

АННОТАЦИЯ

Тема выпускной квалификационной работы Разработка веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов Google.

Ключевые слова: веб-приложение, формирование заявки, технологии программирования, облачный сервис, фитнес-центр.

Объект исследования: веб-приложение формирования заявок фитнес центра.

Предмет исследования: технология разработки веб-приложения формирования заявок фитнес центра.

В ходе работы были выполнены задачи: рассмотрение вопросов о технологиях разработки кроссбраузерных веб-приложений; проводился анализ технологий разработки; проектировалось, разрабатывалось веб-приложение на базе PHP.

В теоретической части рассматривается вопрос о востребованности разработки веб-приложений для обслуживания клиентов фитнес-центров. Также рассматривается вопрос о технических особенностях построения приложений. Были выведены достоинства, недостатки готовых решений в данной области и на основе этого принято решение реализовать новое веб-приложение формирования заявок фитнес-центра. Практическая часть посвящена концептуальному и логическому моделированию веб-приложения формирования заявок фитнес-центра. После чего был осуществлен выбор необходимых технологий для реализации приложения.

По итогу проделанной работы реализовано веб-приложение на базе языка программирования PHP и языков разметки HTML и CSS, в котором был автоматизирован процесс формирования заявки для посещения фитнес-центра.

Данная работа состоит из 45 страниц с приложением, включая введение, три главы, заключение, список используемой литературы из 25 источников.

ABSTRACT

Subject of the final qualification work is Development of a web application of formation of applications of the fitness center with use of cloud services of Google.

Keywords: web application, formation of the application, technology of programming, cloud service, fitness center.

Research object: web application of formation of applications fitness of the center.

Object of research: technology of development of a web application of formation of applications fitness of the center.

During work tasks have been carried out: consideration of questions of technologies of development of cross-browser web applications; the analysis of technologies of development was carried out; the web application on the basis of PHP was projected, developed.

In a theoretical part the question of demand of development of web applications for customer service of the fitness centers is considered. Also, the question of technical features of application programming is considered. Advantages have been removed, shortcomings of ready decisions in the field and on the basis of it the decision to realize a new web application of formation of applications of the fitness center is made. A practical part is devoted to conceptual and logical modeling of a web application of formation of applications of the fitness center. Then the choice of necessary technologies for implementation of the application has been carried out.

On a result of the done work the web application on the basis of the PHP programming language and markup languages of HTML and CSS in which process of formation of the application for visiting of the fitness center has been automated is realized.

This work consists of 45 pages, including introduction, three chapters, the conclusion, the list of the used literature from 25 sources.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	6
ГЛАВА 1 АНАЛИЗ СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ ДЛЯ РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЙ.....	8
1.1 Анализ востребованности веб-приложений для фитнес-центров.....	8
1.2 Формирование требований к будущему веб-приложению.....	9
1.3 Обзор и анализ существующих веб-приложений фитнес-центров	12
1.4 Достоинства веб-приложений для фитнес-центров.....	15
1.5 Недостатки веб-приложений для фитнес-центров.....	16
ГЛАВА 2 ПРОЕКТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ ФОРМИРОВАНИЯ ЗАЯВОК ФИТНЕС-ЦЕНТРА С ИСПОЛЬЗОВАНИЕМ ОБЛАЧНЫХ СЕРВИСОВ GOOGLE.....	19
2.1 Разработка алгоритма работы веб-приложения.....	19
2.2 Моделирование веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов.....	19
2.2.1 Архитектура веб-приложения.....	20
2.2.2 Функциональное моделирование веб-приложения.....	21
2.2.3 Логическое моделирование веб-приложения.....	22
ГЛАВА 3 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ ФОРМИРОВАНИЯ ЗАЯВКИ ФИТНЕС-ЦЕНТРА С ИСПОЛЬЗОВАНИЕМ ОБЛАЧНЫХ СЕРВИСОВ GOOGLE.....	27
3.1 Выбор средств реализации.....	27
3.2 Выбор языка разметки.....	29
3.3 Выбор системы управления базой данных.....	30
3.4 Облачные сервисы Google.....	30
3.5 Реализация интерфейса и описание работы веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов Google.....	34
ЗАКЛЮЧЕНИЕ.....	38
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	39

ПРИЛОЖЕНИЕ А.....	42
ПРИЛОЖЕНИЕ Б.....	43

ВВЕДЕНИЕ

Использование сети Интернет в современном мире стало уже абсолютно привычным, и с каждым днем повышается актуальность веб-приложений, которые в свою очередь предоставляют пользователям и клиентам доступ через веб-браузер. Веб-приложения выполняют основную часть бизнес-логики на стороне одного и более серверов. Актуальность подобных приложений обусловлена тем, что они являются кроссплатформенными и кроссбраузерными. Таким образом, нет необходимости устанавливать и настраивать специальные программы, приложения не зависят как от операционной системы персонального компьютера, так и от браузеров.

Работа основной части программного обеспечения производится на сервере, благодаря чему повышается надежность систем, и сокращаются расходы ресурсов и времени, предназначенные для получения ответа на запрос в браузере. Множество серверов заранее определено и пользователям предоставляются уже готовые результаты в неискаженном виде, то есть, корректное отображение в различных платформах и браузерах.

Главное преимущество данной архитектуры заключается в единовременном доступе для множества пользователей к информационному хранилищу.

Спрос на услуги фитнес-центров растет с каждым днем, а вместе с этим увеличивается и количество спортивных центров. Каждый, кто приходит в фитнес центр, должен заполнять необходимые формы и анкеты для получения членства и клубной карты. И естественно, что в век высоких технологий человек максимально стремится упростить весь процесс оформления тех или иных документов посредством интернет-сервисов и различных приложений.

Объект исследования: веб-приложение формирования заявок фитнес центра.

Предмет исследования: технология разработки веб-приложения формирования заявок фитнес центра.

Цель работы: разработать веб-приложение для фитнес-центра с использованием облачных сервисов.

Для достижения целей данной выпускной квалификационной работы необходимо решить следующие задачи:

1. Исследовать предметную область.
2. Проанализировать существующие технологии разработки веб-приложений.
3. Рассмотреть основные этапы разработки веб-приложений.
4. Изучить литературу по разработке веб-приложений.
5. Выбрать подходящие технологии, методы и средства для разработки веб-приложения.
6. Реализовать веб-приложение.

В первой главе проведен анализ информации: описание предметной области, обзор существующих технологий разработки веб-приложения.

Во второй главе процесс разработки веб-приложения разбивается на этапы и выбираются необходимые средства и технологии для ее разработки.

В третьей главе описывается работа приложения.

В заключении формулируются выводы, полученные в процессе выполнения данной выпускной квалификационной работы.

ГЛАВА 1 АНАЛИЗ СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ ДЛЯ РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЙ

1.1 Анализ востребованности веб-приложений для фитнес-центров

В 21 веке посещение фитнес центров стало набирать популярность с геометрической прогрессией. Для большинства людей, живущих в больших городах и ведущих малоподвижный образ жизни, занятия в спортзалах и фитнес центрах, является единственным способом повышать и поддерживать степень своей физической подготовки, а также улучшать состояние здоровья. Согласно проведенным социальным исследованиям ВЦИОМа (Всероссийский центр изучения общественного мнения) уровень заинтересованности в спорте у россиян повысился с 2010 года до настоящего времени на 47%. Это свидетельствует о том, что спрос на услуги растет с каждым днем, а вместе с этим увеличивается и количество спортивных центров. Естественно, что каждый, кто приходит в фитнес центр, должен заполнять необходимые формы и анкеты для получения членства и клубной карты. И если принять во внимание исследование ВЦИОМа, то поток посетителей будет увеличиваться, а заниматься бумажной волокитой и постоянно следить за тем, кто и когда оставляет заявки на посещение центра, очень трудоемко для работников. Конечно, при таком способе обработки данных неизбежны ошибки, и в век высоких технологий человек максимально стремится упростить весь процесс оформления тех или иных документов посредством интернет-сервисов и различных приложений.

По своей сути веб-приложение — это приложение, состоящее из клиентской и серверной части, которое позволяет пользователю взаимодействовать с сервером с помощью браузера. Другими словами — это посредник между клиентом и сервером. Обмен информацией осуществляется по сети, данные хранятся в большей части на сервере.

Клиентская часть отвечает за реализацию интерфейса пользователя и формирование запросов к серверу и обработка ответов на эти самые запросы.

Серверная часть принимает запросы от клиентов, выполняет вычисления после этого формируются веб-страницы, которые отправляются клиентам, использующие протокол HTTP.

На стороне клиента используются: HTML, XHTML, CSS, ActiveX, Adobe Flash, Adobe Flex, Java, JavaScript, Silverlight. Рассмотрим подробнее технологии, используемые на стороне сервера и технологии, применяющиеся на стороне клиента.

На стороне сервера для создания приложения используют различные технологии и языки программирования, осуществляющие вывод на консоль. Наиболее часто используемые технологии веб-программирования, используемые на стороне сервера: ASP, ASP.net, Java, Perl, PHP, Python.

Для разработки любого веб-приложения в первую очередь необходимо определить алгоритм его работы и выдвинуть функциональные и нефункциональные требования, которым оно должно соответствовать.

1.2 Формирование требований к будущему веб-приложению

Существует классификация FURPS, которая была разработана и предложена в 1992 году Грейди Робертом. Не смотря на год в котором была предложена данная модель, она является актуальной и в наши дни. В ней были описаны требования к программным системам. Данные требования предполагают, что именно и как должно быть реализовано в системе.

Согласно данной классификации составим требования к веб-приложению. Каждый символ FURPS+ характеризует определенное требование:

- F- Functionality (функциональность);
- U- Usability (удобство использования);
- R- Reliability (надежность);
- P- Performance (производительность);
- S- Supportability (поддерживаемость);
- + - ограничения.

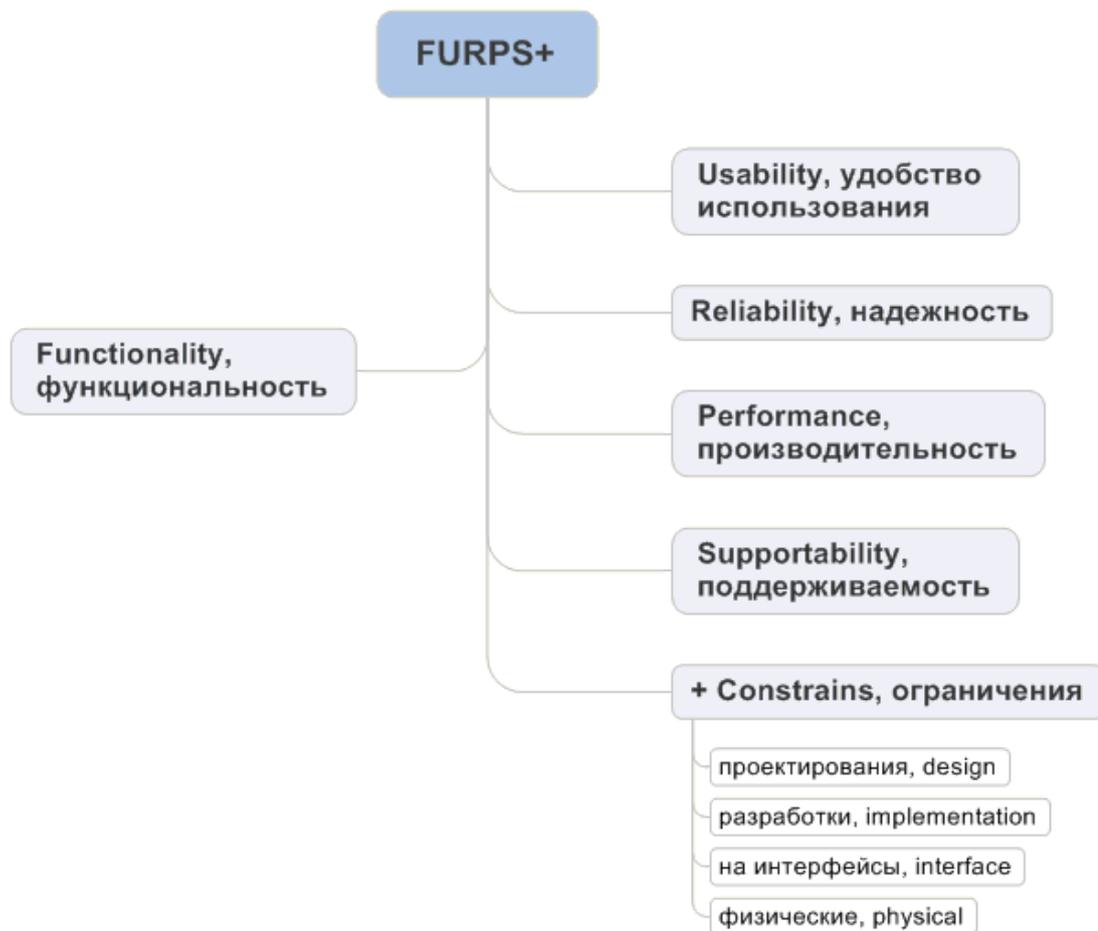


Рисунок 1.1 – Классификация требований FURPS+

Рассмотрим каждый из данных пунктов более подробно.

Функциональные требования описывают свойства и функции системы, которые она выполняет. Например: лицензирование, почта, помощь, печать, отчетность, безопасность.

Удобство использования должно отвечать за следующие виды требований:

- логика пользовательского интерфейса;
- защищенность от человеческого фактора;
- стандарты эксплуатационных документаций;
- квалифицированность пользователей и повышение их квалификации;
- системная справочная информация.

Требования к надежности содержат такие характеристики, как:

- отказоустойчивость;

- предсказуемое поведение;
- режим работы;
- точные вычисления.

К требованиям производительности относят следующие характеристики:

- производительность системы;
- эффективность;
- пропускная способность;
- потребление ресурсов.

К требованиям сопровождения относят:

- масштабирование;
- конфигурирование;
- совместимость;
- доработка;
- локализация;
- интернационализация.

Существует ряд ограничений, накладываемых на разрабатываемые технические и программные продукты.

Ограничения делятся на следующие основные группы:

- ограничения проектирования;
- ограничения реализации написания кода;
- требования к интерфейсу;
- физические ограничения.

Разрабатываемое веб-приложение должно быть простым, понятным и удобным в пользовании. Также оно должно быть кроссбраузерным, независимым от браузеров и соответственно от устройств с доступом в интернет. Код веб-ресурса должен быть простым и понятным для дальнейшей поддержки и внесения возможных изменений. Приложение должно отражать информацию о фитнес-центре и предоставлять пользователям данные об

оказываемых услугах. Данный продукт должен быть ярким и ненавязчивым. Текстовой информации на странице «Главная» должно быть мало, большую часть данной страницы должны занимать фотографии, информативные картинки, слайд шоу. Должен быть раздел «Блог», где будут размещаться полезные статьи и советы от персонала центра. Необходимо реализовать максимально полную форму для составления заявок на посещение с использованием сервиса Google Forms. К странице «Контакты» нужно привязать Google Map и контактные данные администратора фитнес-центра.

После того как были выдвинуты требования к будущему веб-приложению, необходимо рассмотреть готовые решения, выявить их достоинства и недостатки и учесть полученные результаты сравнения при реализации.

1.3 Обзор и анализ существующих веб-приложений фитнес-центров

В ходе работы необходимо реализовать качественное веб-приложение фитнес-центра. Для этого следует рассмотреть уже готовые решения и выявить их достоинства и недостатки. Затем основываясь на результатах сравнения ранее реализованных приложений, разработаем новое.

Для анализа были выбраны веб-приложения наиболее популярных фитнес-центров и клубов в городе Тольятти.

Основные характеристики, по которым оценивались уже готовые веб-приложения:

- простота пользования;
- доступность информации;
- перечень инструкторов и данные о них;
- удобный интерфейс пользователя;
- интерактивность страниц;
- раздел «Блог», где размещена полезная информация о правильном образе жизни;
- отсутствие рекламы на страницах;

- размещенная контактная информация;
- реализация формы для заполнения клиентами;
- полнота формы для заполнения клиентами;
- использование облачных сервисов Google.

На рисунках 1.2, 1.3 и 1.4 представлены главные страницы веб-приложений «Alex Fitness», «Fit line», и «Богиня» соответственно.

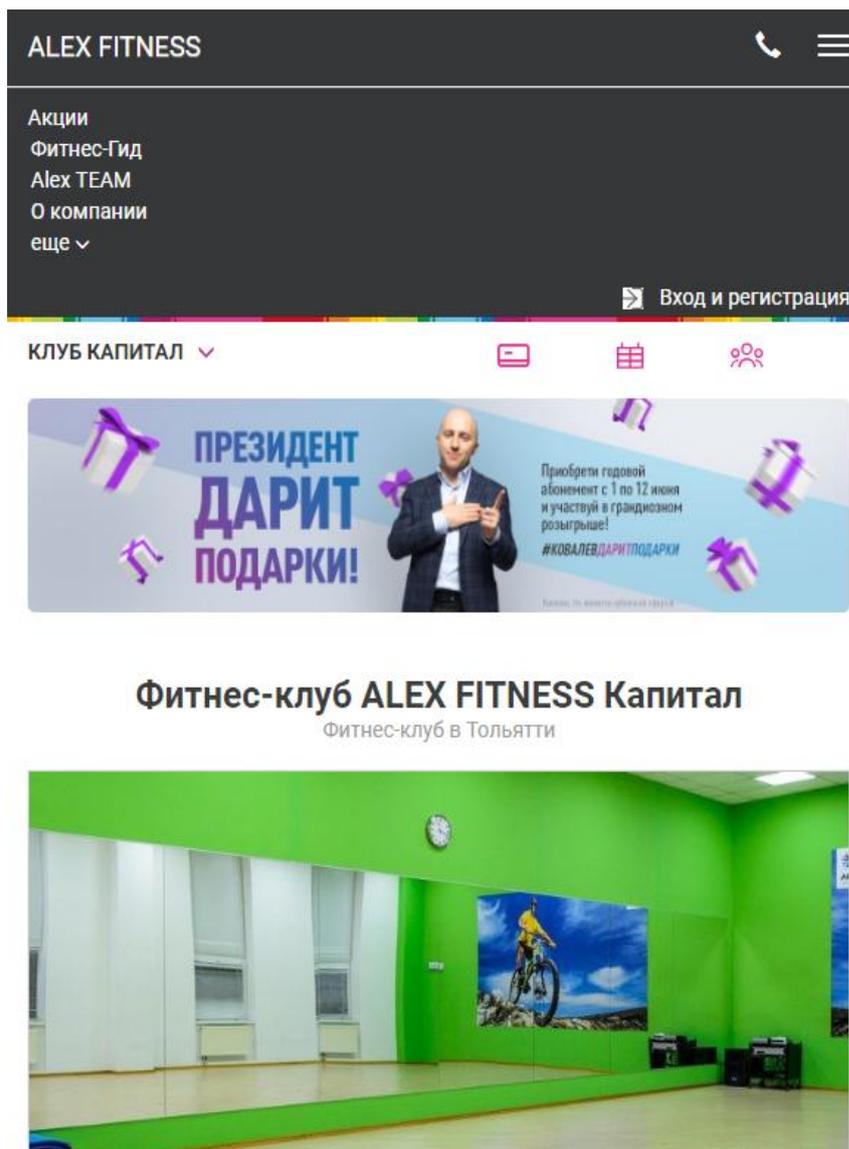


Рисунок 1.2 — Главная страница приложения фитнес-клуба «Alex Fitness»

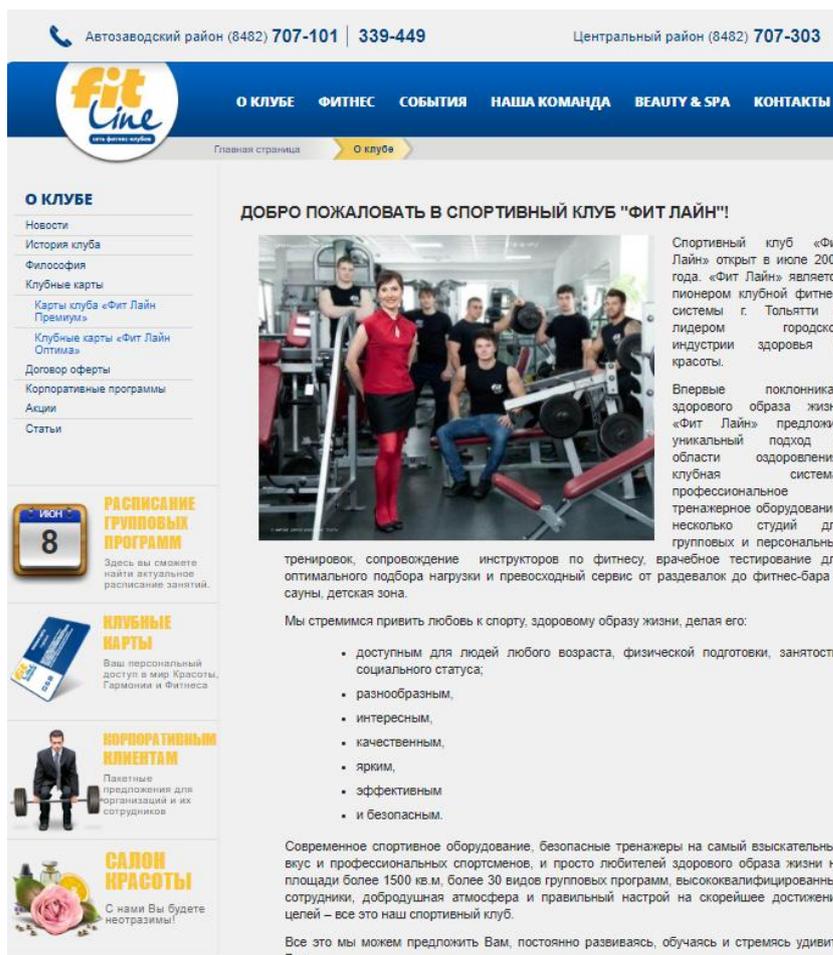


Рисунок 1.3 — Главная страница приложения фитнес-клуба «Fit line»

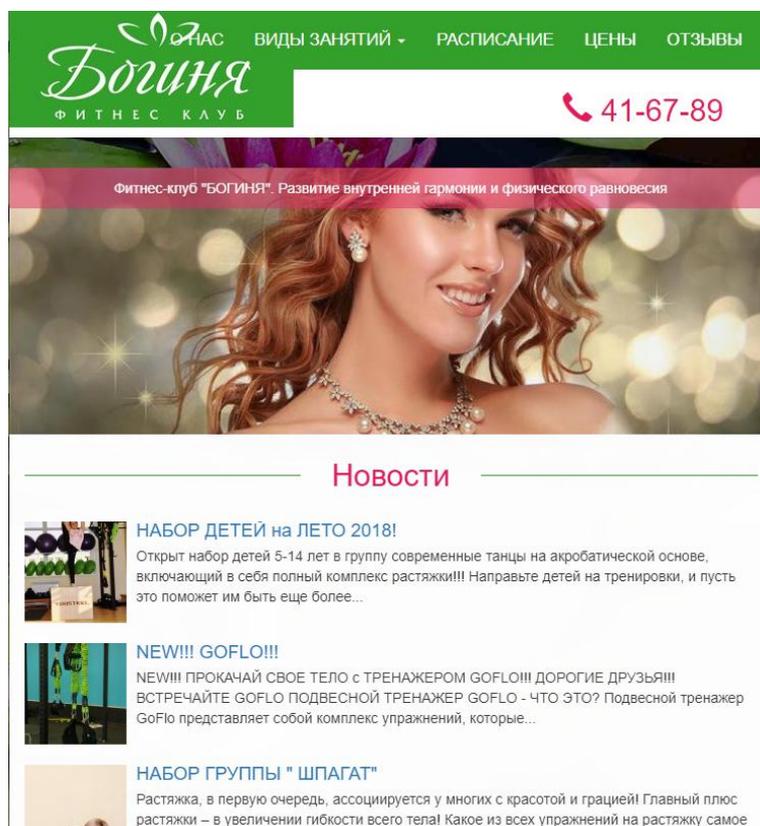


Рисунок 1.4 — Главная страница приложения фитнес клуба «Богиня»

Все три приложения выполняют свою основную функцию, есть информация о действующих услугах, размещены фотографии, демонстрирующие состояние центров, приведен список тренеров и инструкторов, работающих с клиентами.

В приложении «Alex Fitness» есть раздел, посвященный ценам на существующие в клубе услуги, есть возможность заказа абонемента, но нет возможности сразу же выбрать тренера и не предусмотрен выбор желаемого времени первого посещения зала.

В «Fit line» нет возможности заказать абонемент посредством заполнения форм или анкет и данный веб-ресурс довольно статичный. В приложении фитнес-клуба «Богиня» так же, как и в «Fit line» нет возможности заказать абонемент онлайн.

В рассмотренных приложениях либо нет формы для заполнения клиентами, либо она является неполной, хотя это является действительно важным аспектом подобного веб-приложения. Данные готовые решения были реализованы без использования облачных сервисов Google. Для того чтобы клиенты были довольны предоставляемыми услугами, необходимо удовлетворить их потребности и дать возможность самим определять условия посещения фитнес-центра.

На основании полученных данных составим перечень достоинств и недостатков существующих приложений и выявим то, над чем необходимо будет провести работу в новом приложении.

1.4 Достоинства веб-приложений для фитнес-центров

Основное достоинство веб-приложений заключается в том, что можно ознакомиться с концепцией фитнес-центра удаленно без личного присутствия. Это особо актуально при выборе фитнес-центра для людей, с плотным графиком или с полным рабочим днем.

Онлайн-веб-приложение также обладает следующими преимуществами в сравнении с привычными действиями осуществляемыми клиентами фитнес-центра:

- доступ к приложению в любое удобное для клиента время;
- доступ с любого устройства, на котором подключен интернет;
- есть возможность ознакомиться с существующими услугами центра, графиком работы и его местонахождением;
- можно оставлять сообщения персоналу клуба;
- ознакомление с персональными тренерами и получение сведений о их квалификации.

Люди, которые целый день заняты и не могут в середине дня попасть в фитнес-центр, но имеют желание заниматься спортом и посещать оздоровительные занятия могут уладить все формальности посредством веб-приложения и запланировать занятия на удобное для них время.

1.5 Недостатки веб-приложений для фитнес-центров

Несмотря на то, что веб-приложения, ориентированные на обслуживание клиентов спортивных центров, имеют ряд достоинств, они обладают и недостатками, среди которых:

- невозможность составления заявки на посещение фитнес центра, а также на получение абонемента или клубной карты;
- невозможность дистанционного определения состояния здоровья и подходящих программ для клиента;
- у клиента не будет возможности посетить веб-приложение, если на его устройстве не подключены услуги интернет;
- ошибочное указание собственных данных пользователями;
- не использованы облачные сервисы Google.

Если сопоставить достоинства и недостатки веб-приложений для фитнес-центров, то можно убедиться в том, что к недостаткам в большинстве относятся ситуации, в которых решающим является человеческий фактор. Но

данные недостатки можно минимизировать если разумно подойти к реализации нового веб-приложения. Для наглядности была составлена сравнительная таблица характеристик ранее рассмотренных приложений фитнес-клубов (Таблица 1. Сравнительная характеристика готовых веб-приложений).

Таблица 1. Сравнительная характеристика готовых веб-приложений

Характеристики	«Alex Fitness»	«Fit line»	«Богиня»
Простота пользования	-	+	+
Доступность информации	+	+	+
Перечень инструкторов и данные о них	+	+	+
Удобный интерфейс	-	+	+
Интерактивность страниц	+	-	-
Отсутствие рекламы на страницах	-	+	+
Присутствие контактной информации для связи	+	+	+
Реализация формы для заполнения клиентами	+	-	-
Полнота формы опроса	-	-	-
Использование облачных сервисов Google	-	-	-
Раздел «Блог», с полезной информацией	+	-	-
Итого	6	6	6

Данная таблица позволяет понять, что рассмотренные приложения не соответствуют всем требованиям. У каждого из них есть свои достоинства и преимущества, но имеется и ряд существенных недостатков. Другими словами, автоматизация процесса посещения фитнес-центра и формирования заявок посетителями является оправданной и целесообразной. Необходимо учесть ранее выявленные результаты сравнения и реализовать веб-приложение формирования заявок фитнес-центра с использованием

облачных сервисов Google, которое будет соответствовать выдвинутым требованиям.

Выводы по первой главе

В первой главе выпускной квалификационной работы были изучены: достоинства и недостатки готовых веб-приложений фитнес-центров.

Проанализирован вопрос востребованности приложений для спортивных и оздоровительных центров. Изучены основные технические особенности построения подобных приложений.

Анализ показал, что существует большое количество веб-ресурсов, которые направлены на поддержание дистанционной работы с клиентами, а конкретно с потенциальными посетителями фитнес-центров, но нет IT-решений, позволяющих самим клиентам определять интенсивность занятий, выбирать тренеров и услуги, которые они считают действительно необходимыми.

Существующие веб-ресурсы в большей мере направлены на информирование людей, а не на конструктивный сбор данных о клиентах, для дальнейшей эффективности работы.

Поэтому было принято решение о разработке нового веб-приложения формирования заявки фитнес-центра с использованием облачных сервисов Google.

ГЛАВА 2 ПРОЕКТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ ФОРМИРОВАНИЯ ЗАЯВОК ФИТНЕС-ЦЕНТРА С ИСПОЛЬЗОВАНИЕМ ОБЛАЧНЫХ СЕРВИСОВ GOOGLE

2.1 Разработка алгоритма работы веб-приложения

Логика данного веб-приложения должна быть простой и понятной как для опытных пользователей сети Интернет, так и для «новичков».

На рисунке 2.1 представлена блок-схема алгоритма работы веб-приложения.

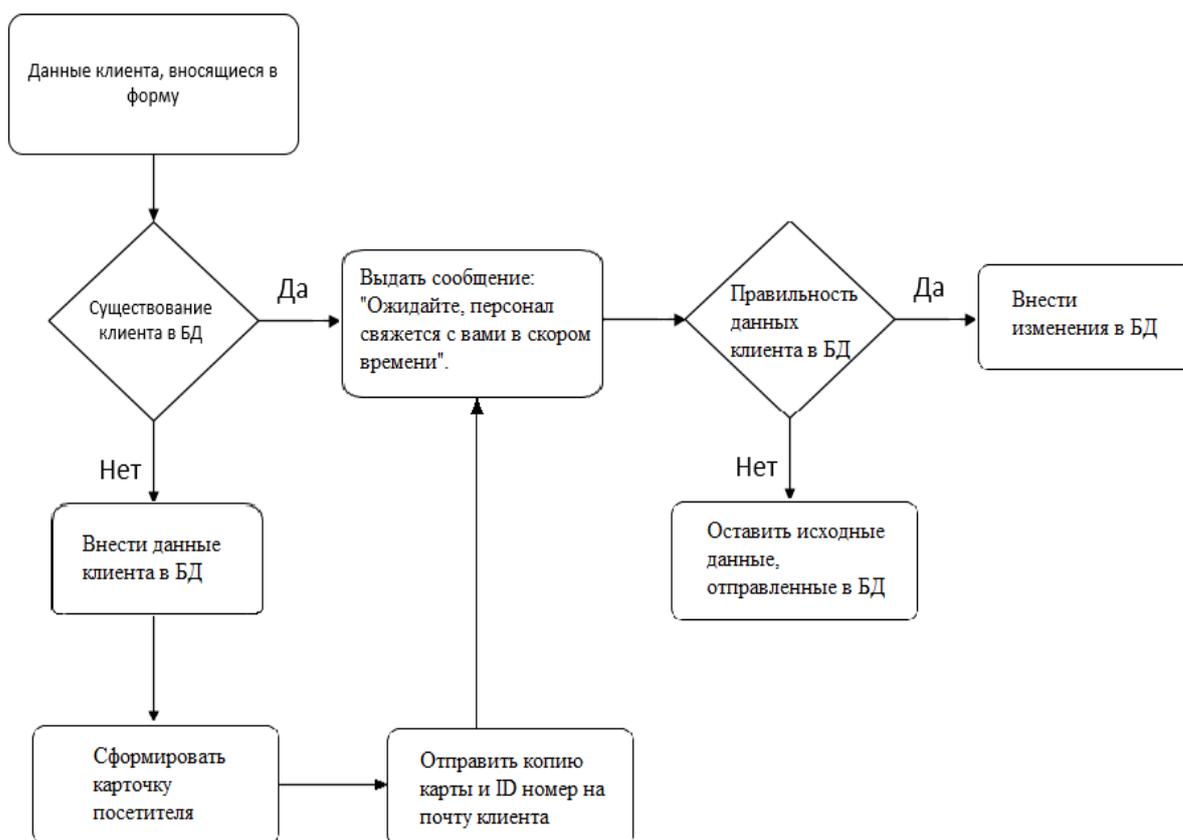


Рисунок 2.1 – Блок-схема алгоритма работы приложения

Пользователь знакомится с информацией, в случае если клиента все устраивает, он заполняет предоставленную форму и отправляет запрос.

После этого данные клиента заносятся в базу данных (БД). В случае, если клиент уже числится в БД, выдается окно с сообщением «Ожидайте, персонал центра свяжется с вами в скором времени».

Администратор связывается с клиентом и, если какие-либо данные изменились, необходимо внести эти изменения в БД.

2.2 Моделирование веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов

2.2.1 Архитектура веб-приложения

При разработке веб-приложения одним из главных компонентов является выбор архитектуры. Веб-приложения в последнее время разрабатываются согласно архитектуре «клиент-сервер». Условно существуют поставщики функций (серверы) и их потребители (клиенты). Основная идея заключается в разбиении веб-приложения на компоненты, реализующие определенные наборы сервисов. При стандартной реализации веб-приложения используется трехзвенная архитектура (рисунок 2.2).

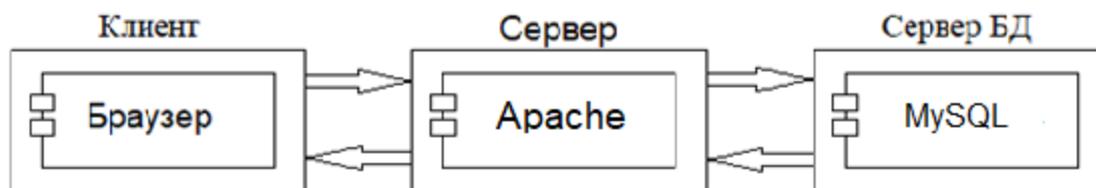


Рисунок 2.2 – Стандартная архитектура веб-приложения

Клиентом является браузер, а сервером - веб-сервер.

Логика веб-приложения распределена таким образом, что происходит постоянное взаимодействие между сервером и клиентом, данные хранятся преимущественно на сервере, все запросы и ответы на данные запросы отправляются по сети.

Данный подход имеет существенное преимущество: клиенты приложения не зависят ни от браузера, ни от платформы своего устройства, если есть доступ к сети интернет- то есть доступ к приложению.

Но так как есть требование к разработке использовать облачные сервисы Google, архитектура веб-приложения немного изменится и приобретет вид, представленный на рисунке 2.3.

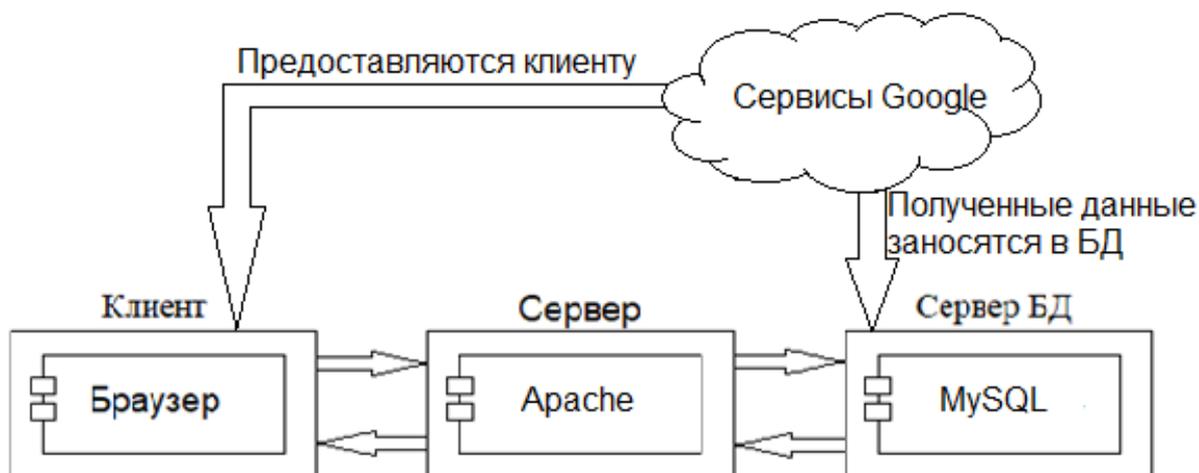


Рисунок 2.3. – Архитектура веб-приложения фитнес-центра

В данной архитектуре важную роль играют облачные сервисы Google. В веб-приложении будет представлена Google Form, она будет напрямую доступна каждому, кто посещает страницу «Оставить заявку», и после заполнения клиентом формы, полученные данные будут заноситься в таблицу, которая располагается на Google Drive администратора центра. Затем все данные проверяются еще раз во избежание внесения некорректных данных и заносятся в базу данных.

2.2.2 Функциональное моделирование веб-приложения

Проектируемое веб-приложение формирования заявок фитнес-центра, должно отвечать определенным функциональным требованиям. На рисунке 2.4 показана функциональная модель приложения, где отображены его основные функции.

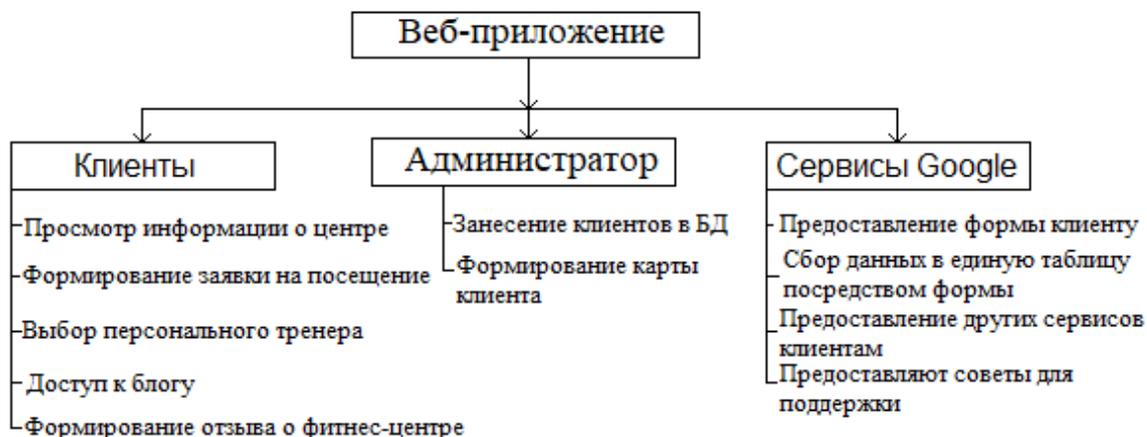


Рисунок 2.4 – Функциональная модель веб-приложения

Основные функции веб-приложения:

- информирование клиентов об услугах;
- предоставление облачными сервисами возможности на формирование заявки на посещение;
- выбор личного тренера;
- получение информации, касающиеся образа жизни клиентов;
- непосредственно формирование заявки на посещение;
- сбор данных в единой таблице;
- предоставление актуального, для клиентов контента.

При открытии приложения в браузере, на экран выводится главное меню, адрес центра, номер телефона по которому можно связаться с персоналом фитнес-центра и электронная почта. Через панель меню пользователь может выбрать интересующую его вкладку, ознакомиться с предоставляемыми услугами, получить информацию о тренерах, а также есть возможность получить знания о здоровом питании, и полезных привычках знаменитых спортсменов. На случай, если у пользователей веб-ресурса возникают вопросы, они могут написать на почту персонала, либо связаться с ними посредством социальных сетей, все данные для этого предоставлены.

Клиент самостоятельно может ознакомиться с существующими программами и в случае, если его все устраивает, он может оставить заявку на посещение. В день, когда назначено посещение центра администратором, должна быть составлена карта посетителя и выдана на руки клиенты, после подписания договора на оказание услуг. Клиент обязан при посещении центра расписаться за технику безопасности, с данной информацией он может ознакомиться также на веб-ресурсе.

2.2.3 Логическое моделирование веб-приложения

Логическое моделирование представляет собой осуществление проверки функционирования логической схемы.

Цель, преследуемая данным видом моделирования, заключается в осуществлении проведения проверки функции проектируемой логической

схемы без полной реализации на данном этапе разработки. Преимущество данной модели заключается в том, что осуществляется проверка, как логических функций приложения, так и её временных соотношений.

Диаграмма вариантов использования описывает взаимоотношения между вариантами использования и действующими лицами. Варианты использования – это описание взаимодействий между пользователем и системой.

На рисунке 2.5 представлена диаграмма вариантов использования веб-приложения.

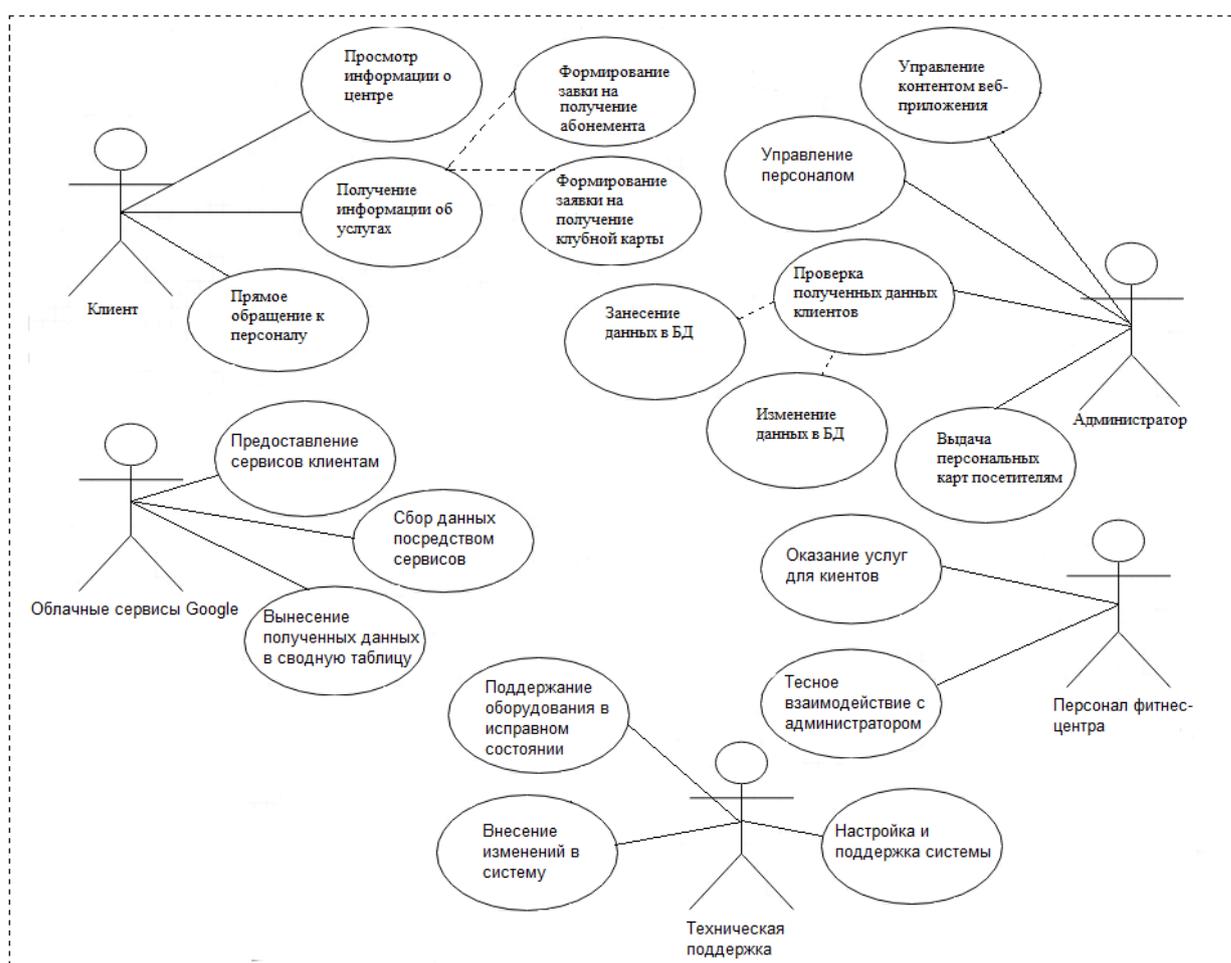


Рисунок 2.5 – Диаграмма вариантов использования

Действующее лицо - пользователь, который является источником взаимодействия с системой с помощью варианта использования.

Действующими лицами в системе являются:

- клиенты (посетители);

- администратор;
- персонал фитнес-центра;
- техническая поддержка;
- облачные сервисы Google.

Диаграмма последовательности – это диаграмма, на которой для определенного набора объектов на одной временной оси демонстрируется жизненный цикл объекта и взаимодействие актеров.

Данная диаграмма необходима для того, чтобы лучше понимать алгоритм действий, производимые различными действующими лицами.

Взаимодействие между объектами и субъектами происходит следующим образом:

1. Серверами Google предоставляются облачные сервисы, которые внедряются в страницы веб-приложения.

2. С помощью таблиц на Google Drive администратор может просматривать информацию о клиентах центра, а также о людях, которые оставили заявку на посещение.

3. Клиент открывает с помощью браузера приложение и тем самым получает доступ к информации о фитнес-центре, может ознакомиться с услугами, данными тренеров. У него есть возможность сформировать заявку и отправить ее.

4. После заполнения формы данные из заявки автоматически вносятся в таблицу, на которую ссылается форма.

5. Внесенные данные проверяет администратор, связываясь с клиентом.

6. После проверки данные либо изменяются самим администратором, либо остаются неизменными.

7. Затем, при условии, что формальности соблюдены, формируется карточка клиента и выдается ему при посещении центра.

На рисунке 2.6 представлена диаграмма последовательности работы веб-приложения.

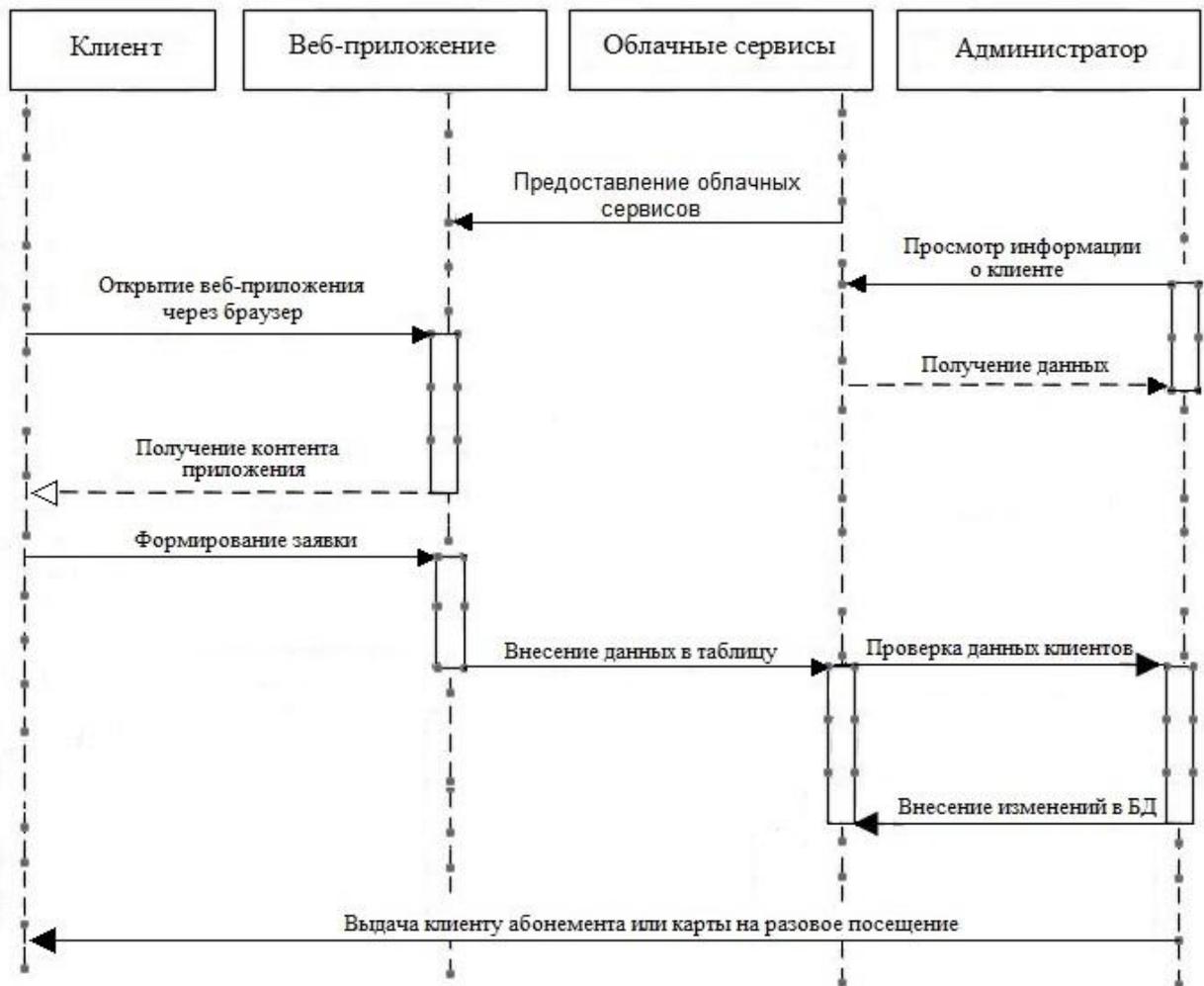


Рисунок 2.6 – Диаграмма последовательности работы веб-приложения

Таким образом, была осуществлена проверка логики функционирования веб-приложения. Описана взаимосвязь и последовательность работы в приложении.

Для более точного определения программных составляющих была составлена диаграмма компонентов.

На рисунке 2.7 продемонстрирована взаимосвязь серверной части с модулями приложения и облачными сервисами. База данных фитнес-центра будет размещена на сервере MySQL. Доступ к БД будет организована с помощью HTTP-сервера Apache. Все разделы приложения будут писаться на языке PHP. Так как к разделам «Контакты» и «Заполните форму» будут прикреплены Google сервисы, необходимо связать разделы приложения с сервисами, которые предоставляются Google-серверами mfe и GFE/1.3.

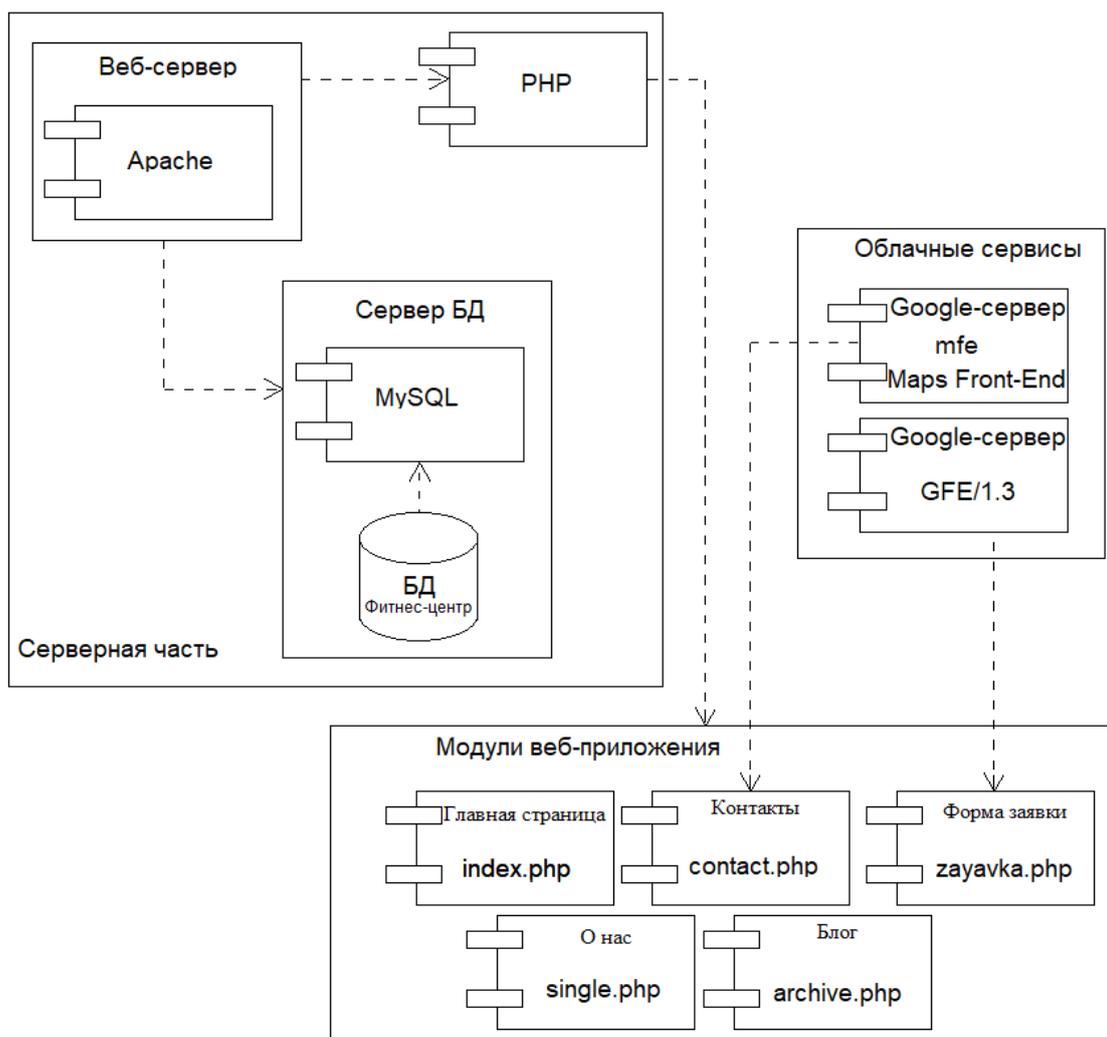


Рисунок 2.7 – Диаграмма компонентов веб-приложения

Выводы по второй главе

В данной главе было произведено проектирование приложения, включающее логическое и функциональное моделирование.

Разработан алгоритм работы веб-приложения. Реализованная логика алгоритма приложения позволит вносить в БД клиентов максимально корректные данные. Смоделировано веб-приложение формирования заявок фитнес-центра. Визуально была представлена архитектура будущего приложения с использованием Google сервисов.

Построена функциональная и логическая модели веб-приложения.

ГЛАВА 3 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ ФОРМИРОВАНИЯ ЗАЯВКИ ФИТНЕС-ЦЕНТРА С ИСПОЛЬЗОВАНИЕМ ОБЛАЧНЫХ СЕРВИСОВ GOOGLE

3.1 Выбор средств реализации

Для того, чтобы сделать правильный выбор технологии для разработки мало иметь лишь субъективные причины, нужно основываться прежде всего на фактах. Прежде необходимо выявить основные критерии выбора технологии:

- размер и тип проекта;
- сложность проекта;
- скорость разработки;
- стоимость технологии;
- доступность технологии;
- актуальность технологии на данное время;
- кроссбраузерность.

Разрабатываемое приложение будет небольшим и ориентированным в большей степени на формирование заявок клиентами, то есть приложение должно быть простым, понятным и удобным для пользователей. В приложении должно быть меню для перехода между разделами приложения. Внешний вид не должен утомлять клиентов, поэтому предоставляемая информация будет максимально содержательной и представленной в интерактивной форме. В данном случае лучше отбросить дальнейшее рассмотрение ASP и ASP.net, так как они ориентированы на большие и серьезные проекты, и они уже успели утратить свою актуальность.

Perl так же не подходит для разработки. Во-первых, данный язык больше ориентирован на практичность и информативность, чем на красоту и интерактивность. Во-вторых, изучение данного языка более трудоемкое по сравнению с Python и PHP.

Несмотря на свою стабильность, читаемость и простоту язык программирования Python не подойдет для разработки, он имеет значимый недостаток- низкое быстродействие и скорее больше подходит для роли основного языка программирования серверной части.

Для создания веб-приложения язык PHP подходит больше всего.

Следует отметить, что PHP остается лидером среди веб-ориентированных языков.

Убедимся в этом на примере диаграммы (рисунок 3.1).

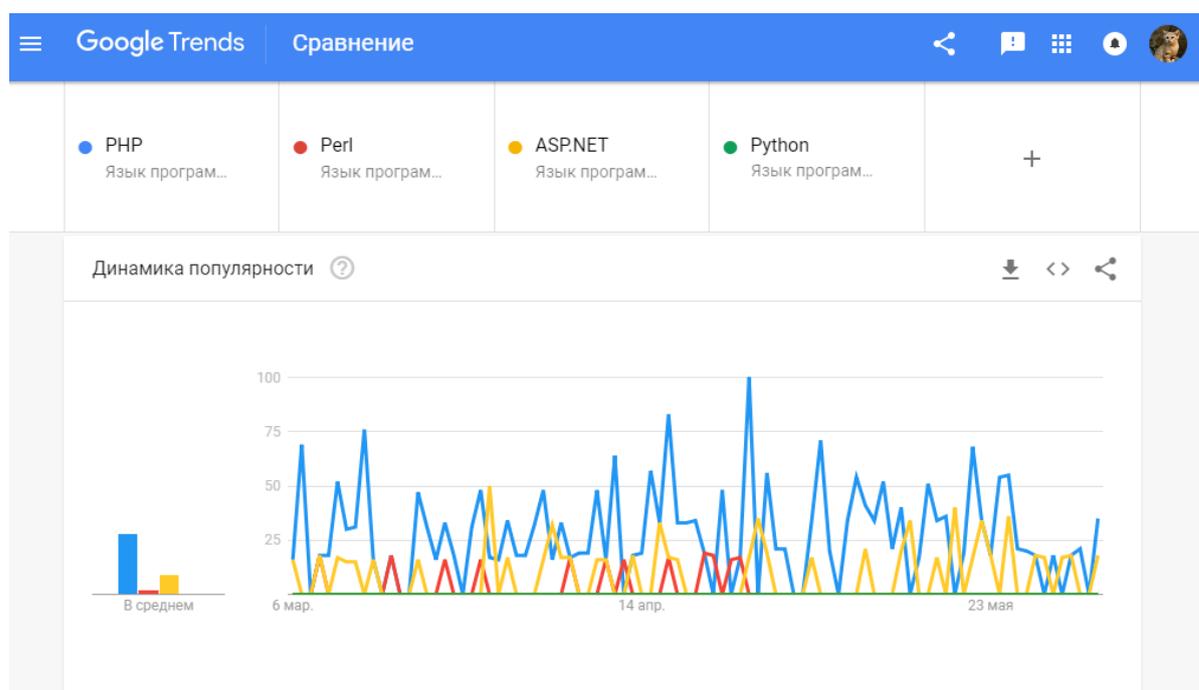


Рисунок 3.1 – Диаграмма динамики популярности языков программирования

С помощью сервиса Google Trends, было установлено, что язык PHP является действительно актуальным в настоящее время. На данном языке написано более 200 миллионов сайтов и веб-приложений. Этот язык не сложный в изучении, удобен в пользовании и у него достаточная функциональность для написания небольшого веб-приложения. На языке PHP были основаны такие сайты и веб-приложения как: Facebook, ВКонтакте, WordPress, Yahoo, Avito, Mail.ru. PHP- лучший язык для реализации веб-приложения. Этот язык является доступным в плане стоимости и работает практически на всех операционных системах. При

возникновении трудностей в написании кода есть большое сообщество разработчиков, которые могут помочь, и литературы по основам и техническим особенностям PHP достаточно. Данная технология предоставляет простые решения для реализации порой сложных задач. Для того чтобы задать «тело» программы, необходимо выбрать технологии, используемые на стороне клиента (языки разметки).

3.2 Выбор языка разметки

HTML (Hyper Text Markup Language)- язык гипертекстовой разметки. Но если быть точнее HTML — это скорее текст со вставками (теги, операторы, маркеры), которые имеют свойства (размер шрифта, его тип, позиция, цвет). В текстовом редакторе пишется нужный текст и с помощью специальных тегов назначаются свойства, затем согласно этим тегам в браузере отображается текст с изменениями. Кроме тегов, операторов можно в текст вставлять ссылки на документы, скрипты. Редактировать файлы, написанные на HTML, можно в стандартных редакторах (notepad, MS Word, Front Page Express).

CSS (Cascading Style Sheets)- каскадная таблица стилей. Призвана управлять внутренними и внешними видами таблиц. CSS задачи схожи с кодом HTML, но имеют больше возможностей, а также способны уменьшать объем страниц HTML. Цель преследуемая CSS заключается в раздельном описании логики и описании внешнего вида веб-страниц. Данное разделение способствует увеличению доступности документов, предоставляет гибкость и возможность управлять представлением, упрощает понимание кода и в структуре документов сокращается число повторений. В технологии CSS правила описываются на формальном языке и находятся они в таблицах стилей. Таблицы стилей могут находиться и в файлах формата CSS, и в веб-документа, где происходит описание его внешнего вида. Для того чтобы реализовать данную работу необходимо выбрать систему управления данными.

3.3 Выбор системы управления базой данных

Для внесения данных о новых клиентах и изменения существующей информации о посетителях фитнес-центра необходима система управления базой данных. Наиболее подходящей является MySQL по ряду причин. Данная система помогает минимизировать вероятность повторения данных. Все данные хранятся в таблицах. Для того, чтобы вносить изменения внутри базы, необходимо знать язык SQL. Благодаря специальным командам, данные можно внести, удалить, обновить или объединить.

Данная СУБД совместима со всеми популярными операционными системами. MySQL обладает возможностью управления структурами, полнотекстовым поиском и индексацией, поддерживает Unicode.

Для работы над данными клиентов фитнес-центра СУБД MySQL является лучшим выбором, она достаточно производительна и легко доступна.

3.4 Облачные сервисы Google

При реализации веб-приложения необходимо использовать облачные средства Google.

Как правило, под облачным вычислением (сервисом) подразумевается обеспечение пользователя компьютерными ресурсами и мощностями в виде интернет-сервисов. Чаще всего пользователю ничего неизвестно, какими именно вычислительными машинами производится обработка его запросов.

Численность серверов наиболее крупных облаков превышает 1000 экземпляров, которые находятся в ЦОД (центр обработки данных). В свою очередь, они направлены на обеспечение ресурсами приложений, которыми в одно и то же время пользуются миллионы обыкновенных пользователей. Стоит рассмотреть основные достоинства и недостатки облачных средств.

К достоинствам облачных средств можно отнести:

- уменьшаются затраты на покупку дорогостоящих вычислительных машин и программного обеспечения, повышается эффективность IT-структуры;

- нет необходимости переживать за память вычислительной машины, всегда будут оставаться свободные ресурсы;

- проблемы с обслуживанием минимизируются;

- совместимость с любой ОС;

- к документам имеется постоянный доступ;

- данные защищены от потери или кражи.

К недостаткам облачных сервисов можно отнести:

- необходимость постоянного доступа к качественному и быстрому Интернету;

- облачные программы могут медленнее работать;

- не все программы доступны для работы в удаленном режиме.

Таким образом, можно утверждать, что облачные сервисы имеют больше плюсов, чем минусов, в отличие от традиционных методов обработки и хранения данных.

В данном веб-приложении были использованы облачные сервисы Google.

Для того, чтобы иметь доступ к сервисам, необходимо иметь Google аккаунт. В случае если его нет, создать аккаунт легко и просто. Корпорация Google позаботилась о своих пользователях и создала пошаговую инструкцию для регистрации в системе. При регистрации каждому человеку предоставляется почтовый адрес, почтовый ящик, облачный диск, сервис для хранения фотографий с возможностью синхронизации с мобильными устройствами, календарь, где можно фиксировать важные события или составлять планы на каждый день и другие. Поддерживается объем памяти 15 Гб.

Для формирования заявки использовался сервис Google Forms. Для того, чтобы работать с формами нужно зайти на облачный диск (Google

Drive) и создать форму. В открывшемся окне создается структура опроса. На странице с опросом можно изменить фон, название формы и создать вопросы. В качестве шаблона ответа на вопрос можно применить один из следующих вариантов:

- текст (строка/абзац);
- один из списка;
- несколько из списка;
- раскрывающийся список;
- дата;
- время;
- шкала;
- сетка (множественный выбор);
- сетка флажков;
- загрузка файлов.

Вопросы можно сделать обязательными для заполнения. Есть возможность добавить описание к вопросу и перемешать ответы. Доступ к форме может контролировать его владелец. Получаемые данные автоматически сохраняются и есть возможность передавать данные прямо в таблицу на облачном диске. При заполнении формы на почту владельца приходит уведомление об этом. Если опрос уже не актуален его можно закрыть и респондентам будет выводиться сообщение о непринятии ответов.

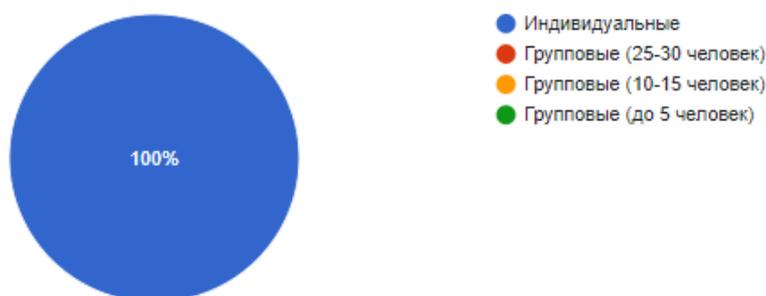
Созданная форма для подачи заявки на посещение включает в себя 13 обязательных вопросов. Форма собирает адреса электронной почты. На четыре вопроса нужно дать краткие ответы, в одном нужно указать дату, в еще одном указать дату и время, в шести выбрать один вариант из представленных и в одном можно выбрать несколько вариантов ответа. Все данные автоматически будут помещаться в Google таблицу на Google Drive владельца.

Еще один плюс Google формы — это автоматически выводимая статистика (рисунок 3.2). Это очень удобно для того, чтобы выявить

аудиторию посетителей, понять в каком направлении необходимо проводить усиленную работу в центре и возможно каким образом поменять политику фитнес-клуба для большего привлечения клиентов.

Какие занятия Вам было бы комфортно посещать?

1 ответ



Какие занятия Вы хотите посещать?

1 ответ



Рисунок 3.2 – Сводная статистика по результатам опроса

В разделе «Контакты» для визуального представления местоположения фитнес-центра прикреплена карта Google Map (рисунок 3.3). На самом сайте Google карт ищем необходимый адрес, отмечаем это место маркером. После проделанных действий копируем ссылку на эту страничку, там будут указаны необходимые высота и широта местоположения для маркера. Данную ссылку вставляем между тегами `<iframe></iframe>` с атрибутом `src` в теле HTML кода.

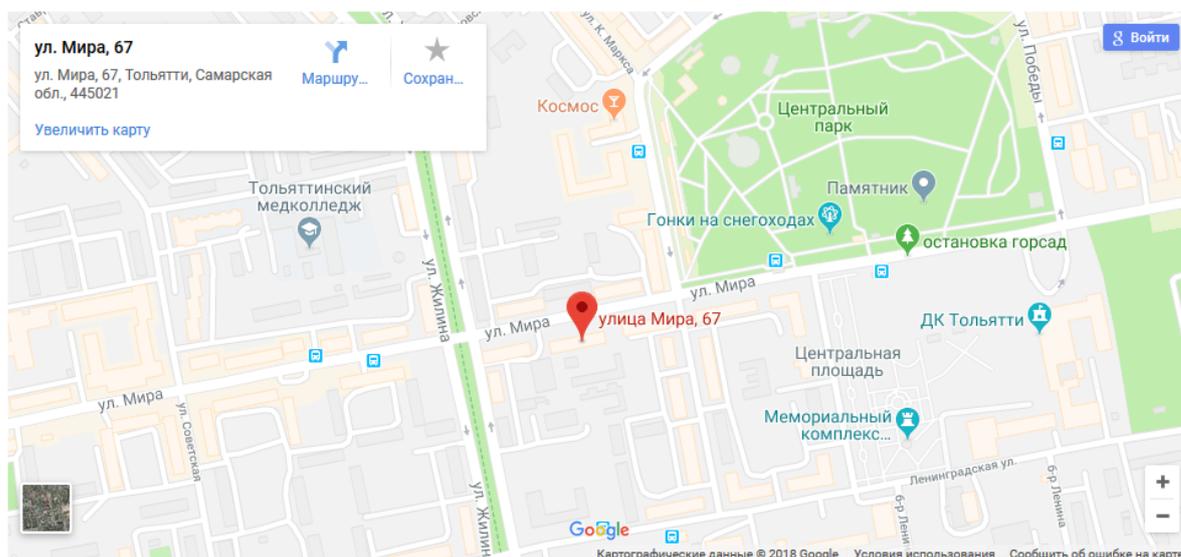
[ГЛАВНАЯ](#)[БЛОГ](#)[О НАС](#)[КОНТАКТЫ](#)[ЗАПОЛНИТЬ ЗАЯВКУ](#)

Рисунок 3.3 – Google карта в разделе «Контакты»

При работе так же используются Google таблицы (Tables). Так как все данные клиентов перенаправляются именно в таблицу. Используется при работе и почтовый ящик как владельца, так и отправителя. Администратор получает оповещение о том, что форма была заполнена, клиент получает на свой почтовый ящик копию заполненной формы.

3.5 Реализация интерфейса и описание работы веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов Google

Разработанное веб-приложение позволяет пользователям видеть список услуг, предоставляемые центром, выбирать тренеров, оставлять заявки на посещение или на получение абонемента, а также получать полезные знания, которые будут помогать им придерживаться здорового образа жизни.

При открытии приложения на главной странице пользователю сразу видно меню для перехода между страницами. Пользователь может

ознакомиться с информацией в веб-приложении, а затем составить и отправить заявку на посещение центра.

На рисунке 3.4 показан раздел, посвященный инструкторам.

Главное меню приложения представлено на рисунке 3.5. На рисунках 3.6 и 3.7 изображен вид приложения с мобильного устройства. Также, в веб-приложении есть раздел, где ведется блог, представленный на рисунке 3.8.

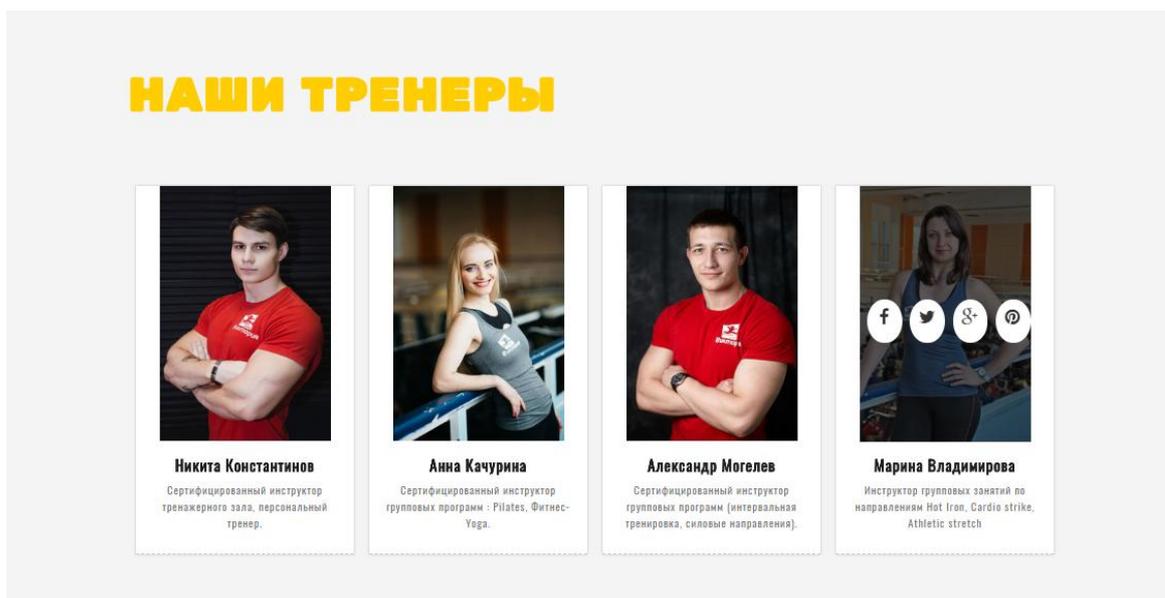


Рисунок 3.4 – Окно выбора тренера

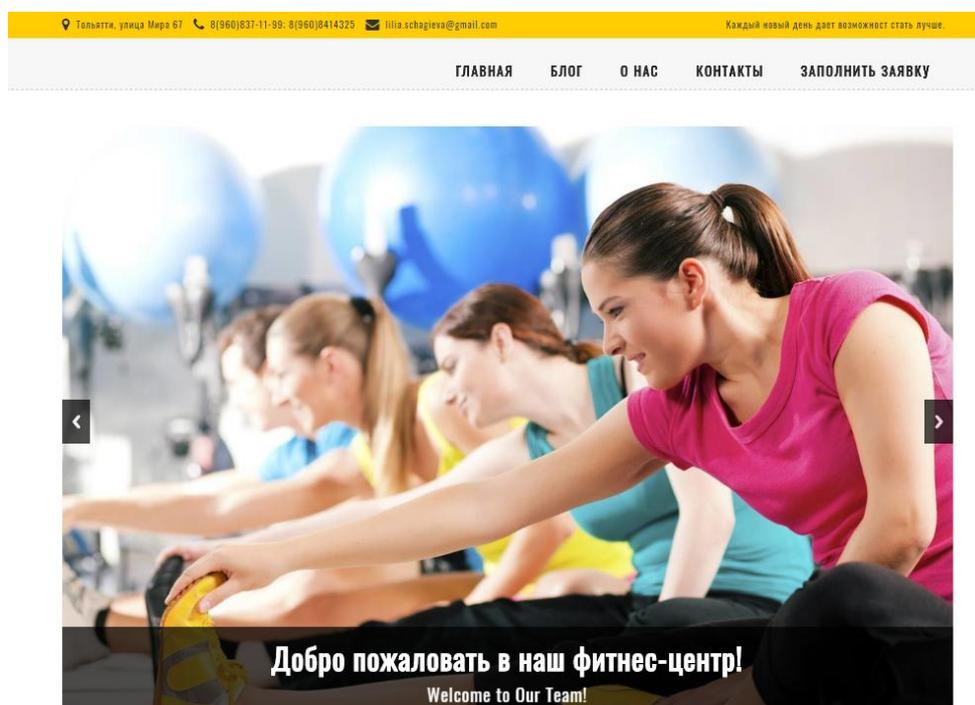


Рисунок 3.5 – Главная страница веб-приложения

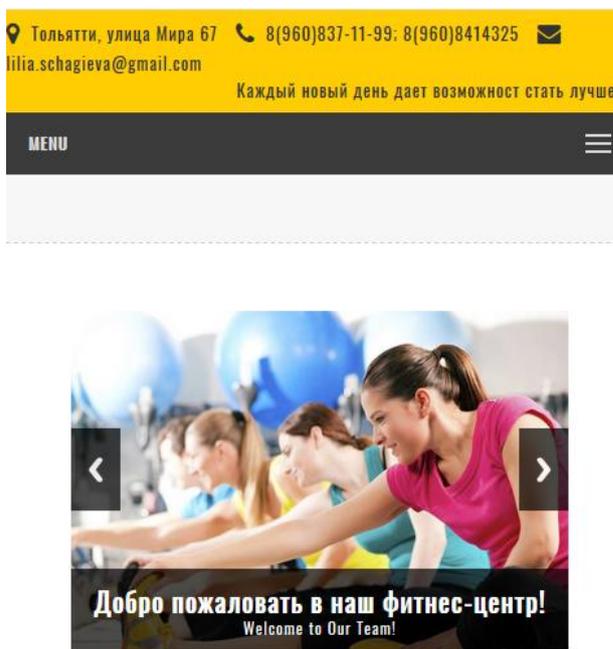


Рисунок 3.6 – Главная страница с мобильного устройства

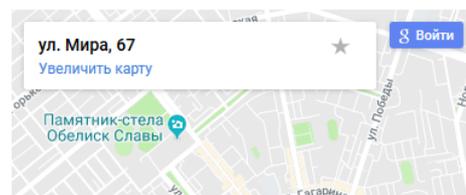
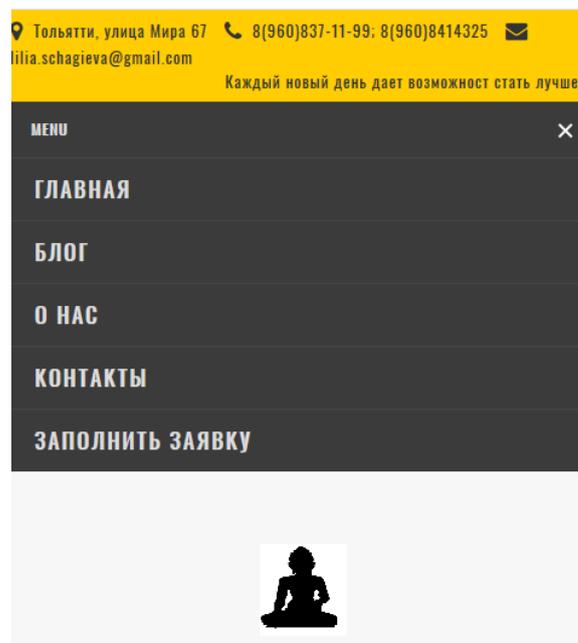
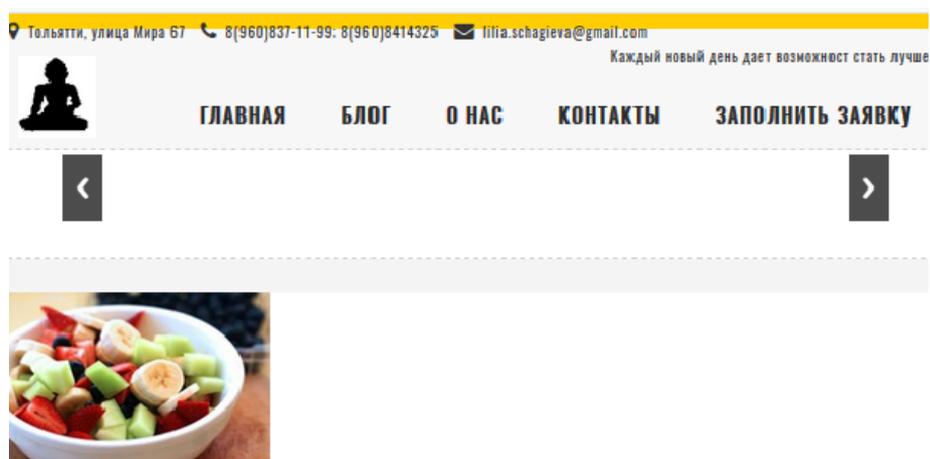


Рисунок 3.7 – Страница «Контакты» с мобильного устройства



Какая еда полезна?

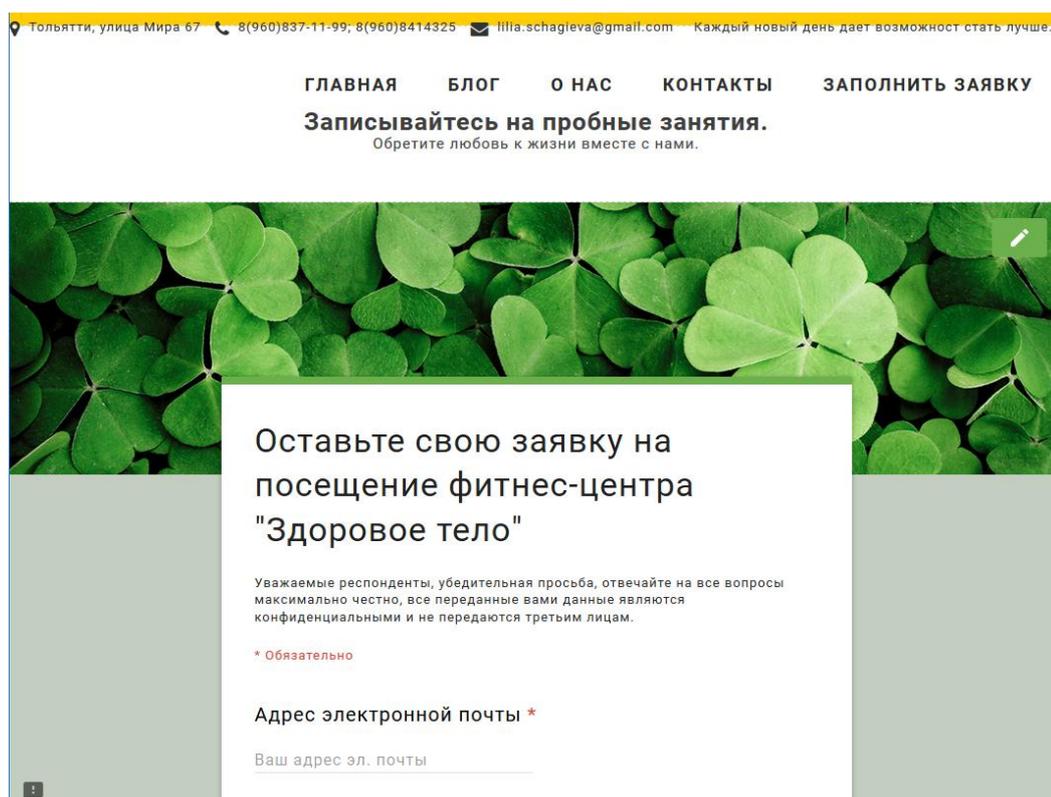
Опубликовано 16.05.2018

Одной из верных рекомендаций по здоровому питанию является совет: «Ешьте овощи и фрукты». Если есть много овощей и фруктов, это поможет вам защититься от болезней сердца и инсульта, предотвратить некоторые виды рака, избежать кишечных болезней, привести в норму давление крови.

Рисунок 3.8 – Блог внутри веб-приложения Рисунок 3.8 – Блог внутри веб-приложения

Главной в данном приложении является форма для подачи заявки.

Данная функция была реализована с помощью облачного средства Google- Google Forms. Данная форма хранится на облачном диске пользователя и просматривать ответы может только он. Готовая форма с рядом вопросов была привязана к странице приложения «Заполнить заявку». Это продемонстрировано на рисунке 3.9.



Тольятти, улица Мира 67 8(960)837-11-99; 8(960)8414325 lilia.schagleva@gmail.com Каждый новый день дает возможность стать лучше.

ГЛАВНАЯ БЛОГ О НАС КОНТАКТЫ ЗАПОЛНИТЬ ЗАЯВКУ

Записывайтесь на пробные занятия.
Обретите любовь к жизни вместе с нами.

Оставьте свою заявку на посещение фитнес-центра "Здоровое тело"

Уважаемые респонденты, убедительная просьба, отвечайте на все вопросы максимально честно, все переданные вами данные являются конфиденциальными и не передаются третьим лицам.

* Обязательно

Адрес электронной почты *

Ваш адрес эл. почты

Рисунок 3.9 – Форма заявки

Таким образом, просмотрев список услуг, создав форму подачи заявки и блог, можно сказать, что данное веб-приложение функционирует и удовлетворяет заданным требованиям.

Выводы по третьей главе

В третьей главе были выбраны технологии для реализации веб-приложения. После изучения некоторых технологий было принято решение, использовать для разработки язык PHP. Также были выбраны языки разметки, HTML и CSS и СУБД MySQL. Так же был реализован интерфейс веб-приложения.

ЗАКЛЮЧЕНИЕ

Целью данной выпускной квалификационной работы являлась разработка веб-приложения формирования заявок фитнес-центра с использованием облачных сервисов Google. В приложении реализованы: форма для создания заявки, блог, привязаны облачные сервисы Google.

В ходе выполнения выпускной квалификационной работы был изучен теоретический материал по технологии разработки веб-приложений, были определены требования к приложению и были выделены задачи, которые необходимо было реализовать.

Проводился анализ существующих технологий разработки веб-приложения. Так же, была представлена логика веб-приложения и архитектура веб-приложения.

При создании формы для подачи заявки были проанализированы уже существующие веб-ресурсы фитнес-центров. Выбран комплекс технических и программных средств реализации. Веб-приложение реализовано на языке PHP и на языках разметки HTML и CSS. В качестве базы данных использована MySQL.

Разработанное веб-приложение позволяет пользователю делать запрос на посещение центра, а также заказать абонемент. К достоинствам разработки можно отнести простоту в эксплуатации и приятный, понятный интерфейс.

В ходе выполнения выпускной квалификационной работы было разработано веб-приложение формирования заявок фитнес-центра с использованием облачных сервисов Google, которое соответствует сформулированным требованиям.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Научная и методическая литература

1. А.П. Пашкевич. Современные технологии программирования: Конспект лекций / А.П. Пашкевич, О.А. Чумаков; Белорусский государственный университет информатики и радиоэлектроники: Минск, 2007. - 64 с.
2. Крис Джамса. Эффективный самоучитель по креативному Web-дизайну / Крис Джамса, Конрад Кинг, Энди Андерсон. - ДиаСофтЮП, 2005. – 672 с.
3. Гущин А. Н. Базы Данных. 2-е изд., испр. и доп.: учебно-методическое пособие / А. Н. Гущин – М. Берлин: Директ – Медиа, 2015. – 311с.
4. Базовый курс // К. Хорстманн. – Москва:Вильямс, 2016. – 464 с. 8. Прохоренок, Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А. Прохоренок, В.А. Дронов. – 4-е изд. – СПб.: Изд-во «БХВ-Петербург», 2015. – 766с. Электронные ресурсы
5. Натан Райлих. PHP в действии / Натан Райлих, Майк Кантелон, Марк хартер. - СПб.: Питер, 2015. - 387 с.
6. Саммерфилд, М. Программирование на Python 3. / М. Саммерфилд ; пер. с англ. А. Киселев. – СПб.: Символ-Плюс, 2009. – 608 с.
7. Таненбаум, Э. Современные операционные системы / Э. Таненбаум. – СПб.: Питер, 2013. - 1120 с.
8. Крахоткина Е.В. Методы и средства проектирования информационных систем и технологий: учебное пособие/ Крахоткина Е.В. - Ставрополь: Северо-Кавказский федеральный университет, 2015. -175 с.
9. Буренин С.Н. Web-программирование и базы данных: учебный практикум / Буренин С.Н. - М.: МГУ, 2014.-132 с.
10. Зандстра. М. PHP. Объекты, шаблоны и методики программирования, 2016, -528с

11. Золотов С. Ю. Проектирование информационных систем : учеб. пособие / С. Ю. Золотов ; Томский гос. ун-т систем управления и радиоэлектроники. - Томск : Эль Контент, 2013. - 86 с.

12. Трутнев Д. Р. Архитектуры информационных систем. Основы проектирования: учеб. пособие. – СПб.: НИУ ИТМО, 2012. – 66 с.

Электронные ресурсы

13. PHP [Электронный ресурс]. – Режим доступа: <http://php.net/>. (дата обращения 22.03.2018)

14. MySQL [Электронный ресурс]. – Режим доступа: <https://www.mysql.com/>. (дата обращения 22.03.2018)

15. Требования к системе: классификация FURPS+ [Электронный ресурс]. – Режим доступа: <https://sysana.wordpress.com/2010/09/16/furps/>. (дата обращения 29.04.2018)

16. Средства автоматизации моделирования UML. [Электронный ресурс]. – Режим доступа <http://citforum.ru/database/case/index.shtml>. (дата обращения 3.04.2018)

17. Google Map [Электронный ресурс]. – Режим доступа <https://www.google.com/maps/>. (дата обращения 4.04.2018)

18. Google Forms [Электронный ресурс]. – Режим доступа <https://docs.google.com/forms/>. (дата обращения 4.04.2018)

Литература на иностранном языке

19. Carlos Buenosvinos, Keyvan AKbary. Domain-Driven Design in PHP. - Springer, Inc., 2015.

20. David Skla. Learning PHP: A Gentle Introduction to the Web's Most Popular Language / David Skla. - O'Reilly, 2016. – 416с.

21. Maurice Charbit, Gérard Blanchet. Learning PHP, MySQL, & JavaScript, Volume 1. ISTE Ltd, 2015.

22. Larry Ullman. PHP Advanced and Object-Oriented Programming: Visual QuickPro Guide, 3rd edition. Inc.,2012.

23. Alan Mark Davis. Just Enough Requirements Management: Where

Software Development Meets Marketing. — Dorset House, 2015.

24. Alan Dennis, Barbara Haley Wixom, David Tegarden: Systems Analysis and Design with UML - 4th Edition, Wiley, 2012.

25. Grinberg. M. Flask Web Development: Developing Web Applications with Python 2nd Edition, 2018, -316p

ПРИЛОЖЕНИЕ А

Фрагмент исходного кода для отправления сообщения персоналу

```
<?php
$text = "<span style='color:red; font-size: 35px; line-height: 40px; margin:
10px;'>Ошибка! Попробуйте еще раз.</span>";

if(isset($_POST['submitcontact']))
{
    $name=$_POST['Имя'];
    $email=$_POST['email'];
    $message=$_POST['Сообщение'];
    $to="https://docs.google.com/spreadsheets/d/1v8ojci2xNzQrBOBGsGwi6io
sMJ2gNzqP8zYXanbKZqk/edit#gid=857043692";
    $subject = "Contact Form";
    $message = " Name: " . $name . "\r\n Email: " . $email . "\r\n Message:\r\n" .
$message;

    $from = "https://docs.google.com/spreadsheets/d/1v8ojci2xNzQrBOBGsGw
i6iosMJ2gNzqP8zYXanbKZqk/edit#gid=857043692";
    $headers = "From:" . $from . "\r\n";
    $headers .= "Content-type: text/plain; charset=UTF-8" . "\r\n";
    if(@mail($to,$subject,$message,$headers))
    {
        $text = "<span style='color:blue; font-size: 35px; line-height: 40px;
margin: 10px;'>Your Message was sent successfully !</span>";
    }
}
?>
```

ПРИЛОЖЕНИЕ Б

Фрагмент исходного кода главной страницы

```
<!DOCTYPE html>
<html lang="ru-ru" dir="ltr">
<head>
  <title>Фитнес-центр Здоровое тело </title>
  <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
  <link rel="stylesheet" href="css/grid.css">
  <link rel="stylesheet" href="css/style.css">
  <link rel="stylesheet" href="css/menu.css">
<script src="js/1.js"></script>
<script src="js/script.js"></script>
<script src="js/res.js"></script>

  <script>
    $(function () {
      $("#slider").responsiveSlides({
        auto: true,
        pager: false,
        nav: true,
        speed: 1700,
        namespace: "callbacks",
        before: function () {
          $(' .events').append("<li>before event fired.</li>");},
        after: function () {
          $(' .events').append("<li>after event fired.</li>");
        }
      });
    });
  </script>
```

```

<div style=' clear: both; text-align:center; position: relative;'>
    </div>
</head>
<body>
<div class="wrap-body">
<div class="top">
<div class="grid">
<div class="row">
<div class="f-left">
<span><i class="fa fa-map-marker"></i> Тольятти, улица Мира 67</span>
<span><i class="fa fa-phone"></i> 8(960)837-11-99; 8(960)8414325</span>
<span><i class="fa fa-envelope"></i> lilia.schagieva@gmail.com</span>
    </div>
    <div class="f-right">
    <span>Каждый новый день дает возможность стать лучше.</span>
    </div></div></div></div>
<header>
    <div class="wrap-header zerogrid">
    <div class="row">
    <div id="cssmenu">
<ul><li class='active'><a href="index.html">Главная</a></li>
    <li><a href="archive.html">Блог</a></li>
    <li><a href="single.html">О нас</a></li>
    <li><a href="contact.html">Контакты</a></li>
    <li><a href="contact.html">Заполнить заявку</a></li></ul>
    </div></div></div>
</header>
<div class="bg-white">
    <div class="grid">

```

//Слайдшоу

```
<div class="callbacks_container">
  <ul class="rslides" id="slider">
    <li>
      <div class="caption">
        <h1>Добро пожаловать в наш фитнес-центр!</h1>
        <span>Welcome to Our Team!</span>
      </div></li>
    <li>
      <div class="caption">
        <h1>Мы всегда рады новым клиентам.</h1>
        <span>Мы работаем, чтобы вы полюбили себя и свое тело.</span>
      </div></li>
    <li>
      <div class="caption">
        <h1>Записывайтесь на пробные занятия.</h1>
        <span>Обретите любовь к жизни вместе с нами.</span>
      </div></li></ul></div>
  <div class="clear"></div></div></div>
</body></html>
```