *class*="bageits element-animation" *style*="width:100px;" *alt*="баГейтс">

          <script>

            function **onEntry**(entry) {

              entry.**forEach**(change => {

                if (change.isIntersecting) {

                  change.target.classList.**add**('element-show');

                }

              });

            }

            let options = { threshold: [0.5] };

            let observer = new **IntersectionObserver**(**onEntry**, options);

            let elements = document.**querySelectorAll**('.element-animation');

            for (let elm of elements) {

              observer.**observe**(elm);

            }

          </script>

          <style>

            .element-animation {

              opacity: 0;

            }

            .element-animation.element-show {

              opacity: 1;

              transition: all 4s;

            }

          </style>

Листинг 5.1 Анимация со скриптами и стилями

Также есть анимация кнопки на «чистом» CSS стилями, переходами и состояниями, анимируется цвет и размер кнопки.

  .animate-grow {

    animation: example 4s infinite;

}

.animate-grow:hover {

  animation: none;

}

@keyframes example {

    from {background-color: **rgb**(233, 148, 37);}

    to {background-color: orangered;}

  0% {

    transform: **scale3d**(1, 1, 1);

  }

  50% {

    transform: **scale3d**(1.1, 1.1, 1.1);

  }

  100% {

    transform: **scale3d**(1, 1, 1);

  }

}

</style>

Листинг 5.2. Анимация на CSS

В разделе с подробной информацией о каждой услуге на карточках после текста присутствует кнопка «Подробнее», при нажатии на которую сама кнопка исчезает, ниже появляется длинный текст с более подробной информацией и кнопкой «Скрыть», при нажатии на которую текст и кнопка «Cкрыть» исчезают и появляется кнопка «Подробнее». Размер карточки-блока адаптируется под содержание,   
Так же с карточке помимо иллюстрации есть кнопка «Хочу», при нажатии на которую в таблице заказа активируется и появляется поле с данным видов услуги. Аналогичный кнопки «Хочу» есть в карточках дополнительных услуг. Обработчик кнопок каждой карточки реализован отдельно.

btnAddFront.**onclick** = function () {

      trFront.classList.**add**('inline');

    }

btnTxtPlusFront.**onclick** = function () {

      spanTxtFront.**setAttribute**('style', 'display: inline;');

      btnTxtMinusFront.**setAttribute**('style', 'display: inline;');

      btnTxtPlusFront.**setAttribute**('style', 'display: none;');

    }

btnTxtMinusFront.**onclick** = function () {

      spanTxtFront.**setAttribute**('style', 'display: none;');

      btnTxtMinusFront.**setAttribute**('style', 'display: none;');

      btnTxtPlusFront.**setAttribute**('style', 'display: inline;');

    }

Листинг 6. Код кнопок из подробной карточки

В разделе отзывов реализован с применением Bootstrap так называемый аккордион или разворачиваемые панели, с анимацией показывающие отзыв по умолчанию или по клику требуемый отзыв.  
В соседнем разделе с сертификатами реализован слайдер иначе называемый каруселью прокручивающий автоматически картинки с сертификатами и грамотами. Слайдер снабжен кнопками прокрутки вручную в желаемую сторону и баром отображающим место прокручиваемого слайда в карусели, а так же поясняющими подписями к картинкам. Реализованы компаненты с использованием Bootstrap и jquery.

<div *class*="accordion" *id*="accordionExample">

          <div *class*="card">

            <div *class*="card-header" *id*="headingOne">

              <h2 *class*="mb-0">

                <button *class*="btn btn-link btn-block text-left" *type*="button" *data-toggle*="collapse"

*data-target*="#collapseOne" *aria-expanded*="true" *aria-controls*="collapseOne">

                  Отзыв о компании

                </button>

              </h2>

            </div>

            <div *id*="collapseOne" *class*="collapse show" *aria-labelledby*="headingOne" *data-parent*="#accordionExample">

              <div *class*="card-body">

                "Никогда не подозревал, что тестирование может быть таким веселым! Присоединился к Компании

                тестировщиков и оказалось, что они настоящие эксперты в поиске самых невероятных багов! Теперь я ржу до

                слез, читая их тестовые отчеты. Команда крутых ребят, которые знают, как превратить баги в мемы. С

                удовольствием погружаюсь в этот юмористический тестовый мир каждый день!" </div>

            </div>

          </div>

Листинг 7.1. Карточка аккордеона

<div *id*="carouselExampleCaptions" *class*="carousel slide" *data-ride*="carousel">

          <ol *class*="carousel-indicators">

            <li *data-target*="#carouselExampleCaptions" *data-slide-to*="0" *class*="active"></li>

            <li *data-target*="#carouselExampleCaptions" *data-slide-to*="1"></li>

            <li *data-target*="#carouselExampleCaptions" *data-slide-to*="2"></li>

            <li *data-target*="#carouselExampleCaptions" *data-slide-to*="3"></li>

            <li *data-target*="#carouselExampleCaptions" *data-slide-to*="4"></li>

          </ol>

          <div *class*="carousel-inner">

            <div *class*="carousel-item active">

              <img *src*="./img/sramota1.jpg" *class*="d-block w-60 " *alt*="грамота1" />

              <div *class*="carousel-caption d-none d-md-block">

                <p>практикум</p>

              </div>

            </div>

Листинг 7.2. Карточка карусели

Далее следует раздел заказа состоящий из таблицы выбранных услуг и формой для заказа с возможностью оставить сообщение и контактную информацию.

В таблицу выбранные услуги попадают при нажатии выше в разделе с описанием услуги кнопки «Хочу» или при выборе из выпадающего списка услуги при клике по кнопке «Добавить еще тестирование» или при нажатии на соседнюю кнопку «Тестировать всё»  
После нажатия на одну из этих кнопок появляется строка таблицы с названием услуги кнопкой «-» (минус), уменьшающей количество тестировщиков, количеством тестировщиков выполняющих услугу (минимум 1), кнопкой «+» (плюс) увеличивающей количество и кнопкой «Это не надо тестировать», которая убирает данную услугу из таблицы возвращая первоначальное количество. При нажатии «Тестировать всё» происходит добавление всех услуг. Реализована видимость строк таблицы с заказанными услугами с помощью добавления стиля с делающим строку видимой.кнопки плюса и минуса меняют количество занятых в услуге, при попытке уменьшить количество до 0 появляется предупреждающее сообщение реализованное через Alert.

    btnPlusFront.**onclick** = function () {

      let num = tdSpanFront.textContent;

      tdSpanFront.textContent = +num + 1;

    }

    btnMinusFront.**onclick** = function () {

      let num = tdSpanFront.textContent;

      if (num > 1) tdSpanFront.textContent = +num - 1;

      else **alert**('Для тестирования нужен минимум один тестировщик!');

    }

    btnNotFront.**onclick** = function () {

      trFront.classList.**remove**('inline');

      tdSpanFront.textContent = '1';

    }

    btnAddAll.**onclick** = function () {

      trStress.classList.**add**('inline');

      trBez.classList.**add**('inline');

      trMob.classList.**add**('inline');

      trBack.classList.**add**('inline');

      trApi.classList.**add**('inline');

      trFront.classList.**add**('inline');

      trAB.classList.**add**('inline');

      trAD.classList.**add**('inline');

      trMark.classList.**add**('inline');

    }

Листинг 8. Код кнопок таблицы

В форме обратной связи есть поля вводе email, телефона, имени и текста, а также скрытое поле в которое при отправке подтягиваются выбранные и видимые поля из таблицы с заказом. Проверяется обязательность заполнения одного из полей с электронным почтовым адресом или номером телефона, при оставлении полей пустыми и нажатии кнопки отправки формы появляется предупреждающий текст между данными полями. Производится при заполнении поля «Email» проверка присутствия знака «@» и символов до и после, иначе возникает подсказка о не полностью введенном адресе. Заполнение имени и текста сообщения не обязательно. Под полями ввода находится чек-бокс согласия на обработку персональных данных не выбранный. Согласие является кнопкой открытия модального окна с полным текстом соглашения о передаче персональных данных. При попытке отправить не поставив галочку появляется предупреждающая надпись об обязательности согласия.  
К кнопке отправки формы привязана функция отправки формы и функция проверяющая заполнение необходимых полей и подтягивающая в отправляемые в отправляемый запрос данные из таблицы.

**$**(document).**ready**(function () {

**$**('#thisForm').**submit**(function (e) {

        e.**preventDefault**();

        var form = **$**(this);

        var actionUrl = form.**attr**('action');

        $.**ajax**({

          type: "POST",

          url: actionUrl,

          data: form.**serialize**(),

          dataType: "json",

          contentType: "application/x-www-form-urlencoded; charset=UTF-8",

**success**: function (data) { }

        });

**alert**("Cообщение отправлено");

      });

    });

Листинг 9.1. Код функции для отправки формы

btnSubmit.**onclick** = function () {

      let tdSpans = document.**getElementsByClassName**('tdSpan');

      let trs = document.**getElementsByTagName**('tr');

      let arr = [];

      for (let i = 1; i < trs.length; i++) {

        let tdSpn = tdSpans[i - 1];

        let tr = trs[i];

        let boolInline = tr.classList.**contains**('inline');

        if (boolInline) {

          let txtSpan = tdSpn.textContent;

          let idSpan = tdSpn.**getAttribute**("id");

          let strin = "Name testing is " + idSpan + " in number " + txtSpan;

          arr.**push**(strin);

        }

      };

      let msgEmail = document.**getElementById**('email').value;

      let msgTel = document.**getElementById**('tel').value;

      let checkbox = checkbx.**hasAttribute**('checked');

      if (!boolCheck) { smallCheck.**setAttribute**('style', 'display: block;'); }

      if (msgEmail == '' && msgTel == '') { smallEmail.**setAttribute**('style', 'display: inline;'); }

      let arrStr = arr.**join**(', ');

      hidden.**setAttribute**('value', arrStr);

    };

Листинг 9.2. Функции для проверки и обработки отправляемых данных

Данные из формы направляются в файл php расположенный на том же сервере, который обрабатывая полученные поля, убирая пробелы по краям и превращая функцией экранирования возможный код в текст предотвращает возможные угрозы безопасности данных и базы данных. Далее пхп обработчик перенаправляет обработанные данные в базу данных, необходимую для доступа к которой информацию он содержит. Сперва с помощью mysqli происходит подключение к базе данных (неудачное подключение обрабатывается сообщением об ошибке) далее путем вставки данных в требуемую таблицу в базе денных. Данные действия происходят в рамках сессии.

<?php

**session\_start**();

**header**('Content-Type: text/html; charset=UTF-8');

$mysqli = **mysqli\_connect**("mysql-191622.srv.hoster.ru", "srv191622\_admin1", "Gfhjkm1", "srv191622\_bd1");

if ($mysqli == false) {

**print**("повторите отправку формы");

} else {

  $email = **trim**(**mb\_strtolower**($\_POST["email"]));

  $tel = **trim**(**mb\_strtolower**($\_POST["tel"]));

  $name = **trim**(**mb\_strtolower**($\_POST["name"]));

  $hidden = **trim**(**mb\_strtolower**($\_POST["hidden"]));

  $textArea = **trim**(**mb\_strtolower**($\_POST["textarea"]));

  $mysqli->**query**("INSERT INTO `diplom3`(`email`, `tel`, `name`, `text`, `zakaz`) VALUES ('$email', '$tel', '$name', '$textArea', '$hidden')");

**echo** ("success");

}

Листинг 10. Обработчик на PHP

Предварительно на хостинге в phpMyAdmin создана база данных и таблица



Листинг 11. Структура таблицы в Базе Данных

В подвале сайта расположен блок с картой. Использовано API карт Google.

Исходные файлы проекта можно просмотреть на GitHub по ссылке <https://github.com/And0nata/frontend-diplom/tree/main> .

Тестирование работоспособности веб ресурса проводилось не в автоматическом режиме, т.е. без написания тестов.

# СПИСОК ЛИТЕРАТУРЫ

**1. Электронные ресурсы.**

1. Сайт

Документация по фреймворку Bootstrap на русском языке [Электронный ресурс]: офиц. сайт. URL: https://bootstrap-4.ru/ (Дата обращения: 18.07.2023).

2. Сайт

Официальная документация Google [Электронный ресурс]: офиц. сайт. URL: https://www.google.com/ (Дата обращения: 18.07.2023).

3. Сайт

Официальная документация ресурса сообщества разработчиков stackoverflow [Электронный ресурс]: офиц. сайт. URL: https://ru.stackoverflow.com/ (Дата обращения: 18.07.2023).

4. Сайт

Электронный учебник по HTMLи CSS[Электронный ресурс]: офиц. сайт. http://htmlbook.ru/html5 (Дата обращения: 19.07.2023).

5. Сайт

Официальная документация Mozilla [Электронный ресурс]: офиц. сайт. URL: https://developer.mozilla.org/ru/ (Дата обращения: 18.07.2023).

6. Сайт

Современный учебник JavaScript [Электронный ресурс]: офиц. сайт. URL: https://learn.javascript.ru/ (Дата обращения: 18.07.2023).

7. Сайт

Сайт о программировании metanit.com [Электронный ресурс]: офиц. сайт. URL: https://metanit.com/ (Дата обращения: 18.07.2023).

8. Сайт

Сайт о jquery [Электронный ресурс]: офиц. сайт. URL: https://api.jquery.com/ (Дата обращения: 18.07.2023).