

Министерство образования и науки Российской Федерации
Филиал Федерального государственного автономного образовательного учреждения
высшего образования
«Южно-Уральский государственный университет»
(национальный исследовательский университет)
в г. Нижневартовске

Кафедра «Информатика»

ДОПУСТИТЬ К ЗАЩИТЕ

И.о.зав.кафедрой «Информатика»

к.ф-м.н, доцент

/ А.В.Ялаев

« ____ » _____ 2018 г.

Разработка информационной системы по обработке данных анкетирования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ-09.03.04. 2018.420.ПЗ ВКР

Консультанты

Экономическая часть

к.э.н., доцент

/А.В.Прокопьев/

« ____ » _____ 2018г.

Безопасность жизнедеятельности

к.ф-м.н, доцент

/ А.В.Ялаев /

« ____ » _____ 2018 г.

Руководитель работы

к.ф-м.н, доцент

/ Д.В.Топольский /

« ____ » _____ 2018 г.

Автор работы

обучающийся группы НвФл-422

/ Б.Б. Гасымов /

« ____ » _____ 2018г.

Нормоконтролер

старший преподаватель

/Л.Н.Буйлушкина /

« ____ » _____ 2018г.

Нижневартовск 2018

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1 ПОСТАНОВКА ЗАДАЧИ.....	7
1.1 Анализ предметной области.....	7
1.2 Обзор существующих аналогов.....	10
1.3 Формулировка задачи.....	14
2 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА.....	23
2.1 Выбор средств разработки для реализации системы.....	23
2.2 Обзор и выбор СУБД.....	27
2.3 Обзор и выбор языка программирования.....	29
3 РАЗРАБОТКИ ДОСКИ ОБЪЯВЛЕНИЙ.....	35
3.1 Разработка системной модели доски объявлений.....	35
3.2 Разработка инфологической и даталогической моделей данных.....	38
3.3 Разработка интерфейса доски объявлений.....	40
3.4 Разработка программного кода.....	44
3.5 Пример использования доски объявлений.....	45
4 ОРГАНИЗАЦИОННО–ЭКОНОМИЧЕСКИЙ РАЗДЕЛ.....	48
4.1 Первоначальные затраты.....	48
4.2 Регулярные ежемесячные расходы.....	50
4.3 Прибыльность доски объявлений.....	51
5 БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ.....	54
5.1 Характеристика условий труда программиста.....	54
5.2 Требования к производственным помещениям.....	56
5.3 Эргономические требования к рабочему месту.....	60

5.4. Режим труда и отдыха при работе с компьютером	62
ЗАКЛЮЧЕНИЕ	66
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	68

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А. СВИДЕТЕЛЬСТВО О ПУБЛИКАЦИИ	71
ПРИЛОЖЕНИЕ Б. ЛИСТИНГ	72
ПРИЛОЖЕНИЕ В. КОМПАКТ-ДИСК.....	85

ВВЕДЕНИЕ

В последнее время широкое развитие Интернет спровоцировало новые формы продвижения товаров, образуя своеобразные коммуникативные пространства – виртуальные площадки. Такие сетевые формы коммуникаций возникают на основе спроса буквально из ничего. Одной из таких форм являются электронные доски объявлений, нацеленные на пользователя, который в данный момент ищет тот или иной товар, услугу.

На территории Ханты-Мансийского автономного округа-Югры такой вид виртуальных площадок особенно актуален в танцевальной индустрии. Это обусловлено мизерным либо отсутствующим вообще ассортименте танцевальных товаров. Что является большой проблемой для родителей танцующих детей. Поэтому есть покупатели, которые желают приобрести бывший в употреблении танцевальный товар в хорошем состоянии по привлекательной цене. Реализация танцевальных товаров актуальна не только для покупателей, но и для продавцов. Дети вырастают из нарядов очень быстро, поэтому практически все они находятся в отличном состоянии и со временем накапливаются, поэтому очевидно что родители рано или поздно задаются вопросом «Куда их деть?».

Объектом исследования в данной выпускной квалификационной работе является продажа танцевальной одежды, предметом исследования – интернет коммерция с использованием современных технологий для создания и администрирования доски объявлений.

Исходя из вышеперечисленных проблемы, была поставлена цель выпускной квалификационной работы: разработка электронной доски объявлений по продаже танцевальной одежды, обуви и аксессуаров.

Для реализации поставленной цели, необходимо решить следующие задачи:

- проанализировать предметную область;
- выбрать программно-инструментальные средства разработки;
- спроектировать базу данных (далее – БД);

– спроектировать и разработать пользовательское Web-приложение;
– рассчитать экономическую эффективность разрабатываемого интернет ресурса;

– рассмотреть вопросы безопасности жизнедеятельности [3].

Выпускная квалификационная работа состоит из введения, пяти разделов, заключения и приложений.

1 ПОСТАНОВКА ЗАДАЧИ

1.1 Анализ предметной области

Сайты бесплатных объявлений – это сайты, на которых размещаются объявления. Их эффективность зависит от большего трафика, больше людей – больше продаж

Электронная доска объявлений функционально подобна обыкновенной: это сайт, где каждый желающий может вывесить своё объявление, а все посетители сайта – прочитать его. Электронная доска объявлений, как правило, поделена на несколько тематических разделов, согласно содержанию объявлений.

Большинство электронных досок – бесплатные. Для размещения своего объявления пользователю нужно лишь ввести в специальной форме его тему, своё имя/псевдоним либо название организации, а также координаты: адрес электронной почты, почтовый адрес, телефон, URL своего сайта и т. п. (набор данных зависит от конкретного ресурса). Как правило, отображаются только имена авторов и темы объявлений, а для просмотра полного текста объявления пользователь должен щёлкнуть по ссылке, ведущей к нему. В некоторых досках объявления могут подавать только зарегистрированные пользователи, в некоторых – все. Сейчас в интернете существует тысячи и даже десятки тысяч досок объявлений. Обычно каждая из них посвящается какому-либо отдельному виду объявлений. Существуют национальные доски объявлений, предназначенных для жителей конкретной местности.

Электронные доски объявлений бывают двух видов: моделируемые (те, у которых есть так называемый модератор – человек, контролирующий работу этой доски) и не моделируемые – работающие автоматически.

Большинство досок объявлений в интернете предполагают размещение объявлений о продаже и покупке товаров и услуг. Но есть и такие сервисы, которые предлагают размещать объявления о передачи различных вещей в дар, а

также о поиске новых хозяев для домашних животных, которые передаются в добрые руки безвозмездно. Бесплатные объявления – это наилучший способ заявить о себе, своих товарах и услугах в сети Интернет. На страницах досок объявлений представлены тысячи бесплатных объявлений о недвижимости, авто, услугах, поиске работы, знакомствах, покупке и продаже оборудования, компьютеров, средств связи и прочему. Доски объявлений позволяет разместить объявление с фотографией и без дополнительной регистрации. Вы можете разместить не только бесплатное объявление, но и объявление на коммерческой основе – для максимальной эффективности.

Объявление – заголовок, описание и фотографии товара или услуги.

Модерация – проверка объявления на уникальность, спам и список запрещённых к продаже товаров.

Роботы для проверки – это различные системы проверки объявлений на уникальность, на проверку указания в тексте или заголовке запрещённых слов, на проверку указания контактов в объявлении и т.д.

Модератор – человек, который проверяет сомнительные объявления, те что отсеял робот, и принимает решение о блокировке или пропуске объявления.

Основным преимуществом рекламы через объявления от других видов рекламы является, то что она нацелена на пользователя, который в данный момент и ищет тот или иной товар, услугу.

Перечислим еще несколько преимуществ:

– минимальный бюджет – на большинстве досок объявлений можно размещаться бесплатно;

– быстрый эффект – первые заказы можно получить уже через пару минут после публикации.

Наряду с преимуществами у объявлений есть недостатки, к основным относят:

– малый срок эффективности – наибольшее внимание привлекают те объявления, который находятся на первой странице;

– неэффективность в некоторых сферах – контекстная реклама бесполезна, если ваш товар или услугу не ищут в интернете, например, продуктовый магазин, или вы выводите на рынок новый продукт, который еще не знают, например, новый «гаджет»;

– тщательная модерация – на основных досках объявлений проверка объявлений достаточно серьезна, для размещения большого количества объявлений, приходится выдумывать различные обходы системы.

Самым первым этапом разработки программного обеспечения по праву называется процедура проведения всестороннего анализа выдвинутых заказчиком требований к создаваемому программному обеспечению, чтобы определить ключевые цели и задачи конечного продукта. В рамках этой стадии происходит максимально эффективное взаимодействие нуждающегося в программном решении клиента и сотрудников компании–разработчика, в ходе обсуждения деталей проекта помогающих более четко сформулировать предъявляемые к программному обеспечению требования. Результатом проведенного анализа становится формирование основного регламента, на который будет опираться исполнитель в своей работе – технического задания на разработку программного обеспечения. Техническое задание должно полностью описывать поставленные перед разработчиком задачи и охарактеризовать конечную цель проекта в понимании заказчика.

В рамках данной выпускной работы описан процесс разработки электронной доски объявлений по продаже танцевальной одежды, обуви и аксессуаров.

Проведя анализ предметной области в танцевальной индустрии, изучив специфику товаров, были выявлены тематические разделы разрабатываемого интернет ресурса «Доска объявлений «Dance Top Collection» представленные на рисунке 1.1.

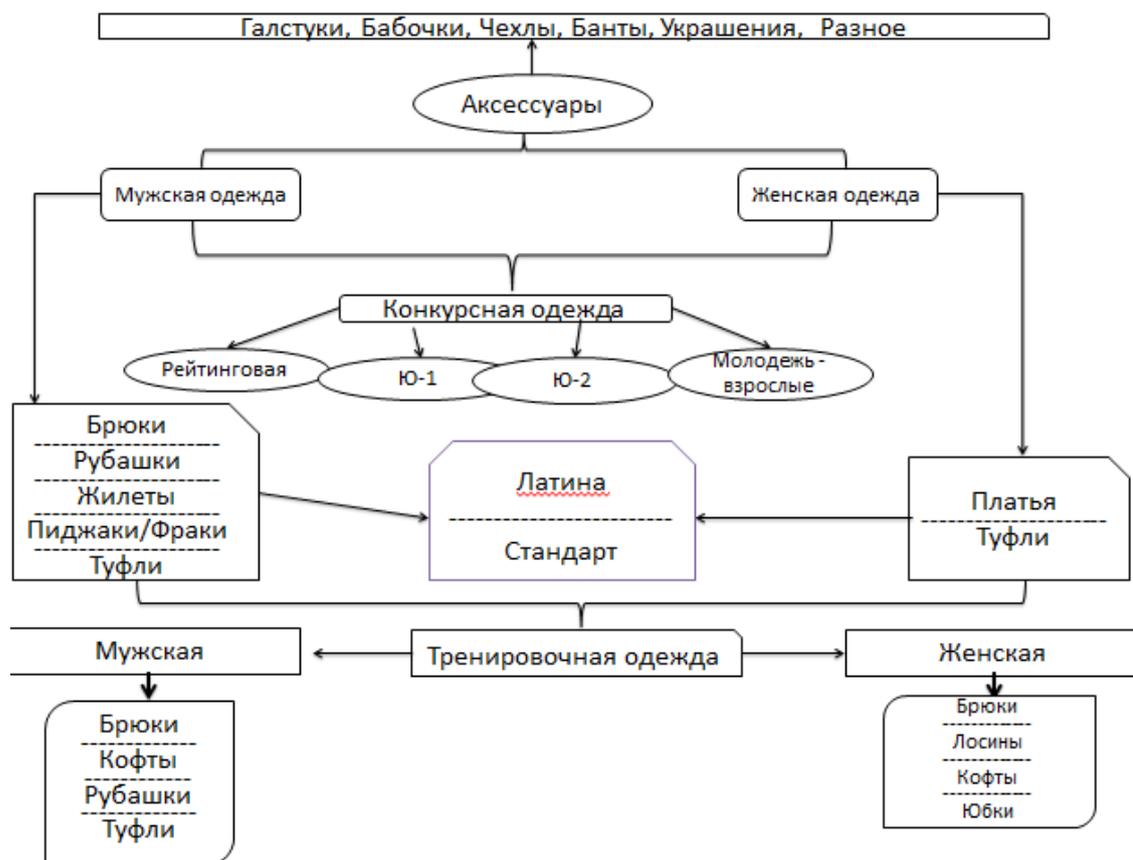


Рисунок 1.1 – Тематические разделы

1.2 Обзор существующих аналогов

Проведя анализ уже существующих разработок, в первую очередь было уделено большое внимание наиболее массовым и популярным электронной доски объявлений. Проводя анализ, отмечался подход к построению интерфейса. Анализируя решения сторонних сервисов можно отметить использование ленты на сервисе «Avito» (Рисунок 1.2). Сервис «Юла» предлагает интересное решение по использованию панели навигации с возможностью поиска по категориям (Рисунок 1.3).

Сравнивая подходы этих сервисов, было отмечено удобство пользования ленточным стилем отображения объявлений. Однако вариант реализации у электронной доски объявлений «Avito» не был найден эталонным. Панель навигации и поиска по категориям в сервисе «Юла», может быть в некотором виде заимствована.

Однако на начальном этапе такое решение излишне в виду малого количества объявлений и немногочисленностью категорий размещаемых объявлений. Также отмечен фильтр, схожий с «яндекс–маркетом». Такой фильтр удобен и может использоваться и на разрабатываемом сайте.

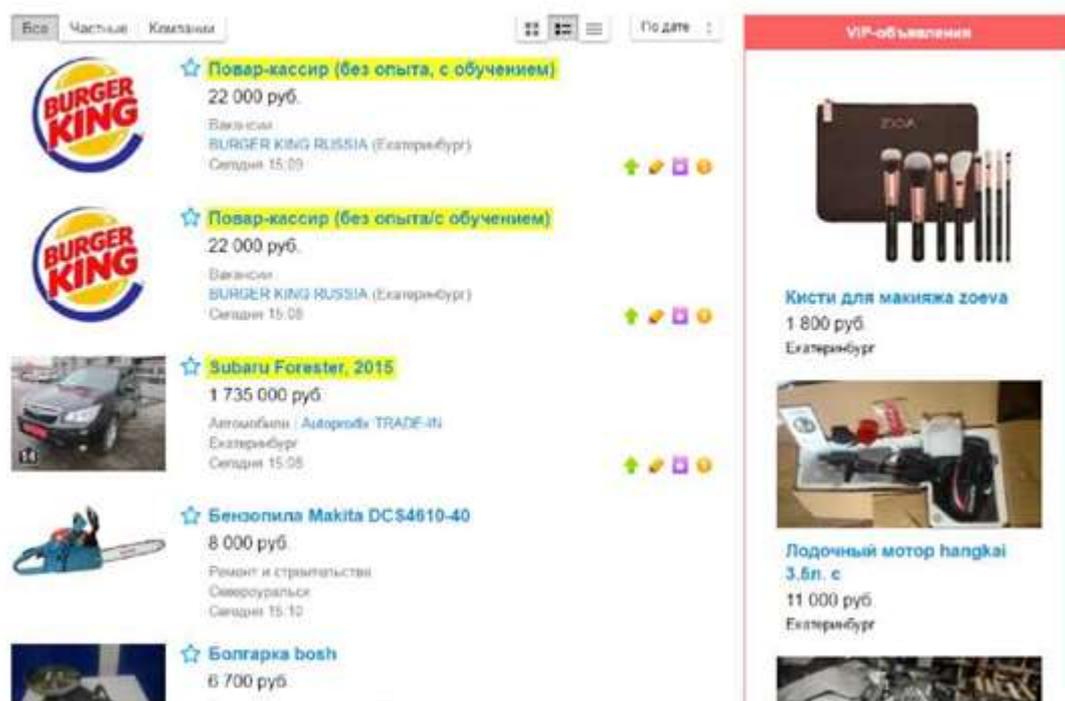


Рисунок 1.2 – Окно сайта Avito.ru

Особое внимание было уделено интерфейсу сайта «ВКонтакте», а именно разделу «группы» (Рисунок 1.4). Так как основная аудитория потенциального электронной доски объявлений – пользователи этой социальной сети.

Здесь стоит отметить похожее решение в размещении объявлений, подобно сервису «Avito». Вдобавок к этому, имеется навигационная панель слева с возможностью перейти на другие страницы. Подобная навигационная панель могла присутствовать и на разрабатываемом сервисе. Например, переход в личные сообщения или просмотр избранных объявлений.

Также в качестве существующего электронной доски объявлений по размещению объявлений, хоть не бывших в употреблении, а новых товаров, можно рассматривать «Яндекс Маркет».

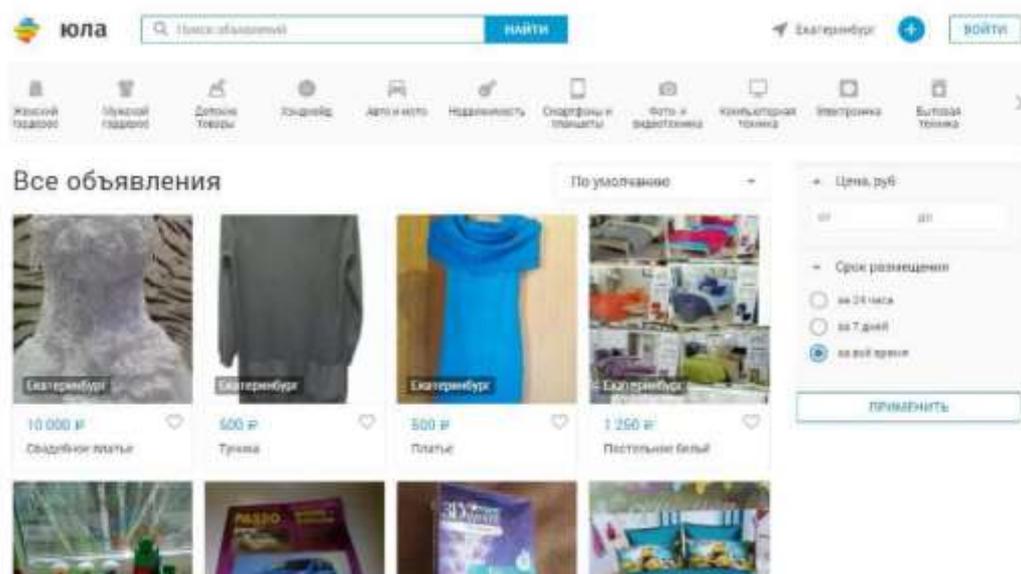


Рисунок 1.3 – Окно сайта Юла

Это один из популярнейших сервисов в Российской Федерации и за её пределами определенно требует анализа. Рассматривая главную страницу можно заметить сгруппированный список категорий (Рисунок 1.5). Такой подход позволяет быстро перейти к любой категории.

Выбрав любую категорию осуществляется переход на страницу с просмотром товаров (Рисунок 1.6). Как можно заметить, размещение блоков с объявлениями, схоже с таковыми на электронной доске объявлений «Avito» и «Вконтакте». Этим стоит подчеркнуть распространённость такого подхода и бесспорно его удобство.

Таким образом, несмотря на наличие существующих решений в области электронных досок объявлений разработка узкоспециализированной доски объявлений по продаже танцевальной одежды является актуальной.

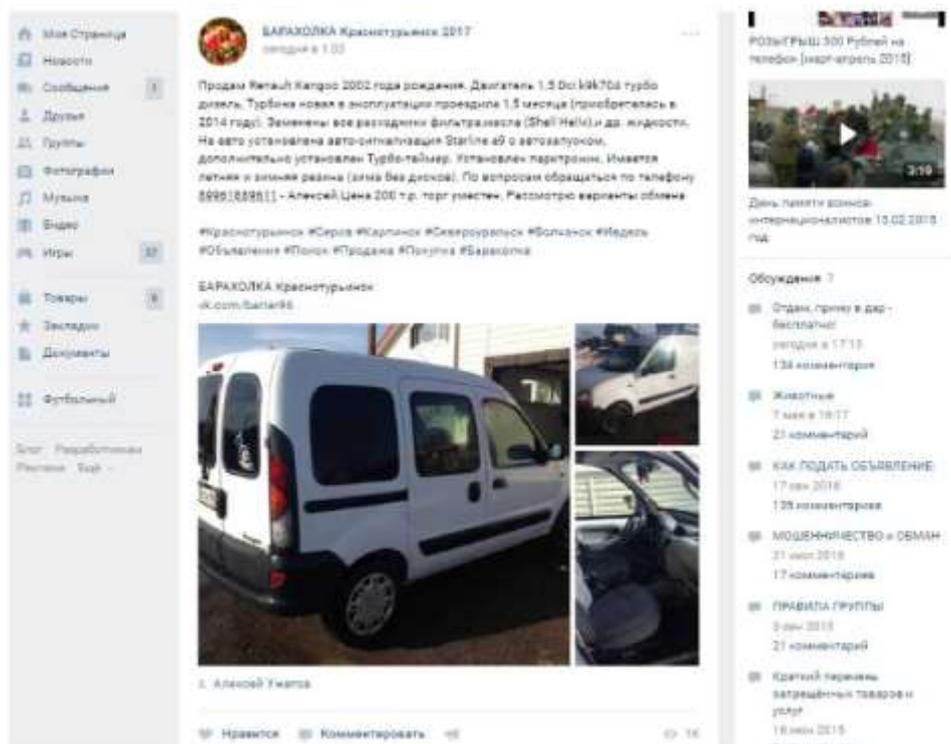


Рисунок 1.4 – Снимок экрана страницы «Барахолка» в социальной сети «ВКонтакте»

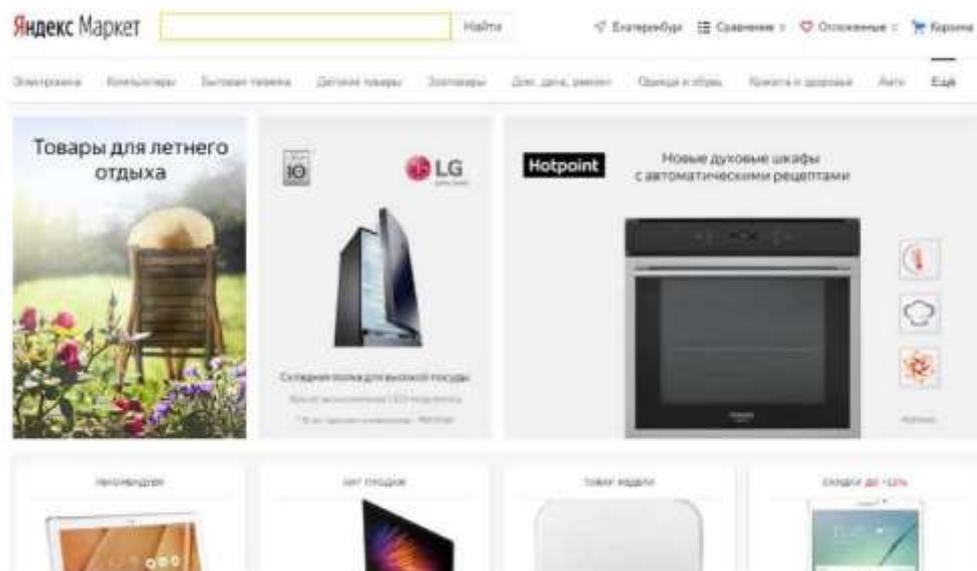


Рисунок 1.5 – Снимок экрана главной страницы «Яндекс Маркет»

Рисунок 1.6 – Снимок экрана главной страницы «Яндекс Маркет»

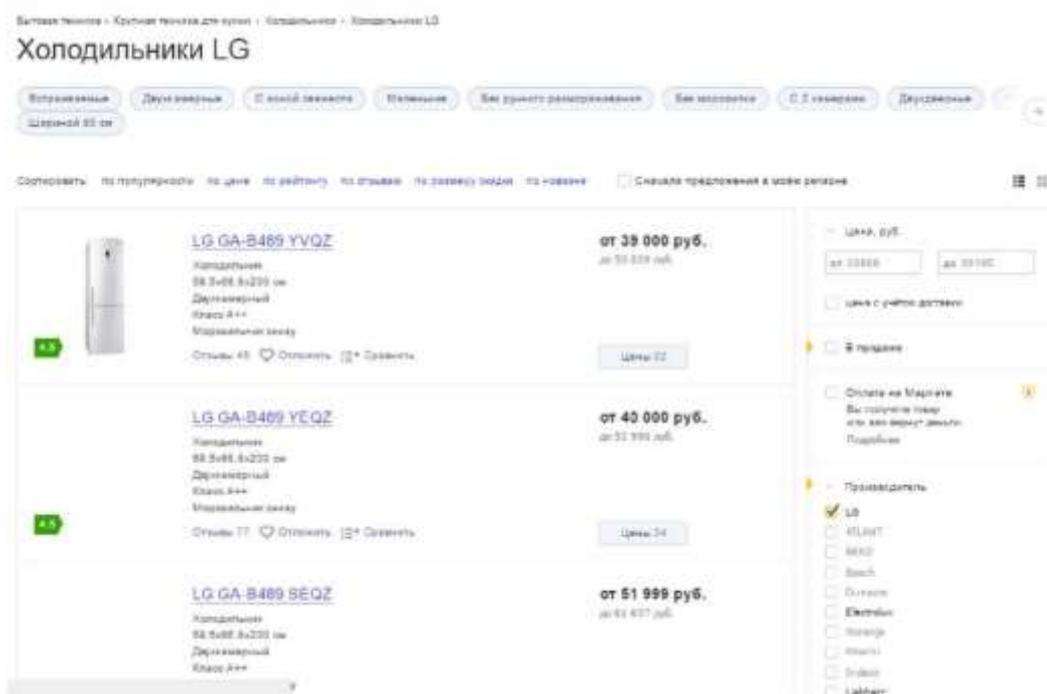


Рисунок 1.6 – Снимок экрана главной страницы «Яндекс Маркет»

1.3 Формулировка задачи

1.3.1 Основные требования

Функциональность электронной доски объявлений можно описать такой концепцией: размещение объявления – связь с продавцом – продажа. Были выделены следующие функции:

1. Регистрация пользователей.
2. Возможность добавления объявлений.
3. Отображение объявлений, возможность поиска, по словам и критериям.
4. Возможность связаться с продавцом.

Первоочередная функция электронной доски объявлений – возможность регистрации. Это позволит идентифицировать пользователя и поспособствует удобной связи с продавцом. На этапе регистрации предполагается внесение пользователем информации о своём имени и номера телефона.

Добавление объявления позволяет поместить текстовое описание. А также по желанию добавить фотографии, осуществить привязку к карте.

Отображение объявлений, пожалуй, самая основная функция электронной доски объявлений. Однако помимо отображения определенное удобство составит и поиск объявлений по критериям.

Возможность связи с продавцом заключается в возможности отображения его номера и телефона и возможности отправки сообщений для дальнейшего обмена ими.

1.3.2 Требования к интерфейсу

Общие требования к сервису: возможность регистрации, возможность рассылки push-уведомлений, скорость загрузки приложения и страниц (чтобы приложение выдерживало большое количество одновременных посетителей).

Для пользователя выделим следующий функционал: у пользователя должен быть личный кабинет (страница управления) для управления своим аккаунтом, заполнение профиля (личной информации о себе), необходимо создать удобную, простую в заполнении форму подачи объявлений, информирование о неверной форме заполнения объявления, информирование о неграмотном написании слов (подчеркивание красным), окно подачи (редактирования) объявления, возможность отслеживания (просмотра) своих объявлений.

Для просмотра чужих объявлений заказчик выделил следующий функционал: наличие ленты объявлений, адаптивный поиск (исключение товаров, выбранная категория).

При подаче объявлений: Загрузка изображения (фотографии) в данный момент с данного устройства (сделал фото и сразу загрузил).

Для описания основных функций пользователей проектируемого web-ресурса разработаны диаграммы прецедентов (Рисунок 1.7).

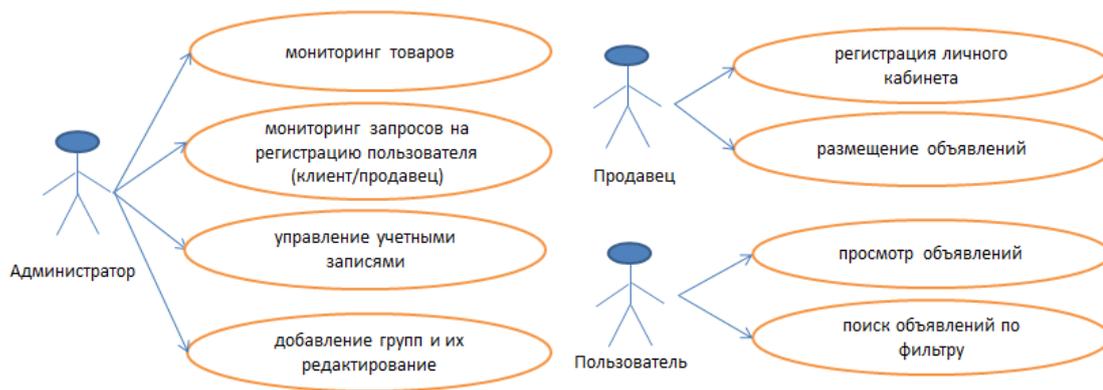


Рисунок 1.7 – Диаграмма прецедентов

1.3.3 Требования к составу страниц

Под требованиями пользователей подразумевают возможность, без вопросов и с лёгкостью получить необходимую информацию (товар) для удовлетворения потребностей. Подробный список требований представлены в таблице 1.1.

Таблица 1.1 – Требования пользователя

Название требования	Описание требования
Регистрация в системе	Должна предоставляться возможность регистрации пользователя. Для регистрации на сайте доски объявлений необходимо ввести логин, адрес электронного почтового ящика, пароль.
Вход в систему	Предоставляется возможность входа в систему с помощью ввода адреса электронного почтового ящика и пароля.
Восстановление пароля в системе	Предоставляется возможность изменить пароль. Для этого необходимо ввести адрес электронного почтового ящика и пароль будет выслан на указанный адрес.
Личный кабинет	В личном кабинете предоставляется возможность изменения личных настроек (имя, фамилия, почта)
Просмотр каталога товаров	Возможность быстрого и удобного ознакомления с товарами, представленными на сайте

Администратор – пользователь, обладающий максимальными правами в системе. Чаще всего, этот человек занимается сортировкой товаров по

категориям/каталогам. Иногда обязанность администратора занимает человек, который разрабатывает этот сайт. Требования администратор представлены в таблице 1.2.

Таблица 1.2 – Требования администратора

Название требования	Описание требования
Управление каталогом товаров	Каталог товаров – раздел сайта в котором можно просматривать все товары, доступные для покупки.
Добавление /изменение/ удаление товаров	При добавлении / изменении товаров должна быть указана следующая информация: название товара, описание товара, стоимость товара, статус товара, материал. Должна быть возможность загружать изображения товаров с компьютера.
Управление материалами	Возможность добавлять / удалять материалы, из которых изготавливаются товары.
Добавление изображений	Администратор должен иметь возможность загрузки изображений товаров. На добавление изображения существует ограничения: максимальный размер файла 2 МВ, формат изображений «JPG», «PNG», «GIF». При загрузки допускается любое расширение, но система автоматически изменяет изображение каталога до 800x500 px., изображение товара 800x800 px., миниатюра товара 300x300 px.
Управление заказами	Возможность менять статус заказа (в наличии, нет в наличии или отобразить дату появления в наличии)
Управление зарегистрированными пользователями	Имеется возможность редактировать / удалять пользователей из системы. Редактировать можно статус пользователя, роль в системе (администратор / покупатель), пароль пользователя.

Приведем обязательные элементы для всех страниц:

- шапка сайта;
- ссылка на профиль;
- ссылки на разделы.

Структура разделов сайта:

- 1) О доске объявлений.
- 2) Каталог товаров:

- товарная группа 1 товар N;
- товарная группа N.

3) Личный кабинет:

- профиль пользователя;
- страница объявлений;
- архив объявлений.

Главная страница. На главной странице должно быть:

- меню;
- каталог объявлений;
- информация.

Каталог объявлений:

- многоуровневый каталог продукции с подразделами;
- на странице выводятся основные разделы;
- товарная группа;
- список товаров, представленный в линейку по 3 шт.

Формат представления товара в каталоге:

- картинка;
- название;
- цена.

По клику на название или картинку – переход на страницу с подробным описанием товара.

Товар.

На странице приводится полное описание товара:

- название;
- картинка (несколько картинок до 5) с возможностью увеличения;
- описание в виде текстового поля;
- цена;
- категория.

1.3.4 Анализ системных требований к доске объявлений

Системные требования – это более детализованное описание пользовательских требований. Они обычно служат основой для заключения контракта на разработку программной системы и поэтому должны представлять максимально полную спецификации системы в целом. Системные требования также используются в качестве отправной точки на этапе проектирования системы.

В принципе системные требования определяют, что должна делать система, не показывая при этом механизма ее реализации. Но, с другой стороны, для полного описания системы требуется детализированная информация о системной архитектуре. На то существует ряд причин.

1. Первоначальная архитектура системы помогает структурировать спецификацию требований. Системные требования должны описывать подсистемы, из которых состоит разрабатываемая система.

2. В большинстве случаев разрабатываемая система должна взаимодействовать с уже существующими системами, это накладывает ограничения на архитектуру новой системы.

3. В качестве внешнего системного требования может выступать условие использования для разрабатываемой системы специальной архитектуры.

Таблица 1.3 – Системные требования

Требования	Описание требования
Работа через SSL для безопасности	Web-сайт должен работать с использованием протокола SSL (HTTPS) для обеспечения защищённого соединения и предотвращения хищения персональных данных пользователей (таких как номер банковской карты)
Работа в популярных браузерах	Web-сайт должен корректно работать в браузерах «Google Chrome», «Mozilla Firefox», «Internet Explorer» версии 9 и выше, «Safari».

Продолжение таблицы 1.3

Требования	Описание требования
Требования к хостингу	Для разворачивания серверной части системы на боевом сервере необходимо, чтобы он обладал следующими характеристиками: минимум 20 Гб свободного пространства на жестком диске, 4 Гб оперативной памяти, четырех ядерный процессор с частотой 3 ГГц, пропускная способность канала 10 МБ/сек.
Требование к web-серверу	В качестве web-сервера должен быть выбран «Apache» из-за высокой скорости работы и возможности гибкой настройки.
Требования к безопасности	При разработке приложения необходимо предусмотреть защиту от взлома на всех этапах использования. Так же необходимо предусмотреть защиту от «csrf» атаки.
Требования к скорости работы сервера	Время ответа на любой запрос не должно превышать 20 секунд.
Требования к системе управления базой данных (далее – СУБД)	Должен использоваться «PostgreSQL», так как данная СУБД предоставляет высокую скорость работы и хорошие возможности.

Системные требования к электронной доске объявлений – это набор характеристик компьютера, которые обеспечивают работу определенного программного обеспечения. Чаще всего это следующие сведения:

- операционные системы, на которых может работать ИС (в подавляющем большинстве на текущий момент это версии Windows);
- процессор (модель, тактовая частота, производитель, или рейтинговое число для процессоров AMD);
- оперативная память (RAM), например, 1000 Мб (1 Гб);
- требования к видеокарте (производитель графического процессора его модель, объем видеопамати);
- необходимое пространство на жестком диске, например, 5 Гб свободного пространства;
- иногда указываются требования к звуковым картам;
- могут быть требования к составу модулей и сервисов ОС.

Системные требования бывают минимальные и рекомендованные.

Таким образом, для нормальной работы с ИС рабочие места пользователей должен иметь характеристики не ниже указанных:

- операционная система Windows XP/VISTA/7;
- установленный браузер, желательно Google Chrome не ниже 10–й версии, IE не ниже 10–й версии;
- ОЗУ 512 МБ и выше;
- процессор частотой не ниже 1 ГГц;
- видеокарта с видеопамью не менее 128 МБ;
- сетевая карта.

Поскольку разрабатывается интернет сайт, то немаловажными являются также требования к хостингу, на котором данный сайт будет размещаться.

Перечислим их:

- Apache / Windows IIS / NGINX;
- PHP 5.2 и выше;
- MySQL.

Выводы по разделу один:

В процессе работы над данной выпускной квалификационной работы был проведен глубокий анализ предметной области, в которой были описаны, что представляют из себя доски объявления в целом, выявлены преимущества и недостатки досок объявлений, и их виды.

Был проведен обзор аналогов, где было выяснено, насколько работа актуальна. Рассмотрены функциональные требования, и в ходе рассмотрения, было, выделено три функции программы, требования к интерфейсу, требования к составу страниц, структура разделов сайта и анализ системных требований к доске объявлений.

Точная формулировка цели проекта и анализ предметной области – основа разработки любого программного обеспечения.

В данном разделе были рассмотрены и проанализированы проблемы, существующие при проведении инвентаризационного учёта в филиале, а также была обоснована необходимость разработки программного продукта.

Исходя из этих проблем, было принято решение о разработке базы данных. Необходимым условием при создании нового проекта является знание предметной области и анализ, который проведен в данном разделе.

2 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

Разработка программного продукта включает в себя 4 этапа:

- 1) Обоснование выбора средств разработки для реализации системы.
- 2) Обоснование выбора СУБД.
- 3) Реализация СУБД.
- 4) Проектирование структуры и интерфейса программы.

2.1 Выбор средств разработки для реализации системы

2.1.1 Методологии проектирования информационных систем

Информационная модель – модель объекта, представленная в виде информации, описывающей существенные для данного рассмотрения параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая путем подачи на модель информации об изменениях входных величин моделировать возможные состояния объекта. Информационные модели нельзя потрогать или увидеть, они не имеют материального воплощения, потому что строятся только на информации. Информационная модель – совокупность информации, характеризующая существенные свойства и состояния объекта, процесса, явления, а также взаимосвязь с внешним миром.

При построении информационной модели следует придерживаться стандарта IDEF0. IDEF0 – методология функционального моделирования, при помощи которой, изучаемая система предстает перед разработчиками в виде набора взаимосвязанных функций, благодаря наглядному графическому интерфейсу.

Как правило, модели IDEF0 несут в себе концентрированную и сложную информацию, поэтому для ограничения их перегруженности в данном стандарте приняты соответствующие ограничения сложности:

– функциональные блоки на диаграмме ограничены тремя–шестью. максимальный предел (шесть) заставляет разработчиков придерживаться иерархии при описании сложных предметов, а минимальный предел гарантирует, что на диаграмме будет достаточно деталей для оправдания ее создания;

– количество подходящих (выходящих) к одному функциональному блоку интерфейсных дуг ограничено четырьмя.

Конечно, строго следовать этим ограничениям совсем не обязательно, однако они являются довольно практичными в работе.

Ресурсно-продуктовая характеристика IDEF0-модели:

– ресурсы IDEF0-модели – полное множество объектов, поставка которых экономическим агентам является необходимым условием начала и завершения исполнения функций последних;

– стандартный ресурс – ресурс, который система IDEF0\DOCTOR может интерпретировать при составлении текстовых документов – отчетов различного рода;

– стандартные ресурсы вносятся в IDEF0-модель автоматически при инсталляции новой IDEF0-модели;

– бизнес-процесс (далее – БП) – функциональная модель экономического агента;

– не существует объектов, поставляемых бизнес-процессам, которые не являются ресурсами;

– IDEF0-модель разрабатывается не с целью определения связей между БП, а с целью описания возможных направлений поставок ресурсов;

– элементарный процесс – БП, находящийся на нижнем уровне IDEF0-модели (или не декомпозированный БП);

– порядок исполнения системы БП задается только для элементарных процессов, непосредственную поставку и потребление ресурсов осуществляют только элементарные процессы;

– продукт – ресурс на выходе БП. услуги и товары являются продуктами (продукцией);

– ресурсы, ассоциированные с БП – ресурсы, поступающие на вход БП и высвобождаемые с выхода БП в соответствии с алгоритмом преобразования ресурсов, осуществляемым рассматриваемым БП;

– цикл элементарного процесса – процедура, описывающая производство единицы продукта элементарного процесса;

– продукт элементарного процесса может быть составлен из нескольких более простых продуктов (например, автомобиль состоит из запчастей);

– правило активации элементарного процесса – элементарный процесс, безусловно, исполняется при поставке ему всех ресурсов, удовлетворяющих его параметрическим требованиям и необходимых для завершения его цикла;

– порядок исполнения БП определяется не произвольно задаваемой последовательностью исполнения БП относительно друг друга, а выше приведенным правилом активации;

– заказ БП – запрос элементарного процесса на поставку ресурсов, содержащий полный перечень ресурсов и их характеристик, необходимых для его завершения;

– цена ресурса – стоимостной параметр ресурса, выраженный в единицах платежа и описывающий размер «платы», осуществляемой БП, за пользование ресурсом;

– норма дохода ресурса – стоимостной параметр ресурса, выраженный в процентах и описывающий превышение цены ресурса над его себестоимостью.

2.1.2 Выбор и обоснование средств разработки

Сущность структурного подхода к разработке информационных систем заключается в ее декомпозиции (разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь

делятся на подфункции, подразделяемые на задачи и так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимоувязаны. При разработке системы «снизу–вверх» от отдельных задач ко всей системе целостность теряется, возникают проблемы при информационной стыковке отдельных компонентов. Все наиболее распространенные методологии структурного подхода базируются на ряде общих принципов. В качестве двух базовых принципов используются:

- принцип решения сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения.
- принцип иерархического упорядочивания – принцип организации составных частей проблемы иерархические древовидные структуры с добавлением новых деталей на каждом уровне [3].

Выделение двух базовых принципов не означает, что остальные принципы являются второстепенными, поскольку игнорирование любого из них может привести к непредсказуемым последствиям. Основными из этих принципов являются следующее:

- принцип абстрагирования – заключается в выделении существенных аспектов системы и отвлечения от несущественных;
- принцип формализации – заключается в необходимости строгого методического подхода к решению проблемы;
- принцип непротиворечивости – заключается в обоснованности и согласованности элементов;
- принцип структурирования данных – заключается в том, что данные должны быть структурированы и иерархически организованы [8].

Логический уровень – это абстрактный взгляд на данные. На нем данные представляются так, как выглядят в реальном мире, и могут называться так, как они называются в реальном мире. Объекты модели, представляемые на логическом уровне, называются сущностями и атрибутами. Данная модель может

быть построена на основе другой логической модели, например на основе модели процессов. Логическая модель данных является универсальной и никак не связана с конкретной реализацией СУБД [5, 6].

Основные компоненты диаграммы Data Modeller – это сущности, атрибуты и связи. Каждая сущность является множеством подобных индивидуальных объектов, называемых экземплярами. Каждый экземпляр индивидуален и должен отличаться от всех остальных экземпляров. Атрибут выражает определенное свойство объекта. С точки зрения БД (физическая модель) сущности соответствует таблица, экземпляру сущности – строка в таблице, а атрибуту колонка таблицы [14].

Построение модели данных предполагает определение сущностей и атрибутов, т.е. необходимо определить, какая информация будет храниться в конкретной сущности или атрибуте. Сущность можно определить как объект, событие или концепцию, информация о которой должна сохраняться.

2.2 Обзор и выбор СУБД

На основе выборной модели рассмотрим СУБД основанных или работающих на реляционной модели. СУБД так же должна обеспечивать определенный тип связи между сущностями из разных таблиц.

Из большого множества СУБД при проектировании были рассмотрены следующие варианты: SQLite, MySQL и PostgreSQL.

Рассмотрим SQLite, которая обладает очень мощной встроенной системой управления. Это легко встраиваемая в приложение СУБД. Так как система базируется на файлах, то она представляет широкий сектор инструментов работы, по сравнению с сетевыми СУБД.

Преимущества SQLite:

– простая структура файлов, что позволяет легко переносить БД на различные машины;

- СУБД поддерживает большую часть стандартов SQL;
- быстрый перенос на новые устройства;
- быстрое получение ответа на запрос, так как данные хранятся в памяти устройства.

Недостатки SQLite:

- невозможность добавить в систему пользователя;
- невозможность увеличить производительность СУБД, так как она напрямую зависит от ЭВМ на котором находится;
- некорректная работа с большим объёмом данных;
- плохая масштабируемость;
- низкий уровень безопасности;
- не поддерживается многопользовательская система [16].

Рассмотрим СУБД MySQL. MySQL является бесплатно распространяемой СУБД и хорошо работает с web-приложениями.

Преимущества MySQL:

- наличие GUI приложения;
- высокий уровень безопасности;
- масштабируемость;
- поддержка многопользовательской системы;
- гибкая настройка;
- присутствует транзакционных механизм;
- за счёт упрощения некоторых стандартов SQL СУБД может работать значительно быстрее других аналогов[18].

Недостатки MySQL:

- медленная обработка простых SQL запросов;
- MySQL обладает сложной настройкой.

PostgreSQL свободно распространяемая и соответствует стандартам SQL. СУБД рассчитана на работу с корпоративными приложения, в отличие от MySQL.

Достоинства PostgreSQL:

- масштабируемость;
- высокий уровень безопасности;
- при использование сложных пользовательских конструкций в SQL запросах;
- возможность сохранения своих процедур, что позволят легко расширить функционал;

Недостатки PostgreSQL:

- при хранении больших объёмов текста и поиска по тексту;
- СУБД плохо адаптировано под многопоточность.

Анализ рассмотренных программных средств помог выбрать в качестве СУБД PostgreSQL для управления базой данных разрабатываемой ИС. Основными причинами для выбора послужили:

- наличие механизма транзакции;
- высокий уровень безопасности;
- разделения системы доступа на пользователей;
- простота настройки;
- высокая скорость чтения данных;
- масштабируемость.

2.3 Обзор и выбор языка программирования

Анализ средств разработки был проведен в соответствии с этапами проектирования и разработки данного электронной доски объявлений.

Первоочередным было проектирование интерфейса электронной доски объявлений (мобильного приложения и web-сайта) на основе актуальности его разработки, потенциальной аудитории и прочих параметров. Проектирование интерфейса осуществлялось подручными средствами в виде набросков, чертежей и схем.

В последующем процессе создания макетов и шаблонов приложения и сайта были использованы графические редакторы. В разработке логотипа электронной доски объявлений был использован Adobe Illustrator, а для разработки макетов приложения и сайта использовалась комбинация Adobe Photoshop и Adobe Illustrator. Выбор этих средств разработки обусловлен их профессиональным функционалом, а также навыками, приобретенными по работе с ними в процессе обучения в университете.

В разработке web-сайта были использованы технологии HTML5, CSS3 с элементами CSS4, JavaScript, язык программирования PHP и база данных MySQL.

Данный выбор обусловлен современностью этих стандартов, легкостью верстки, расширенными возможностями и поддержкой всеми современными браузерами. Помимо этого использована библиотека jQuery для придания сайту визуальной привлекательности и удобства пользования.

Для написания серверной части сайта был выбран язык PHP. Такой выбор был обусловлен несколькими причинами. Одна из них – это высокая популярность языка. По данным индекса Tiobe, на март 2017 года, PHP среди всех языков программирования находится на 7 строчке популярности (Таблица 2.1) [18].

Таблица 2.1 – Рейтинг языков программирования по индексу Tiobe

2017	Programming Language	Ratings	Change
1	Java	16.384%	-4.14%
2	C	7.742%	-6.86%
3	C#	5.184%	-1.54%
4	c++	4.409%	+0.14%
5	Python	3.919%	-0.34%
7	PHP	3.174%	+0.61%
6	Visual Basic .NET	3.009%	+0.24%
8	JavaScript	2.867%	+0.33%

По данным индекса PYPL, на июнь 2017 года, PHP среди всех языков программирования находится и вовсе на 3 строчке популярности (Таблица 2.2) [19].

Таким образом, PHP безусловный лидер по полярности среди серверных языков программирования. Такая популярность обусловлена наличием большого набора встроенных средств для разработки web-приложений. Помимо этого PHP поддерживает различные серверы и базы данных (в том числе MySQL). Несомненным фактом за выбор языка PHP является то, что в настоящее время поддерживается подавляющим большинством хостинг провайдеров.

Таблица 2.2 – Рейтинг языков программирования по индексу PYPL

Worldwide	2017		
Ran	Charge Language	Share	Trend
1	Java	22.7 %	-1.3 %
2	Python	15.7 %	+3.5 %
3	PHP	9.3 %	-1.1 %
4	C#	8.3 %	-0.5 %
5	Javascript	7.9 %	+0.5 %

Использование языка PHP в разработке сайта позволило реализовать такие функции как, регистрация пользователей, подтверждение регистрации на электронной почте, взаимодействие с базой данных и хранением там данных о пользователях, объявлениях и возможности ведения личных переписок.

Использование базы данных PostgreSQL, как говорилось выше, обусловлено в частности хорошим взаимодействием с языком PHP, простой, доступностью и поддержкой большинством хостинг провайдеров.

Иных альтернатив используемых технологий производства, кроме серверных языков программирования ASP.NET, Ruby, Python или Perl, практически не имеется. Поддержка стандарта XHTML прекращена в пользу HTML5, а иные серверные языки являются либо достаточно трудными для освоения (Ruby и Python), либо недостаточно популярными, например как Perl.

Программными средствами разработки web-сайта стал текстовый редактор Notepad++. Такой выбор обусловлен его доступностью и, не смотря на это, вполне приличным функционалом (Рисунок 2.1).

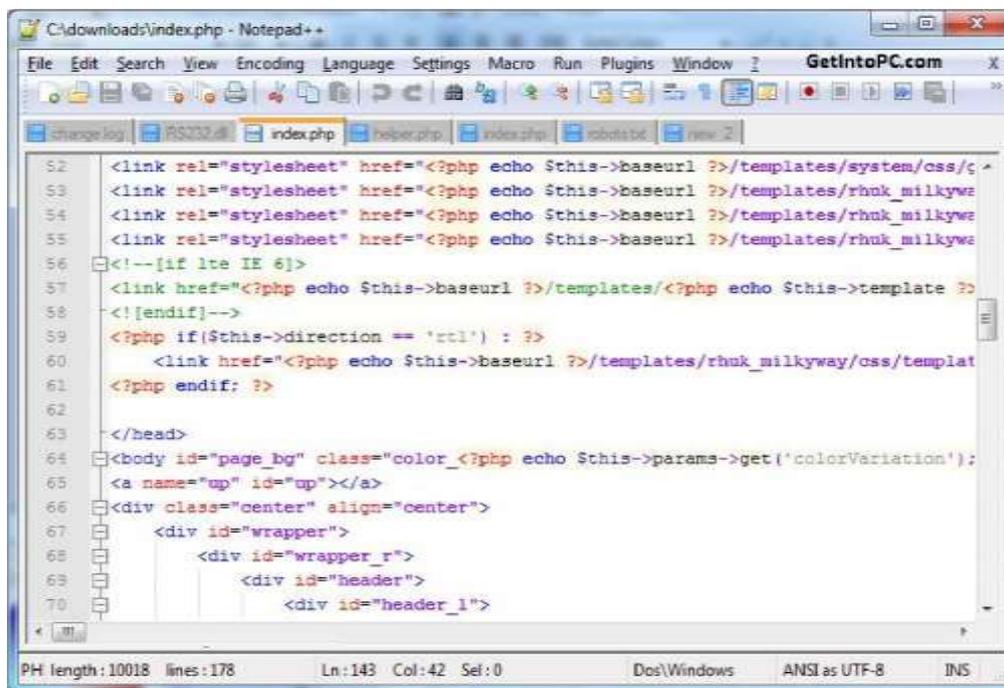


Рисунок 2.1 – Текстовый редактор NotePad++

Среди его возможностей следует отметить подсветку синтаксиса, сворачивание кода, авто дополнение, регулярные выражения и некоторое другие. Его использование возможно как при разработке HTML/CSS кода, так и при разработке PHP/JavaScript сценариев.

Однако анализируя другие средства разработки, были выделены такие IDE (интегрированная среда разработки) как Eclipse PDT и NetBeans IDE. Они являются бесплатными и предоставляют больший функционал, чем обычный текстовый редактор. При разработке JavaScript сценариев, использование IDE стало достаточно необходимым, и дальнейшая разработка ввелась в среде NetBeans (Рисунок 2.2).

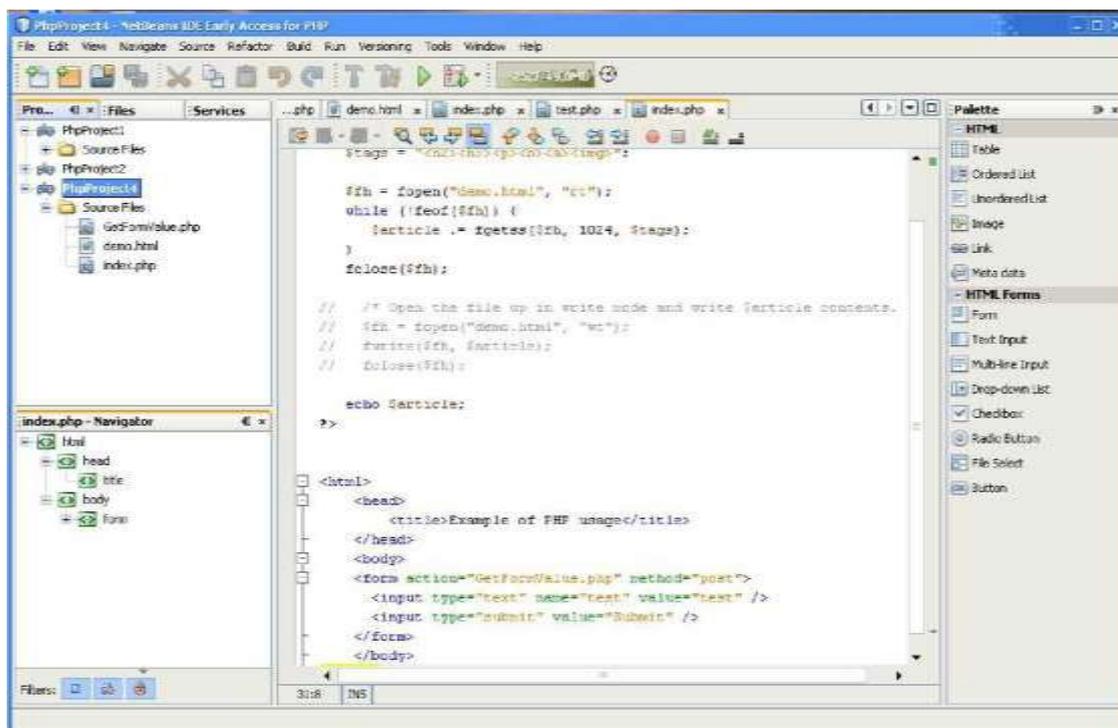


Рисунок 2.2 – Среда разработки NetBeans

В качестве сервера, на котором осуществлялась разработка, и тестирования был выбран Apache. Он уже входит в бесплатный серверный набор DenWep. Также из этого набора для работы с базой данных использовалась программа phpMyAdmin (Рисунок 2.3). В отличие от консольного управления базой данных, такой подход более удобный, визуально привлекательный и очень простой. Например, по нажатию одной кнопки, может выполняться команда длинной более чем одна строка.

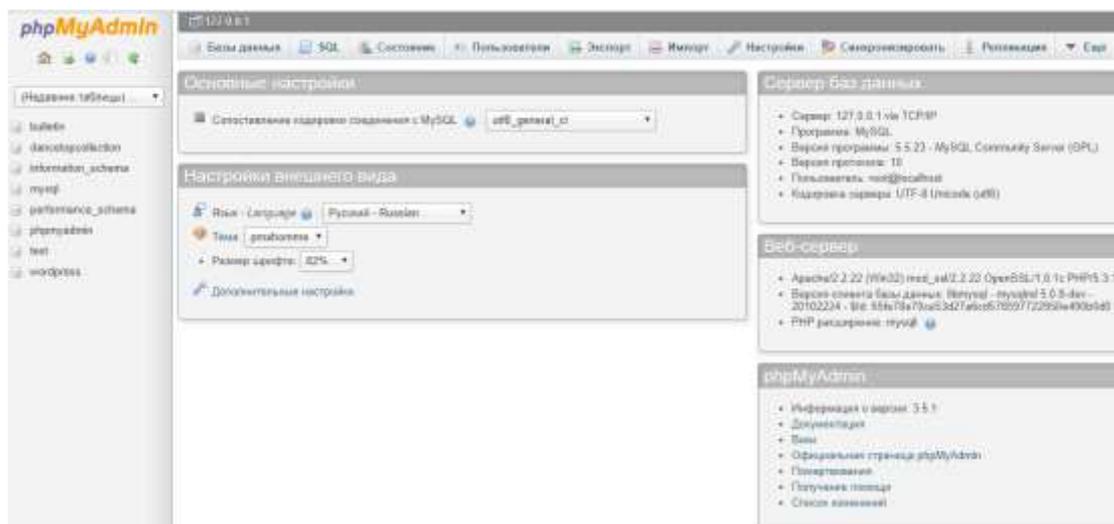


Рисунок 2.3 – РНРMyAdmin

Выводы по разделу два:

В процессе работы над данной выпускной квалификационной работой было проведено более глубокое изучение предметной области, освоены языки программирования РНР и Java. Получены навыки работы с клиент-серверными приложениями и интерфейсами управления в разных средах.

Анализ средств разработки был проведен в соответствии с этапами проектирования и разработки данного электронной доски объявлений. Были описаны преимущества недостатки тех или иных средств разработки.

Программное приложение представляет собой модульное приложение, состоящее из трех программных модулей: серверная часть приложения, клиентская часть приложения. Все разрешенные для публикации исходные модули представлены на компакт-диске (Приложение В). Реализован интерфейс приложения. Таким образом, все поставленные цели по разработке выпускной квалификационной работы, успешно решены.

3 РАЗРАБОТКИ ДОСКИ ОБЪЯВЛЕНИЙ

3.1 Разработка системной модели доски объявлений

Рассмотрим систему управления основными процессами и данными электронной доске объявлений. Основным процессом, который определяет работу с электронной доской объявлений является авторизация пользователей. Мы не будем рассматривать уровень доступа Гость так, как его функциональность сведена до минимума и позволяет только просмотр соответствующих товаров и выделим основные четыре профиля пользователей, которые могут управлять, в той или иной степени электронной доской объявлений: администратор, клиент (Рисунок 3.1).

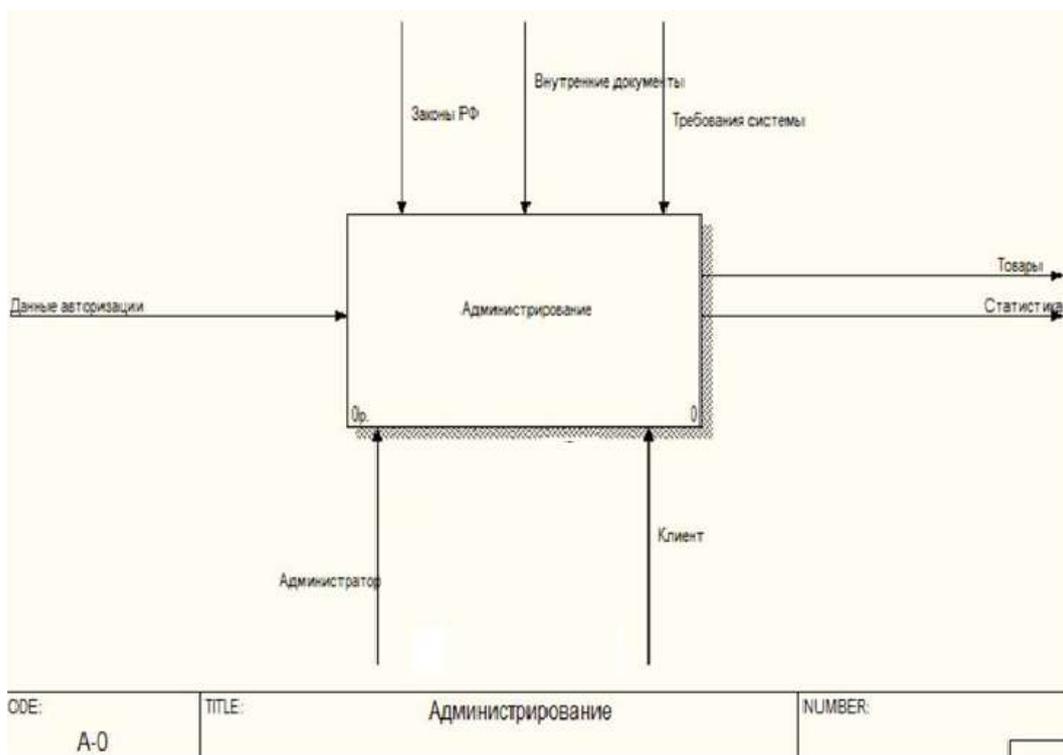


Рисунок 3.1 – Диаграмма A0

При декомпозиции данной диаграммы можно проследить четыре блока процессов (Рисунок 3.2).

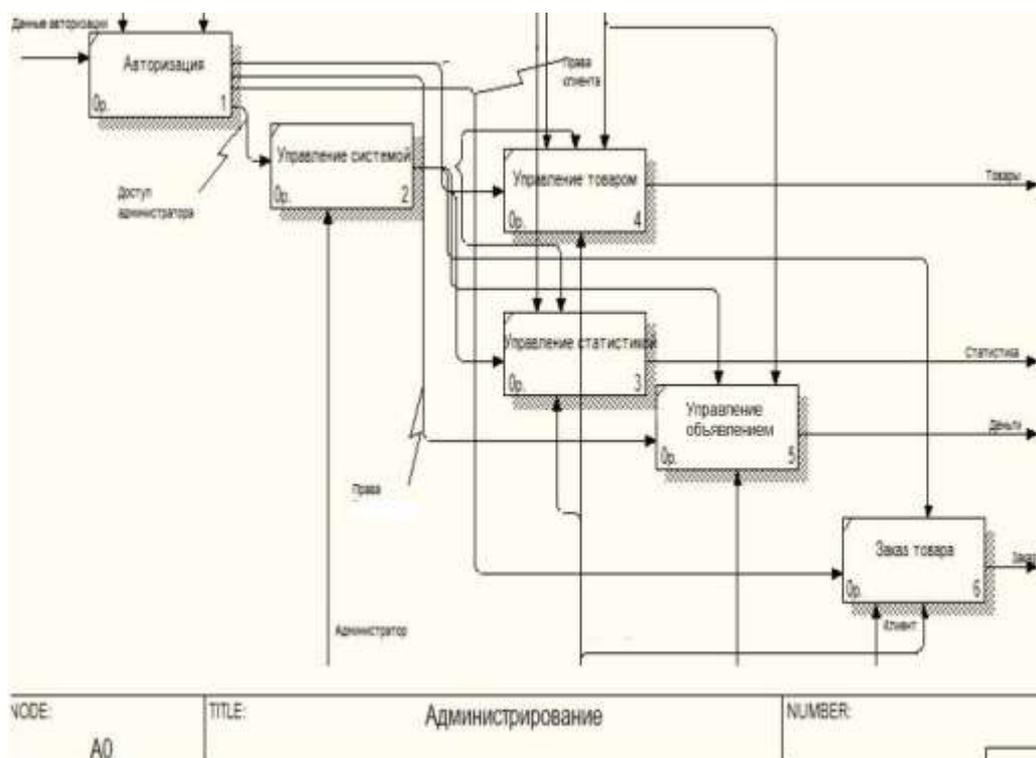


Рисунок 3.2 – Основные процессы в системе электронной доски объявлений

Администратор отвечает за работу всех модулей системы.

Работа клиента по заказу товара предполагает процесс выбора, процесс взаимодействия с продавцом и процесс получения товара.

Продавец управляет всеми процессами, связанными с товаром в объявлении.

При разработке информационной системы был использован нисходящий метод проектирования. Данный метод заключается в том, что сначала строится модульная структура программы в виде дерева [1].

Затем поочередно программируются модули программы, начиная с модуля самого верхнего уровня (головного), переходя к программированию какого-либо другого модуля только в том случае, если уже запрограммирован модуль, который к нему обращается.

После того, как все модули программы запрограммированы, производится их поочередное тестирование и отладка в таком же (нисходящем) порядке. Метод нисходящего проектирования иногда называют функциональной декомпозицией. Каждый модуль отвечает за реализацию одной из функций системы.

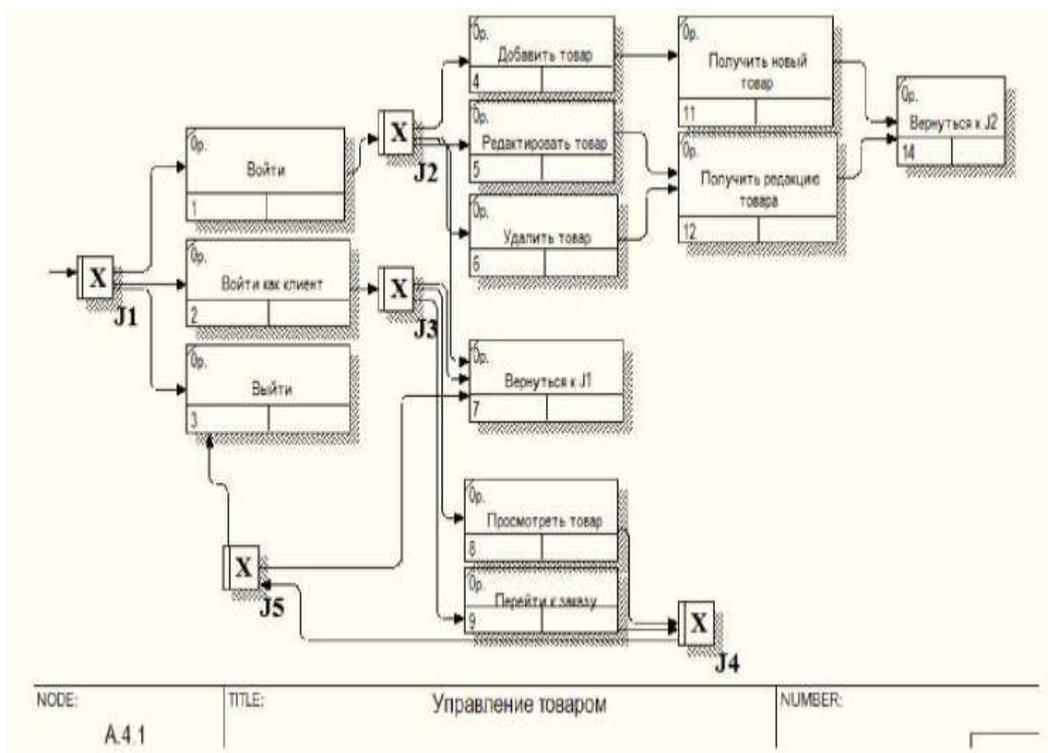


Рисунок 3.3 – Процессы продавца по управлению товаром

Электронная доска объявлений состоит из модулей. Структура модулей представлена на Рисунке 3.4.

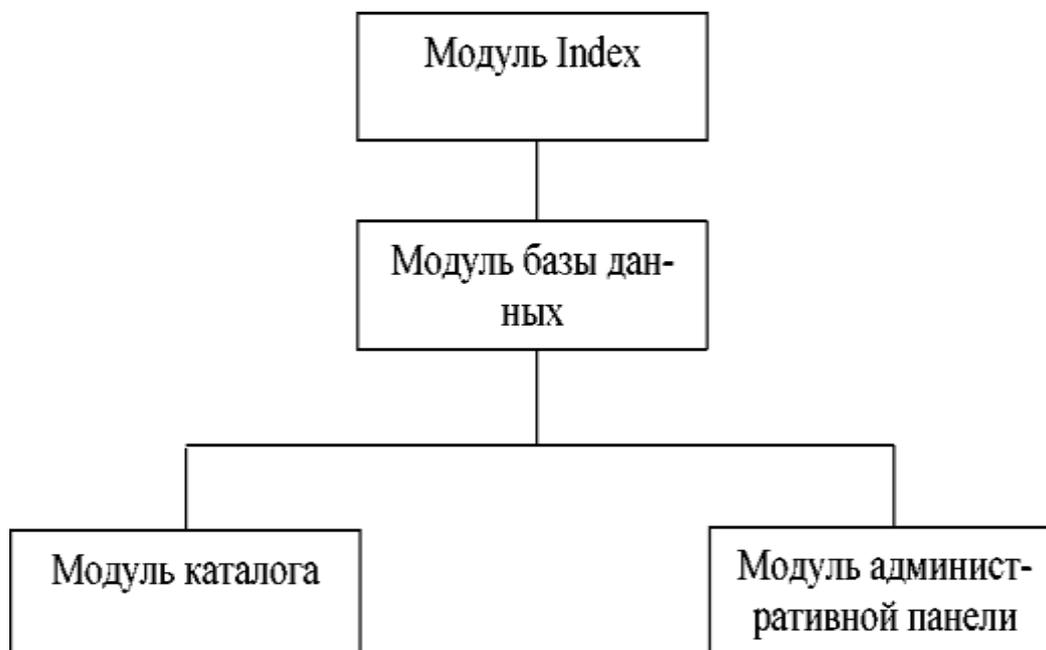


Рисунок 3.4 – Структура модулей электронной доски объявлений

3.2 Разработка инфологической и даталогической моделей данных

Концептуальное (инфологическое) проектирование – построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных [8].

Логическое (даталогическое) проектирование – создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных.

Физическое проектирование – создание схемы базы данных для конкретной СУБД.

На рисунке 3.5 представлены уровни моделей данных.

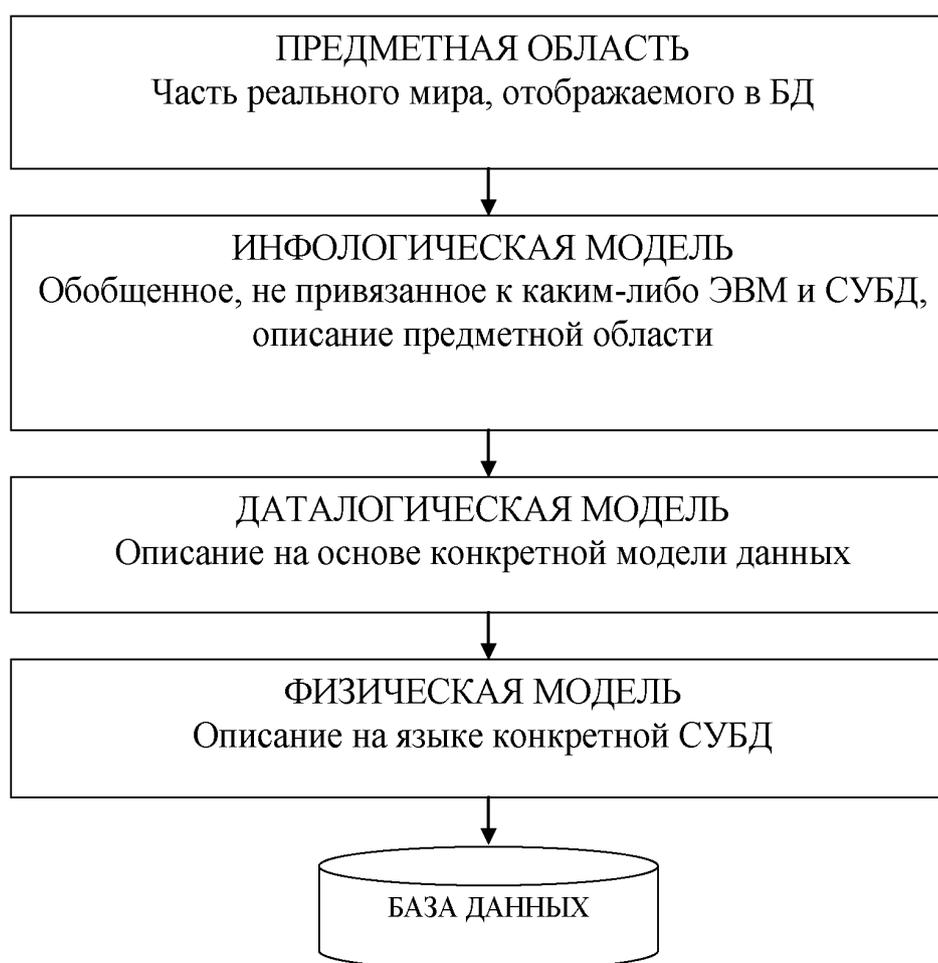


Рисунок 3.5 – Уровни моделей базы данных

При проектировании моделей предметной области обычно применяется метод семантического моделирования. Данный метод представляет из себя моделирование структуры данных, при этом опираясь на смысл этих данных [13]. Как инструмент семантического моделирования часто используют ER- диаграммы (Entity-Relationship Diagrams, «сущность-связь»).

ER-диаграммы применяют для разработки моделей данных. Они представляют собой стандартный метод определения данных, а также отношений между ними. Этим способом производится детализация хранилищ данных [19]. Главными понятиями ER-диаграммы считают атрибут, сущность, связь.

Сущность представляет собой класс однотипных объектов, данные о которых должны быть учтены в модели [18]. В свою очередь, каждая сущность диаграммы должна иметь свое наименование, выражаемое существительным в единственном числе, и изображаемое в виде прямоугольника на схеме.

Экземпляр сущности представляет собой конкретный представитель этой сущности [18].

Под атрибутом сущности понимают именованную характеристику, которая является определенным свойством этой сущности [18]. Наименование атрибута может быть выражено существительным (с характеризующими прилагательными). Сами атрибуты изображают в пределах прямоугольника, который определяет сущность.

Ключ сущности представляет собой не избыточный набор атрибутов этой сущности, причем значения атрибутов являются уникальными для каждого отдельного экземпляра данной сущности [38]. Не избыточность определяется тем, что при удалении некоторого атрибута из ключа теряется его уникальность. Сама же сущность может обладать несколькими различными ключами. Ключевые атрибуты обычно изображаются на ER диаграмме подчеркиванием.

Для хранения информации о поданных объявлениях, о продавцах, товарах и других данных была спроектирована БД методом «ER-диаграмм». Схема спроектированной БД представлена на рисунке 3.6.

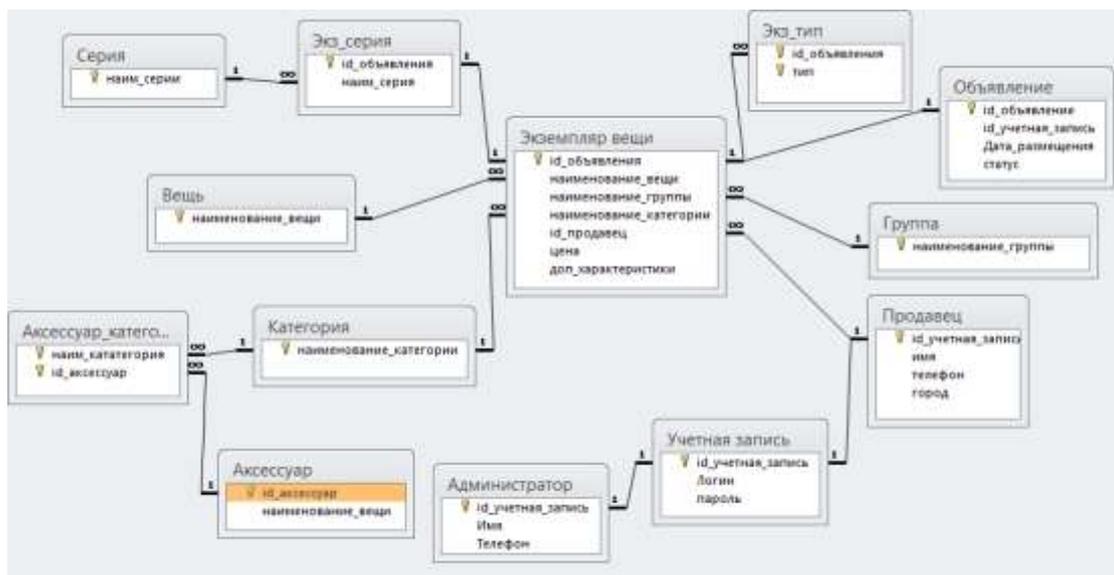


Рисунок 3.6 – База данных таблицы

3.3 Разработка интерфейса доски объявлений

Web-сайт электронной доски спроектирован таким образом, что позволяет просматривать страницы в любом разрешении пропорционально размеру экрана монитора. Наличие графической информации в виде картинок позволяют пользователям уверенно ориентироваться на сайте. В случае если у пользователя в настройках браузера отключена функция просмотра рисунков, то вместо рисунков на экране будет отображаться альтернативный текст, подсказывающий о наличии в том месте определенного рисунка.

Пользователь всегда будет делать ошибки, даже в отлично спроектированной программе, поэтому в ней должна быть предусмотрена защита от ошибок. В предлагаемом web-сайте контроль ошибок осуществляется на этапе отправки информации введенной пользователем в формы. Во время возникновения ошибочной ситуации, например, неправильно введенных данных в форму, пользователю ничего не сообщается. Все операции добавления информации в базу данных производятся только после проверки их корректности на сервере, и в случае если данные или действия корректны, пользователь получает подтверждение.

Внутренняя структура сайта – это система организации ссылок между страницами сайта. Внутренняя структура должна продумываться на первых этапах создания проекта до начала разработки дизайна.

Можно выделить три основных способа организации внутренней структуры сайта:

– Линейная структура (Рисунок 3.7). Web-страницы идут одна за другой, и пользователь должен просматривать их как слайд-шоу. Используется редко, поскольку очевидная простота данной структуры не позволяет реализовать проекты с большим объёмом информации и ставит пользователя в жёсткие рамки. Чаще всего данная структура используется при создании онлайн-презентаций.

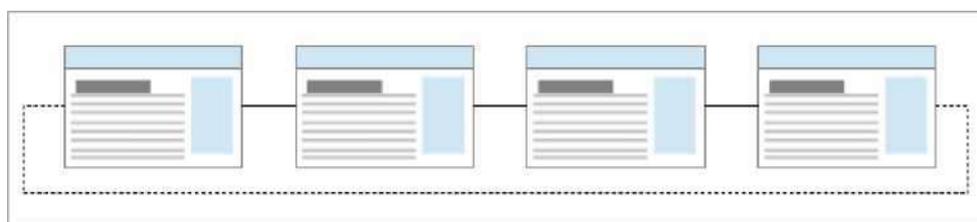


Рисунок 3.7 – Сайт с линейной структурой

– Решетчатая структура (Рисунок 3.8). Это та же линейная структура, в которой все страницы размещаются в различных ветках. У пользователя есть возможность перемещаться по ним не только вертикально (вверх-вниз) но и горизонтально (то есть между ветками на разных уровнях). Например, в интернет-магазине данная структура позволит просмотреть товары, объединённые по двум признакам сразу.

– Древовидная структура (Рисунок 3.9). Наиболее часто встречающаяся модель организации ресурса. Позволяет управлять «глубиной» просмотра страниц сайта. Пользователь при заходе на главную страницу оказывается перед выбором, куда идти дальше. После перехода в нужный раздел, он подбирает необходимый подраздел и т. п.

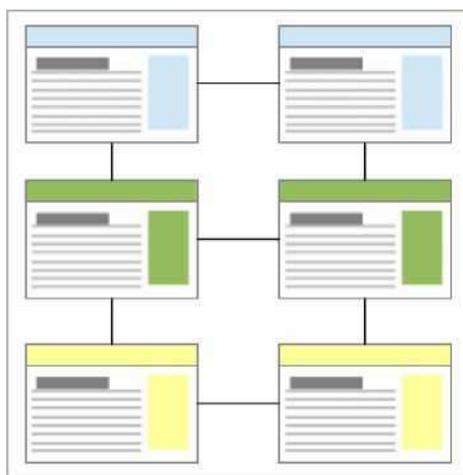


Рисунок 3.8 – Сайт с решётчатой структурой

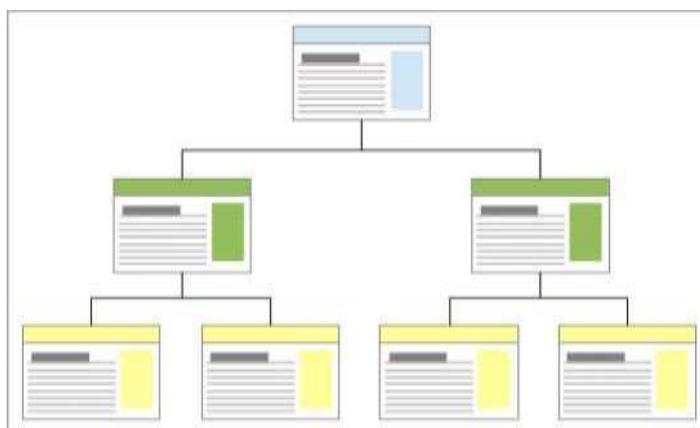


Рисунок 3.9 – Сайт с древовидной структурой

На практике приведённые выше структуры редко используются в исходном виде. Чаще всего для создания сайтов применяется комбинированная модель, сочетающая в себе особенности нескольких более простых структур. Комбинированная модель – это конструктор, где блоками являются необходимые разработчику базовые структуры.

Для создания электронной доски объявлений было решено выбрать, как базовую, древовидную структуру. Общая структура сайта с указанием основных связей между разделами представлена в приложении А.

Внешняя структура сайта – это некая блок-схема, которая определяет порядок размещения отдельных элементов сайта на его страницах. Разработчик сайта

определяет, где должен располагаться основной текст, меню сайта, новостной раздел, рекламные блоки, дополнительные материалы.

В типичную внешнюю схему сайта обычно входят: шапка (header), боковая панель (sidebar), которых может быть несколько, содержание или основная часть (index) и подвал (footer). В настоящее время в интернете всё чаще встречаются сайты без чёткой внешней структуры, но это, в основном, рекламные сайты, промо-сайты и сайты-визитки, скорее ориентированные на произведение внешнего эффекта, чем на донесение информации. Для большинства сайтов по-прежнему важно наличие разработанной внешней структуры, помогающей пользователям с удобством просматривать сайт.

Внешнюю схему сайта также называют каркасом или прототипом сайта. Прототип сайта – это эскизный проект, который создаётся перед изготовлением оригинал-макета сайта. В прототипе все важные элементы отображают в чёрно-белом цвете. Таким образом, каркасное моделирование позволяет сократить время, необходимое для разработки, фокусируя внимание на базовой функциональности.

Цель создания любого web-сайта – не только донести информацию до пользователя, но и произвести на него эстетическое впечатление. Разработка оформления сайта – это творческая задача, направленная на проявление художественных способностей разработчика. Именно дизайн сайта делает его особенным, выделяет в массе других web-сайтов в Интернете. В процессе создания необходимо руководствоваться двумя важными принципами:

- дизайн не должен мешать посетителю получать необходимую информацию;
- дизайн должен быть запоминающимся и привлекательным.

Дизайн сайта, безусловно, имеет не менее важное значение, чем текстовая информация. Посетитель, в первый раз зайдя на сайт, оценивает его в первую очередь с визуальной точки зрения, он делает выбор: остаться на web-странице или закрыть её, и больше на сайт не возвращаться. Оформление способно как помогать пользователю ориентироваться по сайту, так и отвлекать.

Для электронной доски объявлений одежды разработан вариант оформления в спокойных тонах.

Основным аргументом в пользу выбора данного варианта дизайна стало то, что основная идея сайта – простой внешне, но функциональный и красивый сайт, позволяющий удовлетворить основное желание посетителя – купить танцевальную одежду – достигнута при помощи использования стиля Flat и визуальных графических элементов. Основной акцент делается на изображения, чтобы посетитель не заикливался на рассматривании дизайна и не пробирался через сложную структуру.

Элементы просты и понятны, логотип соответствует тематике сайта, цветовая гамма оформления нацелена на основную целевую аудиторию.

3.4 Разработка программного кода

Поскольку электронная доска объявлений представляет собой совокупность модулей, то программирование электронной доски объявлений сводится к программированию модулей и установлению их взаимосвязей [2].

Вообще, для любой информационной системы, в том числе и электронной доски объявлений, следует разработать так называемую точку входа в приложение, через которую осуществляется вся дальнейшая маршрутизация.

В web-программировании такой точкой входа является файл `index.php`. Поскольку ключевой функцией для электронной доски объявлений является каталог объявлений, то в точке входа в приложении следует подключить файл каталога товаров.

```
<?php
header('Content-Type: text/html; charset= utf-8');
session_start();
require_once 'config.php';
require_once 'functions.php';
$categories = getCategories($mysql_link);
$types = get_type($mysql_link);
$user = checkUser($mysql_link);
if(!empty($user) and is_array($user)){
```

```

    $add_mess = privileges($mysql_link, $user['role_id'],
array('ADD_MESS'));
    }
    $action = clearData($mysql_link, $_GET['action']);
    if(!$action){
        $action = 'index';
    }
    $title = getTitle($mysql_link, $action, $user, $types,
$categories);
    $main_menu = getMainMenu($mysql_link, $user, $action);

    if(file_exists('actions/' . $action . '.inc.php')){
        include 'actions/' . $action . '.inc.php';
    }else{
        include 'actions/' . 'index.inc.php';
    }
    require_once THEME . 'index.tpl.php';
    unset($_SESSION['msg']);

```

Для соединения с БД требуется указать параметры подключения. Они расположены в файле config.php:

```

<?php
$config = array(
    'db' => array(
        'dsn' => 'pgsql:dbname=bulletin;host=localhost',
        'user' => 'user',
        'password' => 'password'
    ),
    'pagination' => 12,
);

```

Некоторые листинги программных модулей представлены в приложении В.

3.5 Пример использования доски объявлений

В рамках тестирования доски объявлений был занесен тестовый каталог объявлений о продаже одежды.

Для того чтобы выложить свой товар пользователь должен быть идентифицирован, это можно сделать 2-мя способами авторизоваться путем введения своего Логина и Пароля если пользователь уже есть в базе данных, или

же пройти регистрацию заполнив для этого все существующие поля и уже потом авторизоваться.

Таким образом, разработанная доска объявлений выполняет все базовые функции, присущие подобным системам.

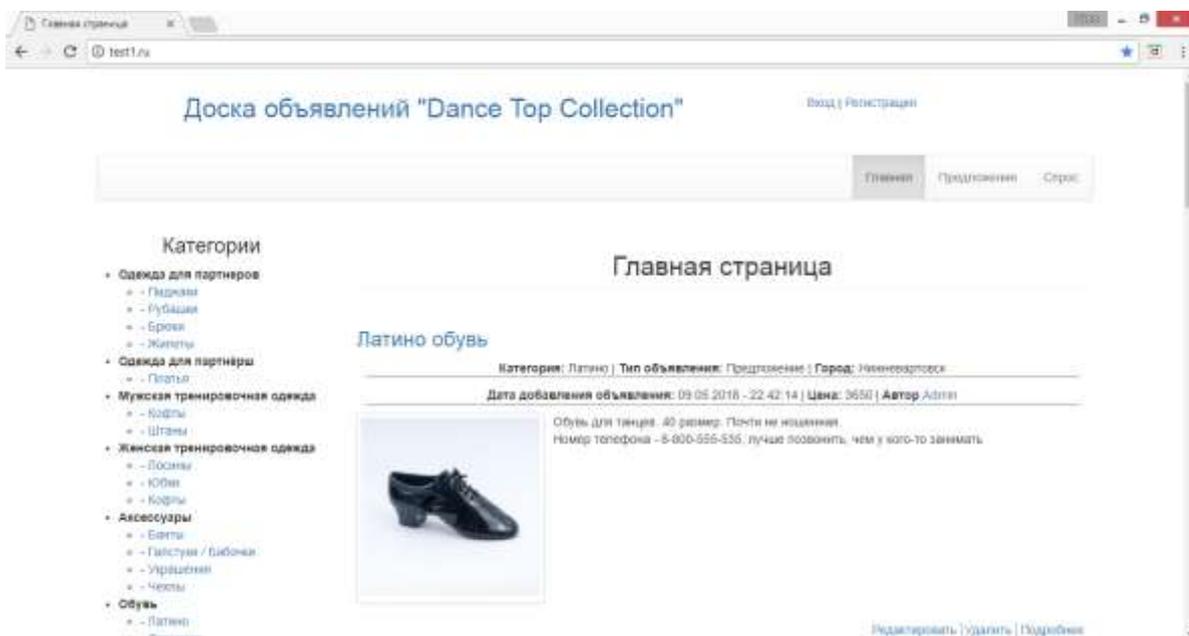


Рисунок 3.10 – Главная страница

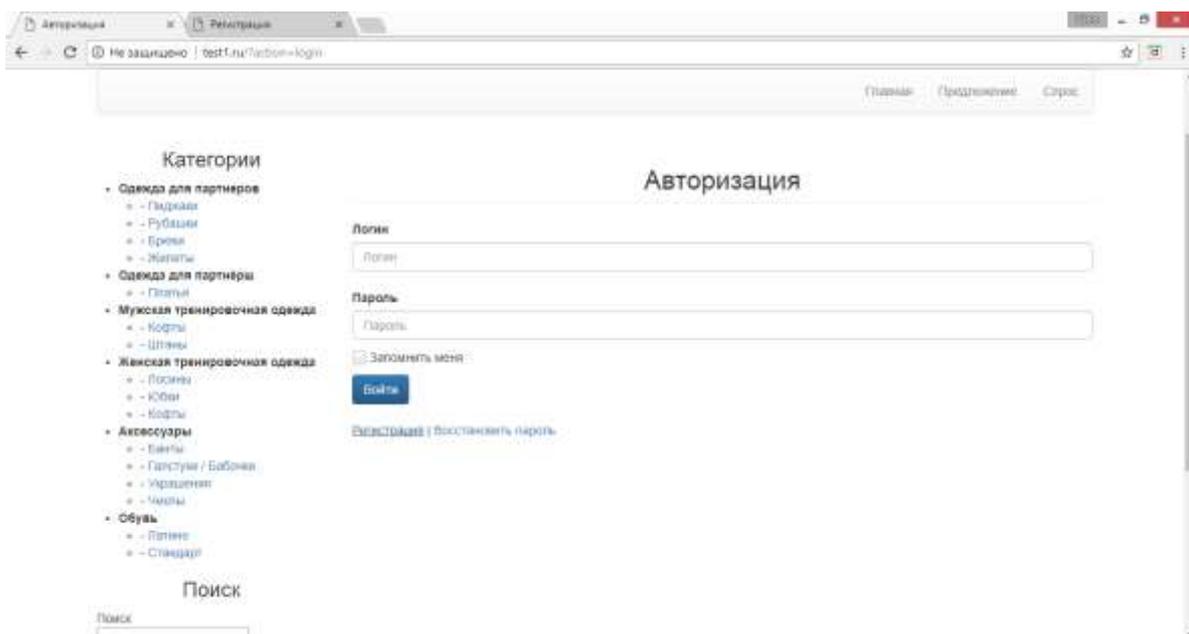


Рисунок 3.11 – Страница авторизации

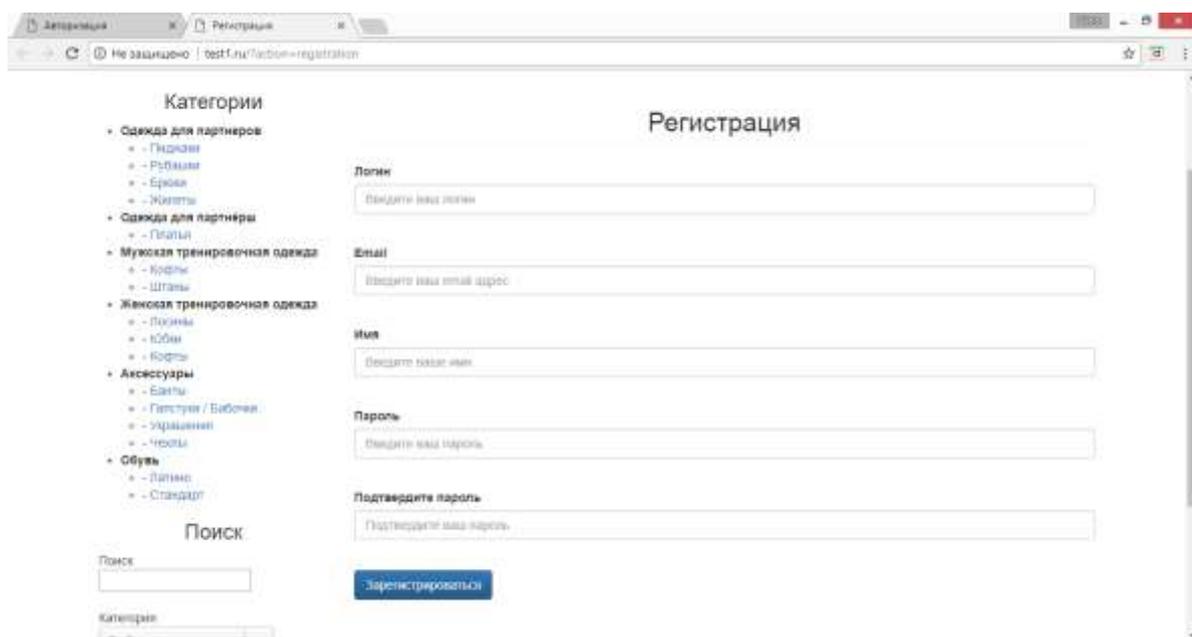


Рисунок 3.12 – Страница регистрации

Выводы по разделу три:

Без использования приложения разработанного в рамках выпускной квалификационной работы происходит нерациональная трата рабочего времени сотрудников.

Основной экономической эффективностью разработанной ИС являются следующие факторы:

- автоматизация обработки данных анкетирования;
- скорость обработки.

Необходимо подчеркнуть, что у программных продуктов практически отсутствует процесс физического старения и износа. Основные затраты приходятся на разработку и внедрение. Так как затраты на разработку данного продукта минимальны, а его экономическая выгодность доказана, программа может быть благополучно внедрена в филиале.

4 ОРГАНИЗАЦИОННО–ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

Расходы на создание Доски объявлений складываются из первоначальных расходов и постоянных во времени расходов, предназначенных для поддержания данного ресурса.

Структура расходов на создание и сопровождение интернет ресурса доска объявлений «Dance Top Collectoin»:

Первоначальные:

- разработка логотипа;
- дизайн сайта;
- услуги программиста;
- покупка домена;
- оплата хостинга;
- менеджер по работе с клиентами в рекламном агентстве.

Регулярные ежемесячные затраты:

- оплата доступа в интернет и хостинга;
- реклама площадки;
- администрирование;

4.1 Первоначальные затраты

Создание логотипа и фирменного стиля. Продуманный бренд, логотип, фирменный стиль и прочие внешние атрибуты это свидетельства надежности бизнеса, на которые обращают внимание как пользователи, так и деловые партнеры.

Основные составляющие концепции бренда:

- web-сайт;
- буклет;
- плакат;

- наружная реклама;
- мультимедийная презентация;
- все формы наружной рекламы, включая интернет рекламу.

Стоимость создания логотипа и фирменного стиля: от 8000 до 25 000 руб.

Внешний вид торговой площадки имеет огромное значение для привлечения новых и удержания старых клиентов. Шаблон доски объявлений, разработан с учетом требований целевой аудитории пользователей и заказчиком, сам по себе является эффективным инструментом продаж.

Домен и хостинг:

- домен – регистрация 550 руб., продление – от 500 руб.;
- хостинг Восток 3 – 600 рублей в месяц (можно найти дешевле, но в ущерб надежности, скорости работы и предоставляемой памяти);

Как правило, при покупке хостинга у официального представителя, предоставляется домен меньше рыночной стоимости, от чего можно сэкономить. Сэкономить также можно, если оплата производится сразу за год.

Зарплата сотрудникам. Многие стартапы на первых порах обходятся минимумом сотрудников. Фактически, готовый интернет-ресурс позволяет владельцу вообще отказаться от штата постоянных работников, передав решение рутинных задач в руки аутсорсинговых компаний или фрилансеров.

Средние расценки на услуги специалистов разного профиля:

- E-mail-маркетолог (7000 руб. в месяц).
- Интернет-маркетолог (от 11 000 руб. в месяц).
- Администратор сайта (4000–6000 в месяц).
- Специалист по стратегическому планированию рекламы (8000 руб. в месяц).
- Менеджер по работе с клиентами в рекламном агентстве.

Сумма затрат на оплату труда сильно колеблется в зависимости от конкретного проекта и профессионализма работников. Следует также учитывать, что постоянную зарплату будет получать лишь администратор сайта и менеджер

по работе с клиентами в рекламном агентстве, к остальным, можно обращаться лишь в тех случаях, когда динамика посетителей сайта снижается.

Таблица 4.1 – Смета первоначальных затрат открытия доски объявлений

Расходные статьи	Первоначальные затраты, руб.
Зарплата персоналу	26000
Покупка домена	550
Оплата интернет-услуг (доступ в Интернет, хостинг и т.д.)	1500
Реклама площадки (контекстная реклама, покупка ссылок)	8000
Менеджер по работе с клиентами в рекламном агентстве.	5000
Администрирование сайта	5000
Итого:	46050

4.2 Регулярные ежемесячные расходы

Регулярные ежемесячные затраты. Создание доски объявлений – это лишь вершина айсберга. Далее предстоит позаботиться о техническом обслуживании площадки и других расходах, которые обеспечивают ее бесперебойную работу. Также расчёт затрат на электроэнергию.

За один час работы за компьютером расходуется 0,4 кВт/ч электроэнергии. В среднем время работы за компьютером составляет 5ч. Следовательно: $5 \times 0,4 = 2 \text{ кВт/ч}$.

За 22 рабочих дня, по 5 часов работы за компьютером: $22 \times 5 = 110$ часов.

За 4 месяца работы расход электроэнергии в размере: $110 \cdot 4 \cdot 2 = 880 \text{ кВт/ч}$.

Следует отметить, что стоимость электроэнергии в Нижневартовске определяется по тарифу и поэтому будет взято максимальное значение по городу, а это 4,68 рублей за 1 кВт/ч, следовательно, затрачиваемая сумма:

$\Sigma = 880 \cdot 4,68 = 4118,4$ рублей.

Примерный список ежемесячных затрат:

– оплата доступа в Интернет – от 700 руб. в месяц;

– оплата хостинга и домена – до 1200 руб. в месяц, при оплате на год вперед предоставляются скидки;

– реклама площадки – здесь все сугубо индивидуально, на начальном этапе владельцы небольших магазинов могут уложиться в 3000–8000 руб. в месяц (покупка ссылок + небольшая компания в Яндекс. Директ или Google. Adsense);

– администрирование сайта – от 4000 руб. в месяц;

– оплата менеджеру по работе с клиентами в рекламном агентстве – от 5000 руб. в месяц.

Ниже приведена смета для открытия доски объявлений.

Таблица 4.2 – Смета для открытия доски объявлений

Расходные статьи	Ежемесячные затраты, руб.
Оплата интернет-услуг (доступ в Интернет, хостинг и т.д.)	1900
Реклама площадки (контекстная реклама, покупка ссылок)	8000
Администрирование сайта	5000
Оплата электроэнергии	1030
Итого	15930

4.3 Прибыльность доски объявлений

При определении прибыльности доски объявлений, зададим следующие данные:

– сумма первичной «закупки» рекламы: 35000 руб.;

– цена и объем рекламы на сайте.

Вышеописанные данные являются истинными практически для 60% площадок электронной торговли. Прибыль сайта исходит из нескольких факторов, это количество рекламы и количество уникальных посетителей на сайте. Цена рекламы на сайте варьируется от числа посетителей на сайте, то есть, чем больше посетителей, тем дороже реклама. Google Analytics и Яндекс Метрика в состоянии

проверять количество посетителей, за день, неделю, месяц и тд. Следует также отметить, что рекламы бывают разные это:

– Контекстные – минимальная цена такой рекламы на сайт до 1000 человек это 6000 руб./мес.;

– Размещение статей и обзоров за деньги – 300 рублей за размещение 1 статьи за месяц в среднем размещается 5–10 рекламных постов. Минимальный заработок 1500 рублей в месяц;

– Баннерная реклама. Цена баннерной рекламы заключается лишь в просмотре данного баннера. СРМ – это рекламная модель, в которой цена устанавливается за тысячу показов баннера рекламодателя. Минимальный заработок от такой рекламы составляет: 1000 просмотров каждый день = 30000 просмотров в месяц $30\ 000 \times 50 = 1500$ рублей в месяц. Цена за баннерную рекламу жестко привязана к тематике Вашего сайта. Также напрямую на цену влияет размер баннера. Чем он больше – тем он дороже.

Таким образом, учитывая первоначальные расходы на создание доски объявлений, ежемесячные расходы и прибыль, можно построить диаграмму. Прибыль доски объявлений, как правило, в первых месяцах ниже средней прибыли. На это имеются вполне обоснованные причины, связанные с постепенным повышением индекса цитирования в интернете и плавным увеличением посещаемости, а так же постепенным ростом доверия посетителей.

Диаграмма прибыли доски объявлений (зеленая линия).

На диаграмме показано, что прибыль положительная уже через четвертый месяц и составляет 23 000 рублей. К моменту времени доска объявлений окупит все расходы на создание и сопровождение пройдет 11 месяцев. Дальнейший прирост прибыли зависит от рекламы и от количества посетителей. В дальнейшем может понадобиться создание мобильного приложения, дабы повысить количество посетителей.

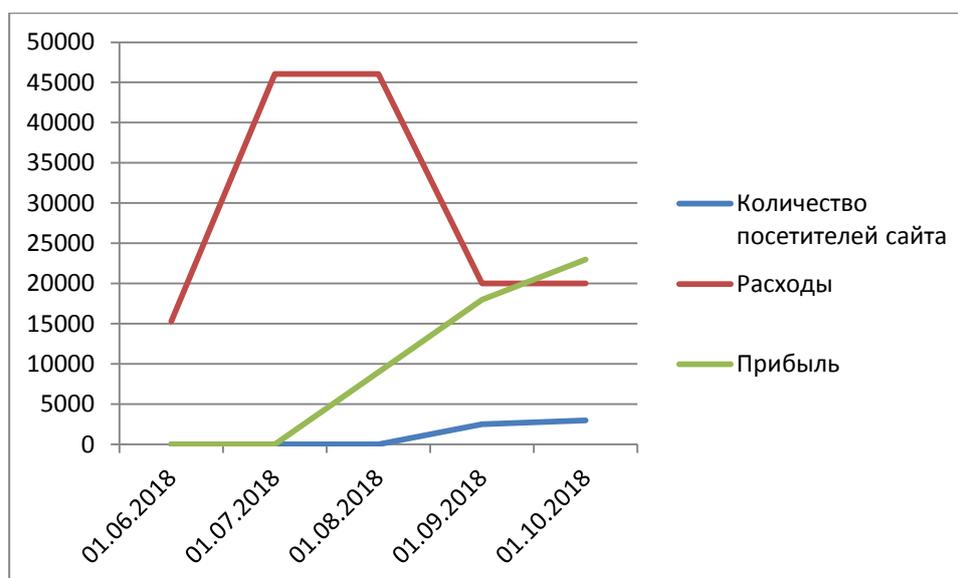


Рисунок 4.1 – Диаграмма роста прибыли доски объявлений

Выводы по разделу четыре:

Создание доски объявлений обязательно включает в себя ряд статей расходов. Большая часть расходов идет на первоначальные вложения и рекламу, но с учетом невысоких дальнейших расходов, доска объявлений окупается через 11 месяцев.

5 БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

Большую роль при исполнении людьми своих трудовых обязанностей играет обеспечение безопасности. Именно по этой причине в 1989 г. была создана наука о безопасности жизнедеятельности человека, которая развивается и по сей день.

На рабочем месте должны быть предусмотрены меры защиты от возможного воздействия небезопасных и вредоносных факторов производства. Для этого существуют предельные значения этих факторов, которые оговорены техническими, правовыми и санитарно-техническими нормами и уровни которых не должны превышать заданные. Данные нормативные документы обязывают к созданию на рабочем месте условий труда, при которых влияние небезопасных и вредоносных факторов на работников либо окончательно устранено, либо находится в пределах допустимого [20].

Данный раздел выпускной квалификационной работы посвящен рассмотрению следующих вопросов:

- характеристика труда программиста;
- требования к производственным помещениям (освещение, микроклимат, шум и вибрация);
- эргономические требования к рабочему месту;
- режим труда и отдыха при работе с компьютером;

5.1 Характеристика условий труда программиста

Современный комплекс организационных мероприятий и технических средств защиты показывает, что в деле устранения воздействия на работающих людей опасных и вредных производственных факторов при проектировании алгоритмов и разработке программного обеспечения возможно добиться значительных успехов.

Производственный фактор является опасным, если его воздействие на работающего человека в определенных условиях приводит к травме или другому внезапному резкому ухудшению здоровья.

Производственный фактор считается вредным, если он приводит к заболеванию или снижению трудоспособности. В зависимости от уровня и продолжительности воздействия вредный производственный фактор может стать опасным. Существуют четыре группы опасных и вредных факторов: физические, химические, биологические и психофизические.

На сегодняшний день состояние условий труда программиста и его безопасности ещё не удовлетворяют современным требованиям. Он сталкивается с воздействием физически опасных и вредных производственных факторов, таких как повышенный уровень шума, повышенная температура внешней среды, отсутствие или недостаточная освещенность рабочей зоны, электрический ток, статическое электричество и др.

Также на программиста может оказываться влияние таких психофизических факторов, как умственное перенапряжение, перенапряжение зрительных анализаторов, перенапряжение слуховых анализаторов, монотонность труда, эмоциональные перегрузки.

Воздействие указанных неблагоприятных факторов приводит к снижению работоспособности, которое вызвано развивающимся утомлением. Появление и развитие утомления связано с изменениями, возникающими во время работы в центральной нервной системе, с тормозными процессами в коре головного мозга.

Помимо снижения производительности труда высокие уровни шума приводят к ухудшению слуха. Длительное нахождение человека в зоне комбинированного воздействия различных неблагоприятных факторов может привести к профессиональному заболеванию.

Проведенные исследования говорят о том, что в основном несчастные случаи происходят от воздействия физически опасных производственных факторов при

выполнении сотрудниками несвойственных им работ. На втором месте случаи, связанные с воздействием электрического тока.

Опасность поражения человека током возникает:

- при снижении сопротивления изоляции;
- при замыкании фазы на корпус;
- при нарушении техники безопасности.

В рабочем помещении программиста используется напряжение питания 220В в сети переменного тока с частотой 50 Гц с заземлённой нейтралью, опасное для человека.

Поэтому для защиты работников от опасности прикосновения к токоведущим частям предусматривается зануление, а также изоляция световой и силовой электропроводки. Сопротивление изоляции между любым проводом и землёй согласно ПУЭ – не менее 0,5 МОм.

5.2 Требования к производственным помещениям

5.2.1 Освещение

В компьютерных залах должно быть естественное и искусственное освещение. Естественное освещение обеспечивается через оконные проёмы с коэффициентом естественного освещения КЕО не ниже 1,2% в зонах с устойчивым снежным покровом и не ниже 1,5% на остальной территории. Световой поток из оконного проёма должен падать на рабочее место оператора с левой стороны.

Искусственное освещение в помещениях эксплуатации компьютеров должно осуществляться системой общего равномерного освещения.

Освещённость на поверхности стола в зоне размещения документа должна быть 300–500 лк. Допускается установка светильников местного освещения для подсветки документов. Местное освещение не должно создавать бликов на

поверхности экрана и увеличивать освещенность экрана более 300 лк. Прямую блескость от источников освещения следует ограничить. Яркость светящихся поверхностей (окна, светильники), находящихся в поле зрения, должна быть не более 200 кд/м² [21].

Отраженная блескость на рабочих поверхностях ограничивается за счёт правильного выбора светильника и расположения рабочих мест по отношению к естественному источнику света. Яркость бликов на экране монитора не должна превышать 40 кд/м². Показатель ослеплённости для источников общего искусственного освещения в помещениях должен быть не более 20, показатель дискомфорта в административно-общественных помещениях не более 40. Соотношение яркости между рабочими поверхностями не должно превышать 3:1 – 5:1, а между рабочими поверхностями и поверхностями стен и оборудования 10:1.

Для искусственного освещения помещений с персональными компьютерами следует применять светильники типа ЛПО36 с зеркализированными решётками, укомплектованные высокочастотными пускорегулирующими аппаратами. Допускается применять светильники прямого света, преимущественно отраженного света типа ЛПО13, ЛПО5, ЛСО4, ЛПО34, ЛПО31 с люминисцентными лампами типа ЛБ. Допускается применение светильников местного освещения с лампами накаливания. Светильники должны располагаться в виде сплошных или прерывистых линий сбоку от рабочих мест параллельно линии зрения пользователя при разном расположении компьютеров. При периметральном расположении — линии светильников должны располагаться локализованно над рабочим столом ближе к его переднему краю, обращённому к оператору. Защитный угол светильников должен быть не менее 40 градусов. Светильники местного освещения должны иметь непросвечивающийся отражатель с защитным углом не менее 40 градусов.

Для обеспечения нормативных значений освещенности в помещениях следует проводить чистку стекол оконных проёмов и светильников не реже двух раз в год и проводить своевременную замену перегоревших ламп.

5.2.2 Параметры микроклимата

В помещении, в котором установлен компьютер, должен поддерживаться определенный микроклимат – влажность, запыленность, температура воздуха и т. д.

С 30 июня 2003 года действуют обязательные санитарно–эпидемиологические правила и нормативы – СанПиН 2.2.2/2.4.1340–03 «Гигиенические требования к персональным электронно–вычислительным машинам и организации работы». На сегодняшний день это основной нормативный документ по безопасной работе на компьютере. В этом документе наряду с другими положениями регламентируются требования к микроклимату помещения.

Для помещения с компьютером существуют определенные требования к температуре, влажности и наличию пыли. Температура должна находиться на уровне 21–25 °С, относительная влажность – 40–60 %, уровень аэроионов – от 400–600 до 50 000 (оптимальный – 1500–5000).

Это оптимальные условия для обеспечения максимально комфортного теплового баланса температуры тела человека и его терморегуляции. Если температура выше нормы, кровеносные сосуды расширяются и теплоотдача в окружающую среду возрастает. При понижении температуры кровеносные сосуды соответственно сужаются приток крови к телу замедляется и теплоотдача уменьшается.

На терморегуляцию организма влияет также влажность воздуха. Слишком высокая влажность (более 85 %) затрудняет терморегуляцию, а слишком низкая (менее 20 %) вызывает пересыхание слизистых, причем не только дыхательных путей, но и глаз.

Не менее важна оптимальная влажность в помещении: чем она выше, тем слабее влияние электростатических и электромагнитных полей, уровень излучения которых в помещении, где установлен компьютер, всегда повышен.

Принципиальным фактором в микроклимате помещения с компьютером является уровень пыли. Человеческий организм плохо адаптирован к условиям повышенной запыленности. Квартирно–офисная пыль сильно отличается от природной. Офисная пыль может содержать частицы мебельных тканей, клея, строительных материалов, частицы кожи человека и домашних животных, в том числе грызунов, споры микроскопических плесневых и дрожжевых грибов, различные виды клещей, волокна хлопка, льна, бумаги, бактерии и вирусы.

Такая пыль, попав в легкие, может вызвать различные заболевания – от аллергических реакций до хронических катаров верхних дыхательных путей, ларингита, хронического насморка, трахеита и даже хронического бронхита.

Учитывая высокий уровень электромагнитного излучения в помещении с компьютером, пыль не оседает на поверхностях. Она электризуется от экрана монитора и висит в воздухе, поэтому гораздо проще попадает в легкие и на слизистые человека. По этой причине чистоте помещения, где есть компьютер, следует уделять особенное внимание. Влажная уборка в таком помещении должна проводиться не реже трёх раз в неделю в офисе и не реже раза в неделю дома.

Кроме этого, помещение, где стоит компьютер, должно хорошо проветриваться. Чистый воздух – это лучший источник легких ионов, который не заменит ни один ионизатор.

5.2.3 Шум и вибрация

Уровни шума на рабочих местах пользователей персональных компьютеров не должны превышать значений, установленных СанПиН 2.2.4/2.1.8.562-96 и составляют не более 50 дБА. На рабочих местах в помещениях для размещения шумных агрегатов уровень шума не должен превышать 75 дБА, а уровень

вибрации в помещениях допустимых значений по СН 2.2.4/2.1.8.566-96 категория 3, тип «В».

Снизить уровень шума в помещениях можно использованием звукопоглощающих материалов с максимальными коэффициентами звукопоглощения в области частот 63–8000 Гц для отделки стен и потолка помещений. Дополнительный звукопоглощающий эффект создают однотонные занавески из плотной ткани, повешенные в складку на расстоянии 15–20 см от ограждения. Ширина занавески должна быть в 2 раза больше ширины окна [22].

В таблице 5.1 указаны предельные уровни звука в зависимости от категории тяжести и напряженности труда, являющиеся безопасными в отношении сохранения здоровья и работоспособности.

Таблица 5.1 – Предельные уровни звука, дБ, на рабочих местах

Категория напряженности труда	Категория тяжести труда			
	I. Легкая	II. Средняя	III. Тяжелая	IV. Очень тяжелая
I. Мало напряженный	80	80	75	75
II. Умеренно напряженный	70	70	65	65
III. Напряженный	60	60	-	-
IV. Очень напряженный	50	50	-	-

5.3 Эргономические требования к рабочему месту

Рабочие места с персональными компьютерами по отношению к световым проёмам должны располагаться так, чтобы естественный свет падал сбоку, желательно слева.

Схемы размещения рабочих мест с персональными компьютерами должны учитывать расстояния между рабочими столами с мониторами: расстояние между боковыми поверхностями мониторов не менее 1,2 м, а расстояние между экраном монитора и тыльной частью другого монитора не менее 2,0 м.

Рабочий стол может быть любой конструкции, отвечающей современным требованиям эргономики и позволяющей удобно разместить на рабочей

поверхности оборудование с учетом его количества, размеров и характера выполняемой работы. Целесообразно применение столов, имеющих отдельную от основной столешницы специальную рабочую поверхность для размещения клавиатуры. Используются рабочие столы с регулируемой и нерегулируемой высотой рабочей поверхности. При отсутствии регулировки высота стола должна быть в пределах от 680 до 800 мм.

Глубина рабочей поверхности стола должна составлять 800 мм (допускаемая не менее 600 мм), ширина – соответственно 1 600 мм и 1 200 мм. Рабочая поверхность стола не должна иметь острых углов и краев, иметь матовую или полуматовую фактуру.

Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной — не менее 500 мм, глубиной на уровне колен — не менее 450 мм и на уровне вытянутых ног — не менее 650 мм [23].

Быстрое и точное считывание информации обеспечивается при расположении плоскости экрана ниже уровня глаз пользователя, предпочтительно перпендикулярно к нормальной линии взгляда (нормальная линия взгляда 15 градусов вниз от горизонтали).

Клавиатура должна располагаться на поверхности стола на расстоянии 100–300 мм от края, обращенного к пользователю.

Для удобства считывания информации с документов применяются подвижные подставки (пюпитры), размеры которых по длине и ширине соответствуют размерам устанавливаемых на них документов. Пюпитр размещается в одной плоскости и на одной высоте с экраном.

Для обеспечения физиологически рациональной рабочей позы, создания условий для её изменения в течение рабочего дня применяются подъемно-поворотные рабочие стулья с сиденьем и спинкой, регулируемые по высоте и углам наклона, а также расстоянию спинки от переднего края сидения.

Конструкция стула должна обеспечивать:

– ширину и глубину поверхности сиденья не менее 400 мм;

- поверхность сиденья с закругленным передним краем;
- регулировку высоты поверхности сиденья в пределах 40–550 мм и углом наклона вперед до 15 градусов и назад до 5 градусов.;
- высоту опорной поверхности спинки 300 ± 20 мм, ширину — не менее 380 мм и радиус кривизны горизонтальной плоскости 400 мм;
- угол наклона спинки в вертикальной плоскости в пределах 0 ± 30 градусов;
- регулировку расстояния спинки от переднего края сидения в пределах 260–400 мм;
- стационарные или съемные подлокотники длиной не менее 250 мм и шириной 50–70 мм;
- регулировку подлокотников по высоте над сиденьем в пределах 230 ± 30 мм и внутреннего расстояния между подлокотниками в пределах 350–500 мм.;
- поверхность сиденья, спинки и подлокотников должна быть полумягкой, с нескользящим неэлектризующимся, воздухопроницаемым покрытием, легко очищаемым от загрязнения.

Рабочее место должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20 град. Поверхность подставки должна быть рифлёной и иметь по переднему краю бортик высотой 10 мм.

5.4. Режим труда и отдыха при работе с компьютером

Режим труда и отдыха предусматривает соблюдение определенной длительности непрерывной работы на персональном компьютере и перерывов, регламентированных с учётом продолжительности рабочей смены, видов и категории трудовой деятельности.

Виды трудовой деятельности на персональном компьютере разделяются на 3 группы:

– группа А – работа по считыванию информации с экрана с предварительным запросом;

– группа Б – работа по вводу информации;

– группа В – творческая работа в режиме диалога с ПК.

Если в течение рабочей смены пользователь выполняет разные виды работ, то его деятельность относят к той группе работ, на выполнение которой тратится не менее 50% времени рабочей смены.

Категории тяжести и напряжённости работы на ПК определяются уровнем нагрузки за рабочую смену:

– для группы А — по суммарному числу считываемых знаков;

– для группы Б — по суммарному числу считываемых или вводимых знаков;

– для группы В — по суммарному времени непосредственной работы на ПК.

В таблице 5.2 приведены категории тяжести и напряженности работ в зависимости от уровня нагрузки за рабочую смену.

Таблица 5.2 – Виды категорий трудовой деятельности с ПК

Категория работы по тяжести и напряжённости	Уровень нагрузки за рабочую смену при видах работы на ПК		
	Группа А Количество знаков	Группа Б Количество знаков	Группа В Время работы, ч
I	До 20000	До 15000	До 2
II	До 40000	До 30000	До 4
III	До 60000	До 40000	До 6

Количество и длительность регламентированных перерывов, их распределение в течение рабочей смены устанавливается в зависимости от категории работ на ПК и продолжительности рабочей смены.

При 8-часовой рабочей смене и работе на ПК регламентированные перерывы следует устанавливать:

– для первой категории работ через 2 часа от начала смены и через 2 часа после обеденного перерыва продолжительностью 15 минут каждый;

– для второй категории работ — через 2 часа от начала рабочей смены и через 1,5–2,0 часа после обеденного перерыва продолжительностью 15 минут каждый или продолжительностью 10 минут через каждый час работы;

– для третьей категории работ — через 1,5– 2,0 часа от начала рабочей смены и через 1,5–2,0 часа после обеденного перерыва продолжительностью 20 минут каждый или продолжительностью 15 минут через каждый час работы.

При 12-часовой рабочей смене регламентированные перерывы должны устанавливаться в первые 8 часов работы аналогично перерывам при 8-часовой рабочей смене, а в течение последних 4 часов работы, независимо от категории и вида работ, каждый час продолжительностью 15 минут.

Продолжительность непрерывной работы на ПК без регламентированного перерыва не должна превышать 2 часа.

При работе на ПК в ночную смену продолжительность регламентированных перерывов увеличивается на 60 минут независимо от категории и вида трудовой деятельности.

Эффективными являются нерегламентированные перерывы (микропаузы) длительностью 1–3 минуты.

Регламентированные перерывы и микропаузы целесообразно использовать для выполнения комплекса упражнений и гимнастики для глаз, пальцев рук, а также массажа. Комплексы упражнений целесообразно менять через 2–3 недели.

Пользователям ПК, выполняющим работу с высоким уровнем напряженности, показана психологическая разгрузка во время регламентированных перерывов и в конце рабочего дня в специально оборудованных помещениях (комнатах психологической разгрузки).

Выводы по разделу четыре:

В данном разделе были рассмотрены проблемы безопасности жизнедеятельности, связанные с работой за компьютером. Проведено исследование вредоносных условий, оказывающих воздействие на органы зрения пользователя ЭВМ. Сформированы общие требования к помещению и освещению. Проведены анализы шума, освещения и пожарной безопасности на рабочем месте пользователя ЭВМ. В связи с этим, были изучены различные нормы, сертификаты и правила; определялись пути решения проблем, для сокращения пагубного влияния на человека и обеспечения безопасных условий труда.

ЗАКЛЮЧЕНИЕ

В современности функционирование любой электронной доски объявлений трудно представить себе с отсутствием web-сайта или мобильного приложения. Разработка динамического web-сайта это длительный и очень интересный процесс.

В ходе разработки электронной доски объявлений, можно выделить несколько основных этапов. На первом этапе была собрана статистика, обоснована необходимость в этом сайте и определена целевая аудитория.

Вторым важным этапом стало проектирование интерфейса и структуры электронной доски объявлений, путём создания эскизов. А также проведён анализ схожих сервисов, поиск их положительных мест и недостатков. На этапе проектирования интерфейса были продуманы необходимые элементы сайта, их расположение и удобство пользования. Проектирование структуры сайта и базы данных, позволило понять, каким образом сайт будет работать и функционировать.

На следующем этапе были созданы макеты и шаблоны для дальнейшей верстки сайта и создания мобильного приложения. Макеты позволили наглядно оценить дизайн и юзабилити разрабатываемого электронной доски объявлений.

Четвертый этап заключался в непосредственной верстке web-страниц, используя созданные шаблоны. Один из наиболее сложных этапов, необходимо было учесть большое количество различных устройств на которых может быть запущен данный сервис и правильно запрограммировать интерфейс, чтобы на любом устройстве было удобно с ним взаимодействовать.

В заключительном этапе программирования электронной доски объявлений, были выбраны средства разработки и реализации функционала сайта, который был описан во введении. На этом же этапе непосредственно этот функционал реализовывался.

На последнем этапе создания электронной доски объявлений я познакомился с новым для себя языком программирования PHP, взаимодействие с базой данных, а также на этапе верстки web-страниц осваивал новые модули CSS4, их реализацией в различных браузерах.

В экономическом разделе были определены затраты, необходимые при производстве программного продукта и при его использовании, определены показатели экономической эффективности.

В разделе безопасности жизнедеятельности были описаны мероприятия, обеспечивающие безопасное и комфортное исполнение программистом своих трудовых обязанностей.

Таким образом, задачи выпускной квалификационной работы решены и цель достигнута.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Аллан, А. Программирование для мобильных устройств на iOS [Текст] / А. Аллан – Санкт-Петербург: Питер, 2013. – 416 с
- 2 Брайан, Х. Идеальный программист. Android. Программирование для профессионалов [Текст] / Х. Брайан, Б. Филлипс – Санкт-Петербург: Питер, 2016. – 640 с.
- 3 Вигерс, К. И. Разработка требований к программному обеспечению [Текст] / К. И. Вигерс, Д. Битти – Санкт-Петербург: БХВ-Петербург, 2015. – 784 с.
- 4 Голицына, О. Информационные системы. Учебное пособие [Текст] / О. Голицына, Н. Максимов, И. Попов – Москва: Форум, 2014. – 448 с.
- 5 Котеров, Д. PHP5 [Текст] / Д. Котеров, А. Костарев – Санкт-Петербург: БХВ-Петербург, 2016. – 1104 с.
- 6 Маклафлин, Б. Объектно-ориентированный анализ и проектирование [Текст] / Б. Маклафлин – Санкт-Петербург: Питер, 2013. – 608 с.
- 7 Мартин, Р. К. Чистый код: создание, анализ и рефакторинг. Библиотека программиста [Текст] / Р. К. Мартин – Санкт-Петербург: Питер, 2016. – 464 с.
- 8 Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript [Текст] / Р. Никсон – Санкт-Петербург: Питер, 2016. – 768 с.
- 9 Подборка документации и справочников по PHP [Электронный ресурс]. — Режим доступа: <http://www.php.ru> – Загл. с экрана. [дата обращения: 25.11.2017].
- 10 Потопахин, В. Искусство алгоритмизации [Текст] / В. Потопахин – Москва: ДМК Пресс, 2014. 320 с.: ил.
- 11 Прохоренок, Н. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера [Текст] / Н. Прохоренок – Санкт-Петербург: БХВ – Петербург, 2015. – 768 с.
- 12 Разбираемся в трендах веб-дизайна. [Электронный ресурс] – Режим доступа: <https://habrahabr.m/post/255287> – Загл. с экрана. [дата обращения 05.09.2016]

13 Роберт, М. Идеальный программист. Как стать профессионалом разработки ПО [Текст] / М. Роберт – Санкт-Петербург: Питер, 2012. – 278 с.

14 Руководство по PHP. [Электронный ресурс] – Режим доступа: <https://metanit.com/web/php> – Загл. с экрана. (дата обращения 08.12.2017)

15 Скляр, Д. Идеальный программист. Как стать профессионалом разработки ПО [Текст] / Д.Скляр, А.Трахтенберг – Санкт-Петербург: Питер, 2012. – 278 с.

16 Справочное руководство по языку Lua 5.1. [Электронный ресурс]. - Режим доступа: <http://www.lua.ru/doc> – Загл. с экрана. [дата обращения 10.01.2018].

17 Брауде, Э. Технология разработки программного обеспечения [Текст] / Э.Брауде – Санкт-Петербург, 2004. – 656 с.

18 TIOBE Index for May 2017 [Электронный ресурс]. - Режим доступа: <https://www.tiobe.com/tiobe-index> – Загл. с экрана. [дата обращения 15.04.2018]

19 PYPL Popularity of Programming Language [Электронный ресурс]. - Режим доступа: <http://pypl.github.io/PYPL.html> – Загл. с экрана. [дата обращения 16.04.2018]

20 Бабокин, И.А. Причины и обстоятельства нарушений правил безопасности в процессе труда и меры профилактики / И.А. Бабонкин – Безопасность труда в промышленности.- 2003.- № 3.

21 Самгин, Э.Б. Освещение рабочих мест / Э.Б. Самгин – М.: МИРЭА, 2009 г. – 27 с.

22 ГОСТ 12.1.003 – 83. Шум. Общие требования безопасности. [Электронный ресурс] – Режим доступа: <http://docs.cntd.ru/document/5200291> - Загл. с экрана. [Дата обращения – 02.05.2018 г.]

23 ГОСТ Р 50923-96. Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения (утв. Постановлением Госстандарта РФ от 10.07.96 N 451).

Методические рекомендации по подготовке и оформлению выпускной квалификационной работы (проекта) для технических направлений подготовки

09.03.01 Информатика и вычислительная техника, 09.03.04 Программная инженерия, 12.03.01 Приборостроение, 23.03.01 Технология транспортных процессов / сост. Л.Н.Буйлушкина. - Нижневартовск, 2017. - 35с.

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А. СВИДЕТЕЛЬСТВО О ПУБЛИКАЦИИ



Рисунок А.1 – Свидетельство о публикации

ПРИЛОЖЕНИЕ Б. ЛИСТИНГ

Программная модель управления объявлением

```
<?php
require_once(__DIR__ . '/../helpers/PDOConnection.php');
require_once(__DIR__ . '/../helpers/Validator.php');
class AdModel
{
    private static $db;
    private static $fields = array(
        'title', 'category', 'short_description', 'description',
        'image', 'phone', 'author',
    );
    private static $validationRules = array(
        'title' => [
            'string' => ['length_max' => 50],
        ],
        'category' => [
            'integer',
        ],
        'short_description' => [
            'string' => ['length_max' => 50],
        ],
        'description' => [
            'string' => ['length_max' => 255],
        ],
        'image' => [
        'file' => ['mime-type' => 'image/jpeg', 'size' => 1048576],
        ],
        'phone' => [
            'string' => ['length_strict' => 12],
        ],
        'author' => [
            'integer',
        ],
    );
    /**
     * @param $id
     * @return mixed
     */
    public static function getOneAd($id)
    {
        self::$db = PDOConnection::getInstance();
        self::$db->getConnection();
        self::$db->prepareStatement("SELECT ads.*, categories.name,
users.firstname, users.lastname
FROM ads LEFT JOIN categories ON ads.category=categories.id
LEFT JOIN users on ads.author=users.id
WHERE ads.id = :id");
```

```

self::$db->executeStatement(array(':id' => (int) $id));
    return self::$db->fetchOne();
}
public static function getManyAds($filter=false, $offset=0)
{
    if(!file_exists(__DIR__ . '/../config.php'))
    {
        return false;
    }
    include(__DIR__ . '/../config.php');

    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    if ($filter)
    {
        self::$db->prepareStatement("SELECT ads.*,
categories.name, users.firstname, users.lastname
FROM ads LEFT JOIN categories
ON ads.category=categories.id
LEFT JOIN users on
ads.author=users.id
WHERE ads.category = :filter
ORDER BY ads.created_at DESC LIMIT :limit OFFSET :offset");
        self::$db->executeStatement(array(':filter' => (int)
$filter, ':limit' => $config['pagination'], ':offset' => $offset));
    }
    else
    {
self::$db->prepareStatement("SELECT ads.*, categories.name,
users.firstname, users.lastname
FROM ads LEFT JOIN categories ON ads.category=categories.id
LEFT JOIN users on ads.author=users.id ORDER BY ads.created_at DESC
LIMIT :limit OFFSET :offset");
self::$db->executeStatement(array(':limit' => $config['pagination'],
':offset' => $offset));
    }
return self::$db->fetchMany();
}
public static function getOwnAds($author, $offset=0)
{
    if(!file_exists(__DIR__ . '/../config.php'))
    {return false;
    }
    include(__DIR__ . '/../config.php');
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
self::$db->prepareStatement("SELECT ads.*, categories.name,
users.firstname, users.lastname
FROM ads JOIN categories ON ads.category=categories.id

```

```

JOIN users on ads.author=users.id
WHERE author = :author ORDER BY ads.created_at DESC LIMIT :limit");
self::$db->executeStatement(array(':author' => $author, ':limit' =>
$config['pagination']));
return self::$db->fetchMany();
}
public static function addAd()
{
$validator=new Validator(self::$fields, self::$validationRules);
if (!$validator->checkFields(array_keys($_POST))) return false;
if (!$validator->validateFields($_POST)) return false;
    if (!$validator->validateFiles($_FILES)) return false;
    $filepath = self::uploadFiles();
    if (!$filepath) return false;
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    self::$db->prepareStatement(
'INSERT INTO ads (title, category, short_description, description,
image, phone, author)
VALUES (:title, :category, :short_description, :description, :image,
:phone, :author)'
    );
    self::$db->executeStatement(array(
        ':title' => $_POST['title'],
        ':category' => (int) $_POST['category'],
        ':short_description' => $_POST['short_description'],
        ':description' => $_POST['description'],
        ':image' => $filepath,
        ':phone' => $_POST['phone'],
        ':author' => (int) $_POST['author'],
    ));
Return self::getOneAd(self::$db->getLastInsertId('ads_id_seq'));
}
public static function updateAd()
{
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    self::$db->prepareStatement(
'UPDATE ads set title = :title, category = :category,
short_description = :short_description, description = :description,
phone = :phone, author = :author where id = :id'
    );
    return self::$db->executeStatement(array(
        ':title' => $_POST['title'],
        ':category' => $_POST['category'],
        ':short_description' => $_POST['short_description'],
        ':description' => $_POST['description'],
        ':phone' => $_POST['phone'],
    ));
}

```

```

':author' => (int) $_POST['author'],
'id' => (int) $_POST['id'],
));
    }
    public static function deleteAd()
    {
self::$db = PDOConnection::getInstance();
self::$db->getConnection();
self::$db->prepareStatement('DELETE FROM ads WHERE id = :id');
return self::$db->executeStatement(array(':id' => (int)
$_POST['id']));
    }
    /**
     * @return mixed
     */
    public static function authorize()
    {
self::$db = PDOConnection::getInstance();
self::$db->getConnection();
self::$db->prepareStatement("SELECT * FROM users WHERE email =
:email AND password = :password");
self::$db->executeStatement(array(':email' => $_POST['email'],
':password' => hash('sha512', $_POST['password'])));
return self::$db->fetchOne();
    }
    /**
     * @return bool
     */
    protected static function uploadFiles()
    {
if (count($_FILES) > 1) return false;
foreach ($_FILES as $key => $file){
$fileContent = file_get_contents($file['tmp_name']);
if ($fileContent === false) return false;
if ($file['type'] != 'image/jpeg') return false;
$filepath = self::makeFilePath($file['name']);
$filepathFull = $_SERVER['DOCUMENT_ROOT'] . $filepath;
if (file_put_contents($filepathFull, $fileContent) === false) return
false;
        }
return $filepath;
    }
    /**
     * @param $prefix
     * @param $oldname
     * @return string
     */
    protected static function makeFilePath($oldname)
    {
$name = explode('.', $oldname);

```

```

$extension = array_pop($name);
    return '/uploads/ads/' . implode('', $name) . '_' . time() .
    '.' . $extension;
    }
}

```

Программная модель категории

```

<?php
require_once(__DIR__ . '/../helpers/PDOConnection.php');
class CategoryModel
{
    private static $db;
    public static function getOneCategory($id)
    {
        self::$db = PDOConnection::getInstance();
        self::$db->getConnection();
        self::$db->prepareStatement("SELECT * FROM categories WHERE id =
        :id");
        self::$db->executeStatement(array(':id' => (int) $id));
        return self::$db->fetchOne();
    }
    public static function getManyCategories()
    {
        self::$db = PDOConnection::getInstance();
        self::$db->getConnection();
        self::$db->prepareStatement("SELECT * FROM categories");
        self::$db->executeStatement();
        return self::$db->fetchMany();
    }
}

```

Программная модель пользователя

```

<?php
if (!file_exists(__DIR__ . '/../helpers/PDOConnection.php'))
{
    return false;
}
require_once(__DIR__ . '/../helpers/PDOConnection.php');
require_once(__DIR__ . '/../helpers/Validator.php');
class UserModel
{
    private static $db;

    private static $fields = array(
        'firstname', 'lastname', 'email', 'password',
    );
    private static $validationRules = array(
        'firstname' => [
            'string' => ['length_max' => 20],

```

```

],
    'lastname' => [
        'string' => ['length_maxh' => 20],
    ],

'email' => [
    'string' => ['length_max' => 50],
],
    'password' => [
        'string' => ['length_strict' => 8],
    ],
);
/**
 * @return bool|mixed
 */
public static function addUser()
{
    $validator = new Validator(self::$fields, self::$validationRules);
    if (!$validator->checkFields(array_keys($_POST))) return false;
    if (!$validator->validateFields($_POST)) return false;
        self::$db = PDOConnection::getInstance();
        self::$db->getConnection();
        self::$db->prepareStatement(
'INSERT INTO users (firstname, lastname, email, password)
VALUES (:firstname, :lastname, :email, :password)'
        );
        self::$db->executeStatement(array(
            ':firstname' => $_POST['firstname'],
            ':lastname' => $_POST['lastname'],
            ':email' => $_POST['email'],
            ':password' => hash('sha512', $_POST['password']),
        ));
return self::getUserInfo(self::$db->getLastInsertId('users_id_seq'));
}
/**
 * @param $id
 * @return mixed
 */
public static function getUserInfo($id)
{
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    self::$db->prepareStatement("SELECT * FROM users WHERE id = :id");
    self::$db->executeStatement(array('id' => (int) $id));
        return self::$db->fetchOne();
}
/**
 * @return bool|mixed
 */
public static function updateUserInfo()

```

```

{
$validator = new Validator(self::$fields, self::$validationRules);
    if (!$validator->validateFields($_POST)) return false;
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    self::$db->prepareStatement(
'UPDATE users set firstname = :firstname, lastname = :lastname, email
= :email where id = :id'
    );
    self::$db->executeStatement(array(
        ':firstname' => $_POST['firstname'],
        ':lastname' => $_POST['lastname'],
        ':email' => $_POST['email'],
        ':id' => $_POST['id'],
    ));
    return self::$db->executeStatement(array(
        ':firstname' => $_POST['firstname'],
        ':lastname' => $_POST['lastname'],
        ':email' => $_POST['email'],
        ':id' => $_POST['id'],
    ));
}
/**
 * @return mixed
 */
public static function authorize()
{
    self::$db = PDOConnection::getInstance();
    self::$db->getConnection();
    self::$db->prepareStatement("SELECT * FROM users WHERE email
= :email AND password = :password");
    self::$db->executeStatement(array(':email' =>
$_POST['email'], ':password' => hash('sha512', $_POST['password'])));
    return self::$db->fetchOne();
}
}

```

Личный кабинет пользователя

```

<?php
if (!file_exists(__DIR__ . '/../models/UserModel.php'))
{
    header('Location: /');
    exit;
}
// include model
require_once(__DIR__ . '/../models/UserModel.php');
class UserController
{
    /**
     * AUTHORIZATION

```

```

*/
/**
 * user login
 */
public static function login()
{
    session_start();
    if(empty($_SESSION['userid']))
    {
        $_SESSION['message'] = 'Please login to enter your cabinet.';
    }
    else
    {
        $_SESSION['message'] = 'You have been already logged in';
    }
    require_once(__DIR__ . '/../views/login.php');
    exit;
}
/**
 * check login credentials
 */
public static function authorize()
{
    if(empty($_POST['email']) || empty($_POST['password']))
    {
        header('Location: /login');
        exit;
    }
    $user = UserModel::authorize();
    if (!$user)
    {
        session_start();
        $_SESSION['message'] = 'Incorrect login or password.';
        require_once(__DIR__ . '/../views/login.php');
        exit;
    }
    session_start();
    $_SESSION['userid'] = $user['id'];
    $_SESSION['group'] = $user['role'];
    header('Location: /cabinet');
}
/**
 * user logout
 */
public static function logout()
{
    session_start();
    unset($_SESSION['userid']);
    unset($_SESSION['group']);
    session_destroy();
    header('Location: /login');
}

```

```

}
/*
 * CREATE, READ, UPDATE, DELETE
 */

/**
 * add user to database
 */
public static function addUser()
{
    $user = UserModel::addUser();
    if (!$user)
    {
        session_start();
        $_SESSION['message'] = 'Error occurred while saving user.';
        exit;
    }
    header('Location: /login');
}
/**
 * show user info
 */
public static function getUserInfo()
{
    $user = UserModel::getUserInfo($_SESSION['userid'])
    if (!$user)
    {
        header('Location: /notfound');
        exit;
    }
    if(!file_exists(__DIR__ . '/../views/users/cabinet.php'))
    {
        header('Location: /form');
        exit;
    }
    require_once(__DIR__ . '/../views/users/cabinet.php');
}
/**
 * edit user info
 */
public static function editUserInfo()
{
    $user = UserModel::getUserInfo($_SESSION['userid']);
    if (!$user)
    {
        header('Location: /notfound');
    }
    if (!file_exists(__DIR__ . '/../views/users/cabinet.php'))

```

```

{
header('Location: /form');

        exit;
    }
    require_once(__DIR__ . '/../views/users/edituser.php');
}
public static function updateUserInfo()
{
    if (!UserModel::updateUserInfo())
    {
        session_start();
$_SESSION['message'] = 'Error occurred while updating user.';
        header('Location: /cabinet/edit');
        exit;
    }
    header('Location: /cabinet');
}
}

```

Страница объявления

```

<?php
require_once(__DIR__ . '/../models/AdModel.php');
require_once(__DIR__ . '/../models/CategoryModel.php');
class AdController
{
    /*
    * CREATE, READ, UPDATE, DELETE
    */
    public static function getOneAd()
    {
        $ad = AdModel::getOneAd($_GET['id']);
        if (!$ad)
        {
            header('Location: /notfound');
            exit;
        }
        if (!file_exists(__DIR__ . '/../views/ads/one_ad.php'))
        {
            header('Location: /main');
            exit;
        }
        require_once(__DIR__ . '/../views/ads/one_ad.php');
    }
    public static function getManyAds()
    {
        if (!empty($_GET['category']))
        {
            $ads = AdModel::getManyAds($_GET['category']);

```

```

}
    else
{
    $ads = AdModel::getManyAds()
    }
    if (!$ads)
    {
        header('Location: /notfound');
        exit;
    }
    if (!file_exists(__DIR__ . '/../views/ads/many_ads.php'))
    {
        header('Location: /main');
        exit;
    }
    require_once(__DIR__ . '/../views/ads/many_ads.php');
}
public static function getOwnAds()
{
    $ads = AdModel::getOwnAds($_SESSION['userid']);
    require_once(__DIR__ . '/../views/users/my_ads.php');
}
public static function getNextAds()
{
    if(!file_exists(__DIR__ . '/../config.php'))
    {
        return false;
    }
    include(__DIR__ . '/../config.php');
    if (!empty($_GET['next']) || $_GET['next'] > 0)
    {
        $next = $_GET['next'];
    }
    else
    {
        $next = 0;
    }
    session_start();
    $_SESSION['page'] = $next;
return AdModel::getManyAds($offset=$next* $config['pagination']);
}
/**
 * add ads to database
 */
public static function addAd()
{
    // add user to database
    $ad = AdModel::addAd();
    if (!$ad)

```

```

    {
        session_start();
        $_SESSION['message'] = 'Error occurred while saving ad.
Please try again later.';
        exit;
        header('Location: /ads');
    }
/**
 * edit ad
 */
public static function editAd()
{
    $ad = AdModel::getOneAd($_POST['id']);
    $categories = CategoryModel::getManyCategories();
    if ($_SESSION['group'] != 'admin' && $_SESSION['userid'] !=
$ad['author'])
    {
        $_SESSION['message'] = 'You are not allowed to do this
action';
        header('Location: /cabinet/ads');
        exit;
    }
    if (!$ad)
    {
        header('Location: /notfound');
        exit;
    }
    require_once(__DIR__ . '/../views/ads/edit_ad.php');
}
public static function updateAd()
{
    $ad = AdModel::getOneAd($_POST['id']);
if ($_SESSION['group'] != 'admin' && $_SESSION['userid'] !=
$ad['author'])
    {
        $_SESSION['message'] = 'You are not allowed to do this
action';
        header('Location: /cabinet/ads');
        exit;
    }
    if (!AdModel::updateAd())
    {
        $_SESSION['message'] = 'Error occurred. Please try again
later.';
        exit;
    }
    header('Location: /ads');
}
public static function deleteAd()
{

```

```
$ad = AdModel::getOneAd($_POST['id']);
    if ($_SESSION['group'] != 'admin' && $_SESSION['userid'] !=
$ad['author'])
    {
$_SESSION['message'] = 'You are not allowed to do this action';
    header('Location: /cabinet/ads');
    exit;
    }
if (!$ad)
{
    //header('Location: /notfound');
}
AdModel::deleteAd();
header('Location: /cabinet/ads');
}
}
```

ПРИЛОЖЕНИЕ В. КОМПАКТ-ДИСК

Содержание:

1. Пояснительная записка к выпускной квалификационной работе.
2. Разработанный программный продукт.
3. Презентация.