Изм.

Лист

№ докум.

Подпись

Дата

Лист

4

кр.ПМ.03.МДК.03.01.09.02.03.19.17.00.00.пз

Разраб.

Шабанов И.Ф.

Провер.

Шепилова А.А.

Н. Контр.

Утверд.

Разработка Web-сайта туристического агентства

Лит.

Листов

21

4-П-2

# Содержание

[1](#_Toc2517841)

[2](#_Toc2517842)

[3](#_Toc2517843)

[Введение 5](#_Toc2517844)

[1 Постановка задачи 6](#_Toc2517845)

[2 Описание предметной области 8](#_Toc2517846)

[3 Структура базы данных 8](#_Toc2517847)

[4 Описание алгоритмов обработки данных 10](#_Toc2517848)

[5 Обоснование выбора программных и аппаратных средств для реализации поставленной задачи 11](#_Toc2517849)

[6 Описательная часть разработанного приложения (базы данных) 12](#_Toc2517850)

[7 Программный код с пояснениями 13](#_Toc2517851)

[8 Методы тестирование и отладки 13](#_Toc2517852)

[Заключение 14](#_Toc2517853)

[Список использованных источников 15](#_Toc2517854)

[Приложения 18](#_Toc2517855)

[Приложение 1.Техническое задание 18](#_Toc2517856)

# 

# Введение

Курсовая работа – мероприятие, направленное на восполнение отсутствующих, систематизацию имеющихся и накопление новых, необходимых для выполнения (самостоятельно выбранного) задания знаний, а также, это замечательная возможность закрепить и отработать полученные в ходе обучения умения на практике. К сожалению, неиспользуемый материал очень быстро «стирается из памяти» (забывается), курсовая работа – возможность получить необходимые, комплексные знания, необходимые для её выполнения. IT–сфера оказала огромное влияние на современный бизнес, невооружённым глазом заметна повсеместная тенденция к автоматизации. Существует множество направлений, в которых задействованы IT–специалисты: проектирование и разработка веб–сайтов (вёрстка удобных интерфейсов и разработка функционала сайта), создание мобильных и настольных приложений, а также множество смежных специальностей, которые сотрудничают бок о бок на благо одной цели – проектировщики программного обеспечения, администраторы баз данных, системные аналитики (DevOps'ы – стремящиеся сократить переход с одной стадии жизненного цикла программного обеспечения к другой, не теряя, а даже улучшая качество разрабатываемого продукта), разработчики интерфейсов (UX&UI–дизайнеры), тестировщики и другие. Хочется отметить и то, что студент имел возможность самостоятельно выбрать тему курсовой работы, по моему мнению, это является исключительно хорошей практикой, так как каждый студент может начать (продолжить) развиваться в интересном ему направлении. На данный момент, каждому верстальщику необходимо иметь в своём арсенале опыт работы со следующими технологиями: Photoshop, стандартом HTML5, CSS3 и jQuery (JavaScript (ES6)), либо TypeScript, в том случае, если ему предстоит работать с фреймворком Angular. Фреймворки используются для создания веб–приложений. Верстальщику же, для трудоустройства (успешного фриланса), сейчас, достаточно знания библиотеки jQuery. Курсовая работа является важным этапом в формировании конкурентоспособного специалиста.

# 1 Постановка задачи

Планирование – важный процесс, позволяющий смоделировать многие детали будущей информационной системы или веб–сайта, не приступая к этапу программирования: интерфейс, функциональные возможности, используемые технологии и многое другое.

При отсутствии хорошего планирования (используемых технологий, интерфейса, функциональных возможностей проекта, требований к скорости выполнения), проект часто получается совсем не таким, каким изначально задумывался. Повышается сложность его разработки и сопровождения, так как программисту неизвестно в каком направлении двигаться. Хорошее, детальное планирование – возможность максимально близко приблизиться к желаемому результату.

Несколько цитат: «Кто не знает, куда направляется, очень удивится, попав не туда» – Марк Твен.

«Очень сложно прийти туда, куда нужно, если вы не знаете, чего конкретно хотите» – zirina479.livejournal.com/519609

«Кто не знает, куда идет, вероятно, придет не туда» – Питер Лоуренс.

Задача технического задания – это свести к минимуму разницу между представлениями о проекте обеих сторон: заказчика и исполнителя. Один раз хорошо обсудив все детали проекта, обе стороны останутся в плюсе: у программиста сократится количество лишней работу, а заказчик, в свою очередь быстрее получит желаемый результат, ведь останется больше времени на правку.

Жизненный цикл программного обеспечения

ISO/IEC 12207:2008 Systems and software engineering — Software life cycle processes – (международный) стандарт ISO, описывающий процессы жизненного цикла программного обеспечения. Данный стандарт, используя устоявшуюся терминологию, устанавливает общую структуру процессов жизненного цикла программных средств, на которую можно ориентироваться в программной индустрии. Стандарт определяет процессы, виды деятельности и задачи, которые используются при приобретении программного продукта или услуги, а также при поставке, разработке, применении по назначению, сопровождении и прекращении применения программных продуктов.

До начала разработки веб – сайта, следует разбить жизненный цикл веб –сайта на несколько этапов, а также постараться представить, что и как будет реализовано на каждом временном промежутке. Также, следует оценить какие результаты должны быть достигнуты при помощи созданного функционала (какие должны быть использованы технологии, желаемое время выполнения имеющихся функций и так далее).

# 2 Описание предметной области

Веб–сайт–прекрасная возможность донести до потенциального клиента сведения об оказываемых компанией услугах (в данном случае, о предлагаемых вариантах экскурсий), организовать и упростить обратную связь. Таким образом компания увеличивает шанс на получение дополнительной прибыли и снимает с себя часть работы, отвечая на часто задаваемые вопросы.

# 3 Структура веб–сайтов

Веб–сайт – это совокупность веб–страниц, объединённых, общим дизайном (стилем). Обычно, файлы сайта хранятся либо в одной директории (папке), либо в подкаталогах основной папке, соответственно их типу, например, мы можем хранить изображения в папке img (images), стили в папке styles и так далее.

Раньше интернет-сайты создавали на чистом HTML, и это было неудобно, так как все данные были представлены как отдельные HTML-файлы. Нельзя было осуществлять поиск, группировку, сортировку информации. К тому же, информация могла часто дублироваться. При появлении PHP (JavaScript и вариантов его исполнения в виде различных фреймворков) у веб-мастеров появилась возможность разделения сайта на его каркас и данные в базе. Теперь структуру сайта можно хранить отдельно от контента, что позволяет быстрее и удобнее администрировать веб-ресурс, легко дорабатывать его дизайн и функционал.

Структура веб–ресурса хранится в коде или в отдельных шаблонах (специальных файлах). Контент размещается в базе данных – определенном наборе таблиц с однотипными данными.

Допустим, необходимо создать обычный сайт–визитку. У разрабатываемого ресурса будет своя, отдельная структура и база данных. В базе будут представлены несколько таблиц: первая – с содержимым страниц, вторая – с новостной лентой, третья – с фотогалереей.

4 Принцип коммутации с веб–сервером, на котором хранится веб–сайт

Как уже говорилось ранее, выполнение многих процессов становится всё более автоматизированным и менее времязатратным, так, например, пользователи при открытии браузера, вводят в адресной строке не ip–адрес, состоящий из цифр, а название нужного сайта. Данную возможность обеспечивает система доменных имён (DNS), которая хранит соответствие имени сайта его ip–адресу в виде таблицы. Возможно, каждый пользователь сталкивался с тем, что при первом обращении к тому или иному сайту, процесс полной загрузки сайта длится дольше, чем последующие. Это объясняется тем, что веб–браузер кэширует (сохраняет) на компьютер пользователя необходимые для загрузки страницы данные, в том числе информацию об ip–адресе сайта, что, также, обеспечивает более быстрый доступ к нужному сайту. В случае, если на компьютере нет информации об ip–адресе запрашиваемого сайта, происходит обращение к resolver’у (ISP–провайдеру), каждый интернет–провайдер знает адрес корневого сервера, именно resolver (в порядке очереди) обращается к корневому серверу, который хранит информацию о доменах верхнего уровня (.com, .org, .ru и так далее). Всего, на данный момент существует 13 корневых серверов.

Веб-приложения

Веб-приложение–клиент-серверное приложение, в котором клиент взаимодействует с сервером при помощи браузера, а за сервер отвечает веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными службами.

Любое веб приложение состоит из двух частей: Back-End и Front-End. Соответственно и программистов делят также, по сути это две несвязанные между собой касты разработчиков.

Бекэнд–серверная часть приложения, которая не видна пользователям. Сюда относится: авторизация, хранение и обработка данных, email рассылки и тому подобное. Скажем так бекэнд это то, что под капотом. Языков программирования тут множество, вот самые распространенные из них:

1. PHP
2. ASP.NET
3. Java
4. Python
5. Ruby
6. Node.js

С текущими требованиями к программистам, для достижения уровня middle пройдет не менее нескольких лет, поэтому, следует сосредоточиться на одном.

Сервер во многом используется для работы с данными. Поэтому нужно будет выучить как минимум бы одну из реляционных баз данных:

1. Oracle
2. Microsoft SQL Server
3. MySQL
4. PostgreSQL

По сложности это примерно тоже, самое, что и освоение языка программирования. Во всех перечисленных БД есть часть, которая одинакова везде–так называемый SQL (язык запросов), а также, процедурный язык который в каждой базе свой. Так как SQL все–таки един, а он основа всему, что связано с обработкой данных, то переход на другую базу данных, не является такой уж и проблемой. Изучив, необходимый набор технологий единожды, на них можно проработать длительное время.

Язык программирования тут один JavaScript, но есть свои нюансы. JavaScript на текущий момент времени имеет два стандарта, которые широко используются: ECMAScript 5 и ECMAScript 6. Разница между ними состоит в том, что в ECMAScript 5 отсутствует полноценное ООП. В ECMAScript 6 оно есть, но поскольку этот стандарт появился относительно недавно, уже большая часть приложений написана на ECMAScript 5, в результате чего нужно будет освоить оба стандарта и как следствие два разных подхода к разработке.

Еще один момент, который нужно учитывать это то что один и тот же код, написанный на JavaScript может по-разному отрабатывать в разных браузерах и на разных ОС. С текущими требованиями к приложениям как правильно нужно будет писать так называемый кроссбраузерный код, то есть код, результат выполнения которого будет давать везде один и тот же результат.

Описанные проблемные моменты решены в современных JavaScript фреймворках (например, в таких как Angular и React). Но, в проектах, которые написаны без использования фреймворков, эти моменты остаются.

Для Front-End характерно очень быстрое развитие трендов. Постоянно обновляются версии фреймворков, появляются новые библиотеки.

Три технологии, которыми должен владеть каждый веб–разработчик

HTML

HTML (от английского HyperText Markup Language – «язык гипертекстовой разметки») – стандартизированный язык разметки документов во Всемирной паутине. Большинство веб–страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства. Не является языком программирования. Технология CSS позволяет придать нужный вид (стиль) созданной с помощью HTML разметке.

Система управления базами данных MS Access–это набор инструментов конечного пользователя для управления базами данных. В ее состав входят конструкторы таблиц, форм, запросов и отчетов. Эту систему можно рассматривать и как среду разработки приложений. Используя макросы или модули для автоматизации решения задач, можно создавать ориентированные на пользователя приложения такими же мощными, как и приложения, написанные непосредственно на языках программирования.

обработки в полях не оказывается никаких значений, система обеспечивает полную поддержку пустых значений.

База данных содержит одну таблицу Меню:

Меню (рисунок 1) с полями (Код меню, наименование, стоимость, описание).

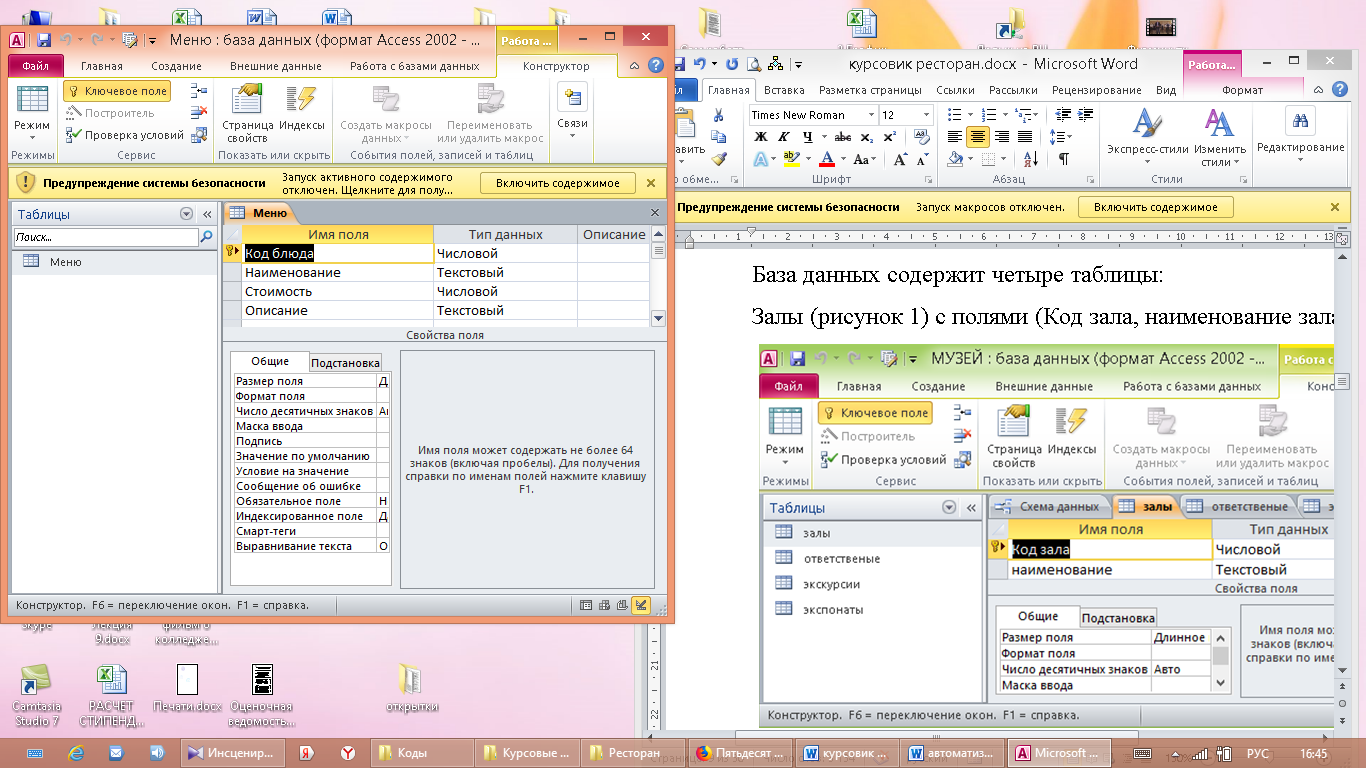


Рисунок 1 – Таблица Меню в режиме Конструктор

На рисунке 2 изображена таблица Меню в режиме ввода записей.

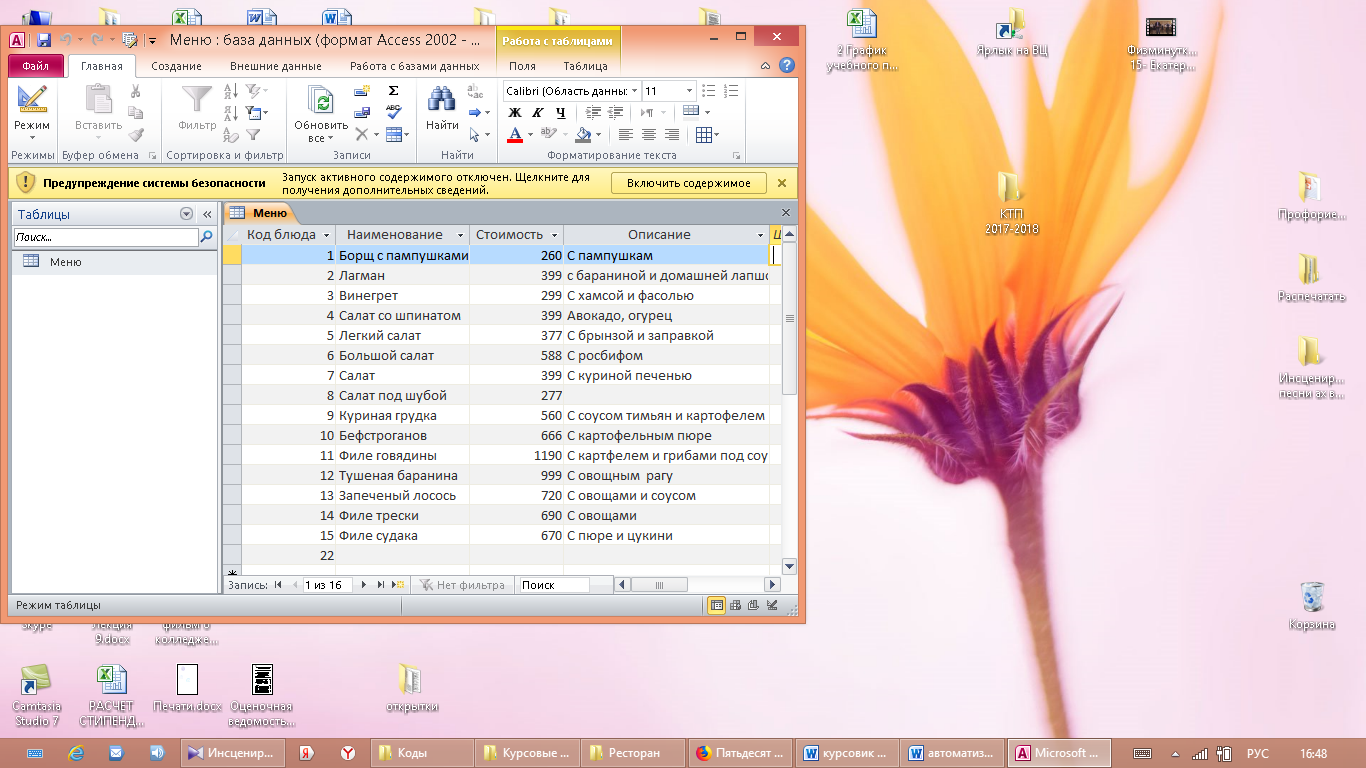


Рисунок 2 – Таблица Меню в режиме ввода записей

# 

# 4 Описание алгоритмов обработки данных

Алгоритм обработкиданных во многих случаях бывает задан. Однако при

программный, был доработан более детально.

# 5 Обоснование выбора программных и аппаратных средств для реализации поставленной задачи

поддаются изменению.

Большинство систем доступа к данным на языке C++ либо рассчитаны на базы данных определенного поставщика, либо требуют значительной доработки для каждой конкретной базы или источника данных, либо используют неэффективные "универсальные" методы подключения к данным. Платформа C++Builder DataSnap® и драйверы доступа к данным dbExpress™ позволяют создать одну систему и с ее помощью легко получать доступ к различным источникам данных. При этом обеспечивается поддержка специфических функций баз данных разных поставщиков без ухудшения их быстродействия, производительности и управляемости.

6 Описательная часть разработанного приложения (базы данных)

Удобнее всего работать с базой данных Access через ADO Connection.

* ADOTable;
* DataSource;
* DBGrid.

Главное окно программы содержит в себе компонент, для отображения данных в виде таблицы, панель управления данными, панель переключения страниц, панель ввода данных и статус панель, представлены на рисунках 9, 10.

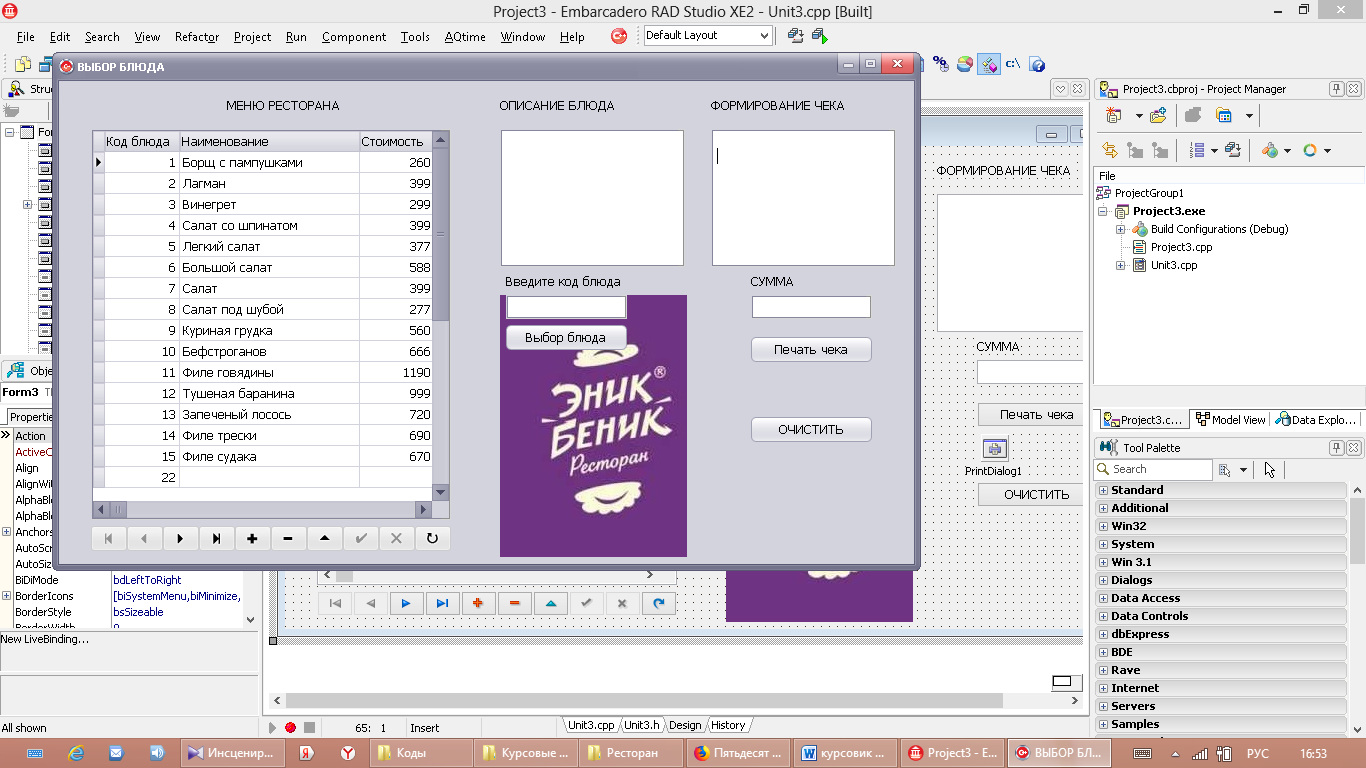


Рисунок 9 – Главное окно программы

Рисунок 10 – Форма программы в режиме проектирования

Для поиска информации использовался метод Locate. На форме помещены компоненты TComboBox, TButton и TEdit, в КомбоБокс внесены названия всех полей таблицы (для этого использовалось свойство Items). В Edit б вводиться значение необходимое для поиска, а из ComboBox выбирается поле, по которому осуществлять поиск.

Рисунок 9 – Форма программы с Поиском информации

# 7 Программный код с пояснениями

<!DOCTYPE html>

<html lang="ru–RU">

<head> <!— начало контейнера со служебной информацией (метаданными) ––>

<meta http–equiv="X–UA–Compatible" content="IE=edge"/>

<meta charset="UTF–8"/>

<title>Аквамарин</title>

<meta name="viewport" content="width=device–width, initial–scale=1.0"/> <meta name="Keywords" content="экскурсии,экскурсии новороссийск,экскурсии Крым,экскурсии по новороссийску,экскурсии геленджик,экскурсии Анапа,Горгиппия,

Горгиппия анапа, гора Митридат, Аквамарин"/>

<meta name="description" content="Страница экскурсионной компании Аквамарин."/>

<link href="styles/styles.css" rel="stylesheet"/>

</head>

# 

# 8 Методы тестирование и отладки

# Заключение

Курсовая работа является важным этапом учебного процесса, способствующим началу самостоятельной, исследовательской работы каждого студента в выбранном им направлении, формировании необходимых каждому разработчику навыков: умение работать с техническим заданием, проектирование внешнего вида и логики будущей информационной системы, кодирование, тестирование, работа с документацией и её последующая разработка, анализ возможных способов решения поставленной задачи (на основании эффективности существующих алгоритмов и используемых технологий, их отладка и модификация).

Иначе говоря, во время написания курсовой работы, студент погружается в рабочую атмосферу: стадии, включающие в себя последовательные этапы разработки программного обеспечения, с которыми сталкивается каждый разработчик (это особенно актуально для разработчиков, работающих в команде). Этапы разработки программного обеспечения на профессиональном языке называют циклом разработки программного обеспечения.

# Список использованных источников

1. Макфарланд Д. «Новая большая книга CSS» – СПб.: Питер, 2016;
2. Гоше Х.Д. «HTML5. Для профессионалов» – СПб.: Питер, 2014;

Материал из сети Internet

1. Tproger – создано программистами для программистов [Электронный ресурс]. – URL: https://tproger.ru/
2. Википедия – свободная энциклопедия [Электронный ресурс]. – URL: https://ru.wikipedia.org/
3. Справочники веб–дизайнера – ценные русскоязычные ресурсы для тех, кто изучает веб–технологии и создаёт сайты: htmlbook.ru, WebReference, HTML5BOOK.RU – [Электронные ресурсы]. – URL: htmlbook.ru, webref.ru, html5book.ru
4. Бесплатный образовательный веб–сайт для изучения веб–технологий (и не только) в Интернете [Электронный ресурс]. – URL: https://www.w3schools.com/
5. Coursera – образовательный онлайн–проект, позволяющий получить доступ к урокам от самых лучших университетов мира [Электронный ресурс]. – URL: https://www.coursera.org/
6. Новейшие рекомендации и техническая спецификация по различным веб–технологиям [Электронный ресурс]. – URL: https://www.w3.org/TR/
7. Разработка адаптивных сайтов, используя технологию Flexbox (обучение в игровой форме) [Электронный ресурс]. – URL: http://flexboxfroggy.com/#ru
8. Разработка адаптивных сайтов, используя (набирающую популярность) технологию CSS Grid (обучение в игровой форме) [Электронный ресурс]. – URL: https://cssgridgarden.com/#ru
9. Хабрахабр – интересные публикации и новости из мира IT [Электронный ресурс]. – URL: https://habr.com/ru/

Список стандартов по курсовому проектированию

В настоящем положении использованы ссылки на следующие стандарты:

ГОСТ 2.004–88 ЕСКД. Общие требования к выполнению конструкторских и технологических документов на печатающих и графических устройствах вывода ЭВМ

ГОСТ 28388–89 Системы обработки информации. Документы на магнитных носителях данных. Порядок выполнения и обращения

ГОСТ 2.104–2006 ЕСКД. Основные надписи

ГОСТ 2.105–95 ЕСКД. Общие требования к текстовым документам

ГОСТ 2.106–96 ЕСКД. Текстовые документы

ГОСТ 3.1102–81 Стадии разработки и виды документов.

ГОСТ 3.1103–82 Основные надписи

ГОСТ 3.1109–82 ЕСТД. Термины и определения основных понятий

ГОСТ 7.1–2003 Библиографическая запись. Библиографическое описание. Общие требования и правила.

ГОСТ 7.12–93 Библиографическая запись. Сокращение слов на русском языке. Общие требования и правила.

ГОСТ 19.101–77 ЕСПД. Виды программ и программных документов

ГОСТ 19.201–78 ЕСПД. Техническое задание. Требования к содержанию и оформлению

ГОСТ 19.401–78 Текст программы. Требования к содержанию и оформлению

ГОСТ 34.601–90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания

ISO/IEC 12207:2008 «System and software engineering – Software life cycle processes» (российский аналог – ГОСТ Р ИСО/МЭК 12207–2010 Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств).

# Приложения

# Приложение 1.Техническое задание

МИНИСТЕРСТВО ОБРАЗОВАНИЯ НАУКИ И МОЛОДЕЖНОЙ ПОЛИТИКИ КРАСНОДАРСКОГО КРАЯ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ КРАСНОДАРСКОГО КРАЯ

«НОВОРОССИЙСКИЙ КОЛЛЕДЖ РАДИОЭЛЕКТРОННОГО

ПРИБОРОСТРОЕНИЯ»

(ГБПОУ КК «НКРП»)

УТВЕРЖДАЮ

Председатель ЦК

О.А. Афиногенова

\_\_.\_\_.2019

Разработка Web–сайта туристического агентства

Техническое задание на курсовой проект

Листов 3

Руководитель, преподаватель \_\_\_\_\_\_\_\_\_\_ А.А. Шепилова

Исполнитель, студент 4–П–2\_\_\_\_\_\_\_\_\_\_\_ И.Ф. Шабанов

Новороссийск, 2019

1. Введение

Наличие веб – сайта – возможность продвижения собственного бизнеса в сети Интернет (получение новых клиентов за счёт (SEO–продвижения) рекламы), обеспечение обратной связи, ответы на часто задаваемые вопросы и многое другое. Правильное использование веб–технологий разработчиком (верстальщиком) позволяет одинаково хорошо отображать сайт на устройствах, независимо от установленной операционной системы (кроссплатформенность), поддерживающих самое различное разрешение экрана: начиная от экранов мобильных телефонов, планшетных компьютеров и заканчивая самыми современными панелями сверхвысокой чёткости (UHD), поддерживающими разрешение экрана: 7680x4320 пикселей.

Современные системы управления сайтом (CMS: WordPress, MODX), позволят предпринимателю (владельцу сайта) донести до своих посетителей всю необходимую информацию. С другой стороны (со стороны разработчика), вёрстка (дизайн) веб–сайтов – востребованная, творческая IT–профессия. Изучение веб–технологий (и, в первую очередь практика) – обязательный шаг, который при должном упорстве и частоте занятий приведёт человека к успеху: возможности получения прибыли. Возможно, это мнение ошибочно, но, кажется, что порог вхождения в вёрстку ниже, чем, например, в те IT–специальности, которые требует знания математики на высоком уровне, такие как Back–end (Full Stack: компетентность в Front и Back – end), разработка всевозможных приложений (для настольных и мобильных платформ). Всё это вызывает искренний интерес к данной специальности.

1. Основание для разработки
   1. Основанием для данной работы служит задание на курсовое проектирование.
   2. Наименование работы: веб–сайт экскурсионной компании «Аквамарин».
   3. Исполнители: студент группы 4–П–2 Шабанов И.Ф., руководитель Шепилова А.А.
2. Назначение работы
3. Оказание помощи в масштабировании бизнеса.
4. Освоение необходимых компетенций для старта в профессии верстальщика.

Наличие веб–сайта должно помочь организации автоматизировать некоторые бизнес–процессы, по ведению учета, ответу на часто задаваемые вопросы, о вариантах экскурсий и, (возможно, в будущем предоставлении чека), что позволит частично снизить нагрузку на сотрудников организации и при этом, что очень важно привлечь новых клиентов.

1. Технические требования
   1. Требования к функциональным характеристикам.
      1. Состав выполняемых функций:

Разрабатываемый веб–сайт должен обеспечивать:

* Хранение информации о вариантах экскурсий;
* Хранение информации о выборе заказа;
* Управление данными (добавление, удаление, редактирование);
* Быть адаптивным (отзывчивым к используемому разрешению экрана).
  + 1. Организация входных и выходных данных:

Исходная информация вносится в систему через поля ввода, затем, сохраненная в базе данных информация, отображается пользователям АРМ.

* 1. Условия эксплуатации и требования к составу и параметрам технических средств:

Информационная система имеет минимальные системные требования. Увеличение системных требований зависит от объема обрабатываемой информации.

* 1. Требования к транспортировке и хранению:

Программа поставляется на лазерном носителе информации. Программная документация поставляется в электронном виде.

* 1. Специальные требования:

1. Веб–сайт должен иметь дружественный и понятный интерфейс, рассчитанный на неопытного пользователя;
2. Пользователь системы должен обладать общими навыками работы на персональном компьютере;
3. Требования к программной документации

Состав программной документации, предъявляемой на испытании:

* описание программы (ГОСТ 19.402–78);
* программа и методика испытаний (ГОСТ 19.301–79);
* текст программы (ГОСТ 19.401–78).

Основными документами, регламентирующими разработку будущих программ, должны быть документы Единой Системы Программной документации (ЕСПД): руководство пользователя, руководство администратора, описание применения.

1. Технико–экономические показатели

Эффективность системы определяется удобством использования АРМ для контроля заказов меню ресторана.