

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**Разработка веб-приложения на основе ASP.NET для автоматизации
работы автосервиса**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 441 группы

направления 02.03.03 Математическое обеспечение и администрирование
информационных систем

факультета компьютерных наук и информационных технологий

Труханова Германа Григорьевича

Научный руководитель,

зав. кафедрой, д.ф-м.н

Андрейченко Д.К.

подпись, дата

Зав. кафедрой МОВКИС,

д.ф-м.н

Андрейченко Д.К.

подпись, дата

Саратов 2020

ВВЕДЕНИЕ

Цифровизация и автоматизация с каждым днем все больше проникает в большинство сфер нашей жизни, поэтому не удивительно, что бизнес, начиная от мелкого и заканчивая крупным, заинтересован в развитии IT технологий и интеграции с ними. Сегодня трудно представить свою жизнь без возможности, что-либо найти и заказать в интернете, поэтому многие предприятия стараются выйти на рынок интернет магазинов, чтобы стать доступнее для потребителя, так как практически каждый человек владеет смартфоном или любым другим устройством с выходом в интернет.

Многие автомобилисты всегда имеют под рукой телефон с доступом в интернет, а некоторые автомобили имеют собственный бортовой компьютер, который тоже способен на работу в интернете, поэтому сегодня есть потребность в ресурсе, который бы давал автомобилисту возможность на запись и оказание ремонтных услуг в автосервисе при помощи обычного браузера.

Целью бакалаврской работы является разработка веб-приложения для оптимизации работы автосервиса, которое адаптировано для использования на устройствах с различным размером дисплея.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучить следующие технологии, необходимые для создания приложения:
 - ASP.Net MVC
 - Entity Framework
 - ASP.Net Identity
 - IoC контейнер Ninject
 - Bootstrap
 - JQuery и JQuery Validation
 - Ajax
2. Создать веб-приложение и оптимизировать его под мобильные устройства.
3. Разместить готовое веб-приложение на ресурсе для хостинга и проверить его работоспособность.

В теоретической части изложена общая информация об архитектурном паттерне MVC, а также о компонентах и технологиях, которые используются для создания серверной и клиентской части веб-приложения.

В практической части описан процесс разработки моделей, контроллеров и представлений целевого веб-приложения, а также процесс его публикации на хостинг ресурсе. Данная часть также содержит подробное описание процесса и результаты тестирования готового веб-приложения.

Структура и объём работы. Бакалаврская работа состоит из введения, трех разделов, заключения, списка использованных источников и четырех приложений. Общий объем работы – 87 страниц, из них 73 страницы – основное содержание, включая 17 рисунков и 3 таблицы, цифровой носитель в качестве приложения, список использованных источников информации – 22 наименования.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Серверная часть приложения» содержит общую информацию об архитектурном паттерне MVC, технологии для создания веб-приложений ASP.NET MVC 5, а также вспомогательных компонентах: Entity Framework, IoC-контейнере Ninject и библиотеке ASP.NET Identity.

Аббревиатура MVC расшифровывается как модель-представление-контроллер (от англ. model-view-controller). Это способ организации кода в проекте по принципу разделения на блоки, которые должны решать разные задачи. Один блок отвечает за работу с данными приложения, другой отвечает за внешний вид, а третий контролирует работу приложения. Данный архитектурный паттерн широко используется во многих существующих веб-приложениях благодаря своей структуре, которая позволяет облегчить поддержку существующего кода, а также упростить процесс масштабирования приложения.

Одной из существующих технологий, которая следует принципам MVC, является фреймворк от компании Microsoft под названием ASP.NET MVC.

Данный фреймворк использует язык программирования C#, а также механизм представлений Razor, который упрощает процесс разработки представлений. Общую информацию об ASP.NET можно разделить на три логических блока в соответствии с аббревиатурой MVC, а именно на модели, представления и контроллеры.

Модели в ASP.NET это обыкновенный класс языка C#, который может содержать классические элементы, такие как свойства, конструкторы, методы и так далее. Модели в MVC описывают некую логику данных, а также позволяют корректно взаимодействовать с базой данных приложения. Для облегчения процесса создания и взаимодействия с моделями в ASP.NET существуют определенные метаданные, которые указывают на некоторую дополнительную информацию об объекте, например, о том, как отображать его свойства в представлении, или о том, как осуществлять валидацию вводимых данных.

Представления в ASP.NET это некий файл с расширением cshtml, который содержит разметку на языке HTML, стили CSS, а также скрипты, написанные на языке JavaScript. Основная роль представлений в веб-приложении – это взаимодействие с пользователем.

Контроллеры в ASP.NET являются центральным компонентом, который получает ввод пользователя, обрабатывает его, а затем, в качестве ответа, возвращает, например, представление. Контроллер – это обычный класс языка C#, который наследуется от базового класса `System.Web.Mvc.Controller`.

Паттерн MVC подразумевает, что основная логика веб-приложения будет содержаться в контроллерах. Такой подход может сделать код приложения трудно модифицируемым. Решить данную проблему и отделить бизнес логику приложения от функционала контроллера позволяет принцип внедрения зависимостей и IoC-контейнер Ninject. **Inversion of Control** (инверсия управления) – это некий абстрактный принцип или набор рекомендаций для написания слабо связанного кода, суть которого заключается в том, что каждый компонент системы должен быть как можно более изолированным от других, не полагаясь в своей работе на детали конкретной реализации других компонентов.

Dependency Injection (внедрение зависимостей) – это одна из реализаций принципа инверсии управления. **IoC-контейнер Ninject** – это фреймворк, который позволяет упростить написание кода с использованием принципа инверсии управления на столько, на сколько это возможно. Вынести логику приложения из структуры контроллера с помощью Ninject можно в несколько этапов:

- Подключить фреймворк Ninject к проекту
- Сформировать набор интерфейсов и их реализаций, которые содержат необходимую логику для контроллеров.
- Связать интерфейсы и реализации в ядре Ninject

В конечном счете, данные манипуляции позволяют изменять бизнес логику приложения, не затрагивая общий функционал контроллеров и приложения в целом.

Современное веб-приложение невозможно представить без быстрого и хорошо спроектированного хранилища данных. База данных для целевого веб-приложения создана и функционирует при помощи фреймворка **Entity Framework**. Entity Framework представляет специальную объектно-ориентированную технологию на базе фреймворка .NET для работы с данными. Центральной концепцией Entity Framework является понятие сущности или entity. Сущность представляет набор данных, ассоциированных с определенным объектом, поэтому данная технология предполагает работу не с таблицами, а с объектами и их наборами.

Одним из нововведений ASP.NET MVC 5, стала новая система авторизации и аутентификации в .NET приложениях под названием ASP.NET Identity. Эта система пришла на смену провайдерам Simple Membership, которые были введены в ASP.NET MVC 4. Предыдущая система авторизации предполагала, что пользователь проходил процедуру авторизации, за счет той пары логин/пароль, которая храниться в базе данных веб приложения, но

развитие веб – технологий не стоит на месте, поэтому система требовала обновления. Одним из нововведений, добавленных вместе с новой системой авторизации и аутентификации, стала возможность входа при помощи социальных сетей, таких как Google+, Facebook, Twitter и т.д

Фреймворк ASP.NET MVC 5 в связке с Entity Framework и IoC-контейнером Ninject позволяют создать качественное веб-приложение, соответствующие современным нормам и требованиям пользователей.

Второй раздел «Клиентская часть приложения» содержит общие сведения о технологиях, которые используются для создания клиентской части целевого веб-приложения.

На сегодняшний день практически каждый человек на планете владеет устройством, начиная от классического настольного ПК и заканчивая мобильным телефоном, способным функционировать в глобальной сети интернет. Для того чтобы веб-приложение было адаптивно для экрана любого устройства сообщество придумало множество инструментов и фреймворков для решения данной задачи.

Bootstrap – это открытый и бесплатный HTML, CSS и JS фреймворк, который используется веб-разработчиками для быстрой вёрстки адаптивных дизайнов сайтов и веб-приложений. Популярность Bootstrap связана с тем, что он позволяет верстать сайты в несколько раз быстрее, чем это можно выполнить на «чистом» CSS и JavaScript. Также его популярность связана с доступностью, так как на нём даже начинающий разработчик может верстать достаточно качественные макеты, которые трудно было бы выполнить без глубоких знаний веб-технологий и достаточной практики.

Одним из полезных фреймворков, который часто используют в паре с Bootstrap, является JQuery. jQuery – библиотека JavaScript, содержащая в себе готовые функции языка JavaScript, все операции jQuery выполняются из кода JavaScript. Библиотека jQuery производит манипуляции с HTML-элементами, управляя их поведением и используя DOM для изменения структуры веб-

страницы. При этом исходные файлы HTML и CSS не меняются, изменения вносятся лишь в отображение страницы для пользователя.

Валидация форм и проверка их полей на соответствие определенному формату и наличие определенных данных – это стандартная задача при работе с формами на сайте, которую программист может решить различными способами. Одним из самых простых решений является использования плагина JQuery Validation, который предоставляет программисту готовый функционал, необходимый для валидации вводимых пользователем данных.

Вышеперечисленный набор инструментов для веб-разработки позволяет программисту создавать современные и хорошо адаптирующийся под размеры экрана большинства устройств веб-приложения.

Третий раздел «Создание приложения на основе ASP.NET MVC 5» посвящен реализации целевого веб-приложения на основе технологии ASP.NET MVC 5.

Целевое веб-приложение создается для автоматизации работы автосервиса, а также для удобства использования его множеством обыкновенных автолюбителей. Приложение должно быть адаптировано под работу на устройствах с небольшим размером экрана, а именно: на смартфонах, планшетах и иных устройствах с диагональю экрана от 5 до 7 дюймов.

С точки зрения пользователя сайт должен быть похож на обыкновенный интернет магазин, который будет предоставлять услуги по обслуживанию автомобиля. Сайт предоставляет следующий набор возможностей:

- Просмотр стартовой страницы, на которой будет содержаться вся основная информация о сайте и его разделах, событиях, связанных с ним, а также скидки на услуги, если таковые имеются.
- Наличие раздела, который будет содержать все услуги, предоставляемые автосервисом с возможностью фильтрации и поиску по имени.

- Наличие корзины, где можно посмотреть текущий набор услуг, а также составить заказ, который будет выполнен автосервисом в указанный срок.
- Наличие раздела, в котором пользователь может корректировать информацию о себе и своих автомобилях, а также просматривать и редактировать текущие заказы.
- Наличие раздела с контактной информацией, где пользователь может посмотреть на адрес автосервиса, номера горячей линии, а также почтовые адреса.
- Наличие раздела с отзывами других пользователей, которое позволит составить первое впечатление о качестве оказываемых услуг.

Также одним из очевидных требований к сайту является его понятность. Пользователь должен понимать, что и зачем он делает на этом сайте и что ему нужно сделать, чтобы достигнуть определенных им целей.

С точки зрения администрации сайта, веб-приложение должно удовлетворять следующим требованиям по функционалу:

- Возможность расширить ассортимент предоставляемых услуг
- Возможность предоставить скидки на определенный ассортимент услуг
- Возможность редактирования контактной информации автосервиса
- Возможность обновлять информацию главной страницы, а также генерировать события, связанные с автосервисом.
- Наличие раздела сайта, где можно производить манипуляции с содержимым сайта.
- Функционал для просмотра не завершенных заказов с возможностью фильтрации данных
- Наличие архива, который хранит все завершенные заказы пользователей, с возможностью фильтрации данных.
- Возможность производить модерирование отзывов пользователей

С точки зрения администрации сайта доступ к настройкам и содержимому сайта должна быть только у администратора, при этом администратор не должен обладать функционалом, который позволял бы ему редактировать личную информацию пользователя. Также общим требованием к веб-приложению будет его размещение на общедоступном хостинге, чтобы любой человек, находящийся в России мог пользоваться данным приложением.

В соответствии с предполагаемыми требованиями был выбран и модифицирован исходный шаблон веб-приложения ASP.NET MVC, предоставленный средой разработки Visual Studio 2019. Помимо модификации исходного шаблона веб-приложения, сайт был разбит на необходимое количество тематических разделов, каждый из которых в основе имеет соответствующий контроллер. Список разделов сайта состоит из следующих наименований:

- Главная
- Услуги
- Вакансии
- Контакты
- Отзывы
- Аккаунт (для пользователя)
- Корзина (для пользователя)
- Настройки (для администрации)
- Новости (для администрации)

Для каждого раздела было создано необходимое для функционирования количество моделей и представлений, а также, если необходимо, настроен процесс валидации данных вводимых пользователем.

Готовое веб-приложение было подготовлено и размещено на хостинг ресурсе Reg.ru и протестировано по нескольким направлениям:

- Тестирование сценария нормального поведения пользователя
- Тестирование функционала администратора
- Тестирование адаптивности сайта

ЗАКЛЮЧЕНИЕ

Современные веб-технологии позволяют создавать удобные и стильные веб-приложения, которые способны адаптироваться под большинство экранов современных мобильных устройств. Благодаря технологии Asp.Net MVC, а также множеству фреймворков для фронтенд разработки, появляется возможность создать приложение, которое способно облегчить жизнь обычным водителям и позволяют выходить малым предприятиям на новый уровень.

Основные источники информации:

1. Чамберс Д. Asp.Net Core Разработка приложений / Джеймс Чамберс, Дэвид Пэккеттб, Саймон Тиммс.; Санкт-Петербург: Издательский дом Питер,2018. 464с.
2. Фримен А. ASP.NET MVC 5 с примерами на C# 5.0 для профессионалов / Адам Фримен.; Москва: Издательский дом Вильямс,2018. 736с.
3. Что такое Bootstrap и зачем он нужен?. [Электронный ресурс]: / itchief.ru. Электрон. дан. – URL: <https://itchief.ru/bootstrap/introduction> – Загл. с экрана.
4. Введение в ASP.NET MVC 5. [Электронный ресурс]: / metanit.com. Электрон. дан. – URL: <https://metanit.com/sharp/mvc5/1.1.php> – Загл. с экрана.
5. Введение в ASP.NET Identity. [Электронный ресурс]: / docs.microsoft.com. Электрон. дан. – URL: <https://docs.microsoft.com/ru-ru/aspnet/identity/overview/getting-started/introduction-to-aspnet-identity> – Загл. с экрана.